Oracle® Fusion Cloud EPM

Administering FreeForm

F51503-29

Oracle Fusion Cloud EPM Administering FreeForm,

F51503-29

Primary Author: EPM Information Development Team

Contents

Documentation Accessibility

Documentation Feedback

1 Creating and Running an EPM Center of Excellence

2 Creating a FreeForm App

Understanding FreeForm 2-1
FreeForm App Sources 2-3
Important Considerations for FreeForm Apps 2-4
Creating a FreeForm App 2-5
- Creating a FreeForm App Using an Outline File or Snapshot 2-6
- Creating a FreeForm App Using the Application Creation Wizard 2-7

Managing Application Ownership 2-9
Frequently Asked Questions (FAQ) About FreeForm 2-9

3 Getting Started

About the Home Page 3-1

4 Setting Up Access Permissions

About User and Role Management 4-1
Application Artifacts That Can Be Assigned Permissions 4-1
Types of Access Permissions 4-2
Managing Permissions to Artifacts 4-4
- About Assigning Permissions to Artifacts, Rules, and Folders 4-4
- Adding, Changing, and Removing Permissions to Artifacts and Folders 4-5
- Adding, Changing, and Removing Permissions to Rules and Rules Folders 4-5

Reporting on Access Permissions 4-6
- Working with Access Permissions Reports 4-7

Retrieving Changed Information from Access Control 4-7

5 Managing Applications

Application Overview 5-1
Refreshing the Application 5-2
Managing Cubes 5-2
Viewing and Managing Cubes 5-2
Adding Cubes 5-3
Clearing Cubes 5-4
About Clearing Cubes 5-4
Creating Clear Cube Jobs 5-4
Starting and Deleting Clear Cube Jobs 5-5
Scheduling Clear Cube Jobs 5-6
Improving Cube Performance 5-6
Managing Dimensions 5-7
Filtering the Dimension View by Cube 5-8
Creating Dimensions 5-9
Setting User Preferences 5-9
Administering Variables 5-10
Working with Substitution Variables 5-10
About Substitution Variables 5-10
Creating and Assigning Values to Substitution Variables 5-10
Deleting Substitution Variables 5-11
Working with User Variables 5-11
About User Variables 5-11
Creating User Variables 5-12
Managing User Variables 5-13
Setting Limits for User Variables on Forms 5-13
Deleting User Variables 5-13
Viewing Activity Reports 5-14
Importing and Exporting Data and Metadata 5-14
Importing Metadata 5-15
Creating the Metadata Import File 5-15
Loading the Metadata Import File 5-18
Exporting Metadata 5-20
Importing and Exporting Data 5-22
Importing Data 5-22
Driver Member Casting Errors 5-25
Exporting Data 5-25
Viewing Data Import and Export Status 5-28
Validating the Essbase Outline 5-28

Creating and Refreshing Application Databases 5-29
Creating Application Databases 5-29
Before Refreshing the Database 5-30
Refreshing Application Databases 5-30
Validating Metadata 5-31
Adding an Aggregate Storage Outline to an Application 5-32
About Aggregate Storage 5-32
Aggregate Storage Outline Cube Characteristics 5-32
Process for Adding an Aggregate Storage Database to an Application 5-33
Removing an Application 5-33
Setting the Daily Maintenance Process Start Time 5-33
Uploading and Downloading Files Using the Inbox/Outbox Explorer 5-35

6 Using the Member Selector

Working with Members 6-1
Making Selections 6-2
Member Relationships 6-4
Using Wildcards in Searches 6-5
Selecting Attribute Values as Members 6-6
Selecting Members for Forms 6-8
Selecting Substitution Variables as Members 6-9
Selecting User Variables as Members 6-10
Selecting UDAs as Members 6-11

7 Connecting Environments in EPM Cloud

About Connecting EPM Cloud Environments 7-1
Considerations for Migrating EPM Cloud Connections 7-3
Creating, Editing, and Deleting Connections to Other EPM Cloud Environments 7-4
Connecting to External Web Services 7-6
Specifying Advanced Options for External Connections 7-7
Navigating Across EPM Cloud Environments 7-7
Customizing Navigation Flows to Access Other EPM Cloud Environments 7-8
Grouping Cards from Other EPM Cloud Environments into Clusters 7-9
Configuring Cards with Tabs from Multiple EPM Cloud Environments 7-12
Using Direct URLs to Integrate Connected Environments 7-15
Copying Direct URLs for Artifacts 7-16
Exporting All URLs to a CSV File 7-16

8 Designing Custom Navigation Flows

Understanding Navigation Flows 8-1
- What Can Be Customized in the Business Process Interface? 8-1
- Navigation Flow Customization Categories 8-2
- Navigation Flow Permissions 8-3
- Predefined Navigation Flows 8-3
- Understanding Navigation Flow Status after Import 8-3

Viewing and Working with Navigation Flows 8-4
- Navigation Flow Design Best Practices and Naming Considerations 8-5
- Creating and Duplicating Navigation Flows 8-5
- Editing a Navigation Flow 8-6
- Activating and Deactivating Navigation Flows 8-7
- Using Validate to Find Missing Artifacts in Navigation Flows 8-7
- Resolving Navigation Flows that Display a Warning Icon 8-8
- Customizing Labels for Cards, Tabs, and Clusters 8-9
- Customizing Icons for Cards and Vertical Tabs 8-9
- Hiding and Unhiding Clusters, Cards, and Tabs 8-10
- Changing the Display Order of Cards on the Home Page 8-10
- Adding Cards 8-11
- Adding Tabs to a Tabular Page 8-12
- About Using URLs to Embed Third-Party Pages in EPM Cloud Applications 8-15
- Removing Navigation Flows, Cards, and Tabs 8-16
- Grouping Cards into Clusters 8-16
- Reloading a Navigation Flow 8-18
- Switching Navigation Flows at Runtime 8-18
- Sharing the Current Page URL for Cards, Tabs, and Sub Tabs 8-19

9 Editing Dimensions in the Simplified Dimension Editor

About Editing Dimensions in the Simplified Dimension Editor 9-2

Dimension Overview 9-2
- About Dimensions and Members 9-3
- About Sparse and Dense Dimensions 9-3
- About Dimension Hierarchies 9-3
- About Custom Dimensions 9-3
  - Aggregation Options 9-4
  - Storage Options 9-4
- About Entities 9-6
- About Accounts 9-6
  - Account Types 9-6
  - Saved Assumptions 9-10

Data Type 9-10
Accounts, Entities, Periods, and Cubes 9-10
Accessing the Simplified Dimension Editor 9-11
Working in the Edit Member Properties Grid 9-11
Switching to Another Dimension 9-12
Customizing the Column Layout 9-12
Context Menu Options 9-14
Viewing Ancestors 9-14
Showing Member Usage in an Application 9-15
Focusing Your Editing 9-15
Finding Members 9-16
Moving Members 9-16
Sorting Members 9-17
Moving Members to Another Hierarchy 9-17
Working with Member Formulas 9-17
Copying Member Names from Microsoft Excel 9-18
Editing Member Properties 9-19
Accessing Edit Member Properties 9-19
Adding Members 9-22
Editing Members 9-23
Deleting Members 9-24
Adding Shared Members 9-24
Assigning Access to Dimension Members 9-25
Adding, Editing, and Removing Access to Members 9-25
About Effective Access Permissions to Shared Members 9-26
Editing Dimension Properties 9-27
Working with Attributes 9-28
Deleting Attributes 9-30
Working with UDAs 9-30
Working with Attribute Values 9-31

10 Designing Dashboards

The Power of Dashboards 10-2
Creating and Managing Dashboards 10-4
About Dashboard Versions 10-4
Viewing Dashboards 10-6
Converting Dashboard 1.0 Dashboards to 2.0 10-8
Creating and Managing Dashboards 1.0 10-9
Concepts in Designing 1.0 Dashboards 10-9
Creating Dashboard 1.0 Dashboards 10-11
Creating Dashboards Containing Master Forms and Details 10-13

About Your Dashboard's Layout - Version 1.0 10-14
About the Gauge Chart Type 10-14
About the Tile Chart Type 10-16
Customizing Dashboard Colors 10-18
About Global and Local POVs in 1.0 Dashboards 10-19
Dashboard POVs and Valid Intersections 10-21
Creating and Managing Dashboards 2.0 10-21
Concepts in Designing Dashboard 2.0 Dashboards 10-22
Object Palette 10-24
Dashboard Toolbar 10-25
Dashboard Workspace 10-26
Properties Panel 10-27
Dashboard Components 10-31
Considerations for Dashboard 2.0 10-32
Creating Dashboard 2.0 Dashboards 10-33
Working with the Member Selector When Designing Dashboards 2.0 10-34
About Forms 2.0 Grids in Dashboard 2.0 10-35
About the Geomap Chart Type 10-36
About the Pyramid Chart Type 10-38
About the Waterfall Chart Type 10-39
Using Tables in Dashboards 10-41
About the Gauge Chart Type for Dashboard 2.0 10-43
About the Radar Chart Type for Dashboard 2.0 10-46
About the Combination Chart Type for Dashboard 2.0 10-47
About the Tile Chart Type for Dashboard 2.0 10-49
About Global and Local POVs in Dashboard 2.0 10-51
About Quick Analysis 10-54

11 Designing Infolets

About Infolets 11-1
Anatomy of an Infolet 11-3
Determining Infolet Content 11-6
Designing Forms for Infolets 11-7
Designing Charts for Infolets 11-7
Using the Infolets Designer 11-7
Creating Infolets 11-10
Working with Infolets 11-11
Customizing the Interface to Access Infolets 11-12

12 Bring Your Own ML: About Machine Learning Model Import

Considerations for Bring Your Own ML 12-2
Importing an ML Model 12-3
Deploying an ML Model to Planners 12-4

13 Defining Valid Intersections and Cell-Level Security

Defining Valid Intersections 13-1
- Understanding Valid Intersections 13-1
  - Valid Intersection Groups 13-2
  - Valid Intersection Rules 13-2
  - Anchor and Nonanchor Dimensions 13-3
  - Valid Intersection Examples 13-3
  - Redundancy or Overlap in Valid Intersection Rules 13-6
  - Shared Members and Valid Intersection Rules 13-6
  - Substitution Variables and Valid Intersection Rules 13-6
  - Evaluation Order 13-7
- Creating Valid Intersections 13-7
- Managing Valid Intersections 13-8
  - Viewing Valid Intersections 13-8
  - Filtering Valid Intersections 13-9
  - Importing and Exporting Intersections 13-10
  - Changing the Valid Intersection Group Evaluation Order 13-12
  - Disabling and Enabling Valid Intersection Groups 13-13
  - Editing Details for a Valid Intersection Group 13-13
  - Duplicating Valid Intersection Groups 13-14
  - Deleting a Valid Intersection Group 13-15
- Suppressing Invalid Data in Forms 13-15
- Clearing Invalid Data 13-15
  - About Invalid Data 13-16
  - Working With Invalid Intersection Reports 13-16
  - Clearing Data at Invalid Intersections 13-17
- Working with Valid Intersections 13-17
  - Working with Valid Intersections in Application Forms 13-17
  - Working with Valid Intersections in Calculation Manager Rule Runtime Prompts 13-18

Defining Cell-Level Security 13-19
- Understanding Cell-Level Security 13-19
- Creating Cell-Level Security Definitions 13-20
- Viewing Cell-Level Security Definitions 13-21
- Filtering Cell-Level Security Definitions 13-22
- Importing and Exporting Cell-Level Security Definitions 13-22

Testing Cell-Level Security 13-24
Reordering the Cell-Level Security Definitions List 13-25

14 Defining Data Maps and Creating File-Based Integrations

15 Managing Jobs

How Jobs Save You Time 15-1
Viewing the Status of Jobs 15-1
Scheduling Jobs 15-2
Job Types 15-3
Scheduling Jobs to Run Later 15-5
Scheduling Hourly Jobs 15-6
Editing and Canceling Jobs 15-9
Duplicating Jobs 15-9
Canceling Rules Jobs and Ruleset Jobs 15-10
Downloading Export Files From Your Outbox 15-10

16 Auditing Tasks and Data

Auditing Overview 16-1
Enabling Audit Tracking 16-2
Viewing Audit Details 16-3

17 Managing Data Validation

Creating and Updating Data Validation Rules 17-1
Formatting Cells 17-4
Viewing Data Validation Rules 17-5
Order of Evaluation and Execution for Data Validation Rules 17-5
Conditions Supported by the Rule Builder 17-6
If Condition Values 17-6
Then Condition Values 17-13
Range Condition Values 17-13
Data Validation Conditional Operators 17-14
Data Validation Rule Scenarios 17-15

18 Managing Application and System Settings

What Application and System Settings Can I Specify? 18-1
Defining User Variables 18-6

Customizing Your Display 18-7
Announcing Upcoming Events 18-7
Specifying Artifact Labels 18-8
Which Artifact Labels Can be Localized? 18-8
Working With the Artifact Labels Grid 18-9
Adding Languages and Defining Localized Artifact Labels 18-9
Exporting and Importing Artifact Labels for Editing 18-10

19 Accessing More Administrative Tasks

About the Navigator Menu 19-1
Administering Data Load Settings 19-2
Importing Using Data Integration 19-3
Administering Action Menus 19-3
Creating and Updating Action Menus 19-3
Working with Action Menu Items 19-4
Defining Action Menu Items 19-4
Administering Alias Tables 19-7
About Aliases 19-7
About Alias Tables 19-8
Working with Alias Tables 19-8
Specifying a Default Alias Table and Setting Member and Alias Display Options 19-9
Administering Dimensions 19-10
About Dimensions 19-10
Working with Dimension Hierarchies 19-10
About Custom Dimensions, Entities, Accounts, Periods, and Cubes 19-12
Adding or Editing User-Defined Custom Dimensions 19-12
Working with Members 19-14
Working with Attributes 19-22
Working with Attribute Values 19-25
Customizing Calendars 19-27
Setting up Dynamic Time Series Members 19-31
Working with UDAs 19-33
Working with Member Formulas 19-34
Administering Forms 19-39
About Forms 19-39
Form Components 19-39
Form Design Considerations 19-41
Understanding Implied Sharing in Forms 19-42
Creating Forms 19-43
Defining the Layout 19-44
About Precision Settings 19-56

Setting Form Precision and Other Options 19-56
Moving Data Using Smart Push 19-57
Creating Asymmetric Rows and Columns 19-57
Adding Formula Rows and Columns 19-57
Defining Form Page and Point of View 19-58
Designing Specific Types of Forms 19-59
Working with Forms and Form Components 19-64
Selecting and Opening Forms and Folders 19-64
Previewing Forms 19-65
Printing Form Definitions 19-66
Searching for Forms 19-66
Editing Forms 19-66
Moving, Deleting, and Renaming Forms 19-67
How Cell Formatting in Smart View Persists in FreeForm 19-68
Managing Forms and Folders 19-68
Creating Folders 19-69
Working with Folders 19-69
Administering Rules 19-70
About Rules 19-70
Adding and Removing Rules in Forms 19-71
Setting Business Rule Properties 19-72
Viewing Rules Usage 19-73
About Runtime Prompts 19-76
Understanding Runtime Prompts 19-76
Using Groovy Rules 19-80
About Creating a Groovy Business Rule 19-80
Java API Reference for Groovy Rules 19-81
Groovy Business Rule Examples 19-81
Groovy Business Rule Tutorial Videos 19-82
Groovy Rule Business Scenarios 19-82
Administering Rules Security 19-94
Assigning Access to Rules 19-95
Adding, Editing, and Removing Access to Rules 19-95
Assigning Access to Groovy Templates 19-96
Administering Smart Lists 19-96
Working with Smart Lists 19-96
Synchronizing Smart Lists in Reporting Applications 19-97
Setting Smart List Properties 19-98
Defining Smart List Entries 19-99
Previewing Smart Lists 19-100
Displaying #MISSING with Smart Lists 19-100
Administering Task Lists 19-100

Working with Task Lists 19-101
Adding Instructions to Task Lists 19-101
Adding Tasks to Task Lists 19-101
Editing Task Lists 19-104
Editing Tasks 19-104
Copying and Moving Tasks 19-105
Moving and Reordering Task Lists 19-105
Clearing Task Lists 19-106
Deleting Tasks and Task Lists 19-106
Assigning Access to Task Lists 19-106
Adding Access to Task Lists 19-107
Changing and Removing Access to Task Lists 19-107
Clearing Cell Details 19-108
Copying Data 19-109
Administering Application Diagnostics 19-110
About Application Diagnostics 19-110
Assumptions 19-111
How Application Diagnostics Works 19-111
Using Application Diagnostics Graphs 19-111
Launching Application Diagnostics 19-112
Modifying Artifacts for Optimal Performance 19-113

A Naming Restrictions

Naming Restrictions for Applications and Databases A-1
Naming Restrictions for Dimensions, Members, and Aliases A-2
Dimension and Member Names in Calculation Scripts, Report Scripts, Formulas, Filters, and Substitution Variables A-5
Restrictions for User and Group Names A-5

B Form Formula Functions

About Form Formula Functions B-1
Working with Formulas B-1
Creating Formulas B-1
Editing Formulas B-2
Deleting Formulas B-2
Formula Functions B-2
Arguments B-3
Numeric Arguments B-4
Row, Column, or Cell Reference Arguments B-4
Property Arguments B-6

Embedded Functions as Arguments B-8
Abs B-8
Average B-9
AverageA B-10
Count B-10
CountA B-11
Difference B-12
Eval B-13
IfThen, If B-13
Notes on Conditions B-15
Complex Conditions B-16
Max B-16
Min B-17
Mod B-17
PercentOfTotal B-18
Pi B-19
Product B-19
Random B-20
Round B-20
Sqrt B-21
Sum B-21
Truncate/Trunc B-22
Variance/Var B-23
VariancePercent/VarPer B-25

C Using Smart View to Manage Applications

About Managing Applications in Smart View C-1
Installing Smart View and the Admin Extension for Application Management C-2
Controlling the Display of Application Management Options in Smart View C-4
Downloading the Application Templates C-4
Downloading the Template in Smart View C-4
Downloading the Application Template Zip File from the Web Interface C-5
Creating an Application C-6
Working with Artifacts in the Application Template C-7
About Working with Artifacts in the Application Template C-7
Application Definition C-8
Dimension Definition C-11
Attribute Dimension Definition C-17
Data Definition C-18
Substitution Variable Definition C-19
Security Definition C-21

Advanced Settings Definition C-22
Updating an Application in Smart View C-26
Deleting an Application C-27
Planning Admin Extension and Office AutoCorrect C-27

D Using Smart View to Import and Edit Application Metadata

About Using Smart View to Work with Application Metadata D-1
Installing Smart View and the Admin Extension for Editing Dimensions D-2
Using Smart View Grids to Import and Edit Application Metadata D-3
About the Smart View Grid D-4
The Smart View Grid and Ribbon Display D-4
Guidelines for Using the Smart View Grid D-6
Default Metadata Dimension Member Properties D-7
Importing Dimensions in Smart View D-9
Importing Dimensions in Smart View for Office D-9
Importing Dimensions in Smart View (Mac and Browser) D-10
Editing Members in Smart View D-12
Adding Application Members in Smart View D-12
Adding Members in Smart View D-13
Guidelines for Adding Members in Smart View D-14
Moving Members in Smart View D-14
Guidelines for Moving Members in Smart View D-15
Working with Attribute Dimensions D-15
Adding Attribute Dimension Members in Smart View D-15
Adding Attribute Dimension Members in Smart View for Office D-15
Adding Attribute Dimension Members in Smart View (Mac and Browser) D-17
Associating Attribute Dimension Members with Dimension Members D-19
Designating Shared Members in Smart View D-20
Refreshing Databases D-21
Refreshing Databases in Smart View for Office D-21
Refreshing Databases in Smart View (Mac and Browser) D-21
Planning Admin Extension and Office AutoCorrect D-23

E FreeForm Best Practices

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

To provide feedback on this documentation, click the feedback button at the bottom of the page in any Oracle Help Center topic. You can also send email to epmdoc_ww@oracle.com.

1

Creating and Running an EPM Center of Excellence

A best practice for EPM is to create a CoE (Center of Excellence).

An **EPM CoE** is a unified effort to ensure adoption and best practices. It drives transformation in business processes related to performance management and the use of technology-enabled solutions.

Cloud adoption can empower your organization to improve business agility and promote innovative solutions. An EPM CoE oversees your cloud initiative, and it can help protect and maintain your investment and promote effective use.

The EPM CoE team:

- Ensures cloud adoption, helping your organization get the most out of your Cloud EPM investment
- Serves as a steering committee for best practices
- Leads EPM-related change management initiatives and drives transformation

All customers can benefit from an EPM CoE, including customers who have already implemented EPM.

How Do I Get Started?

Click to get best practices, guidance, and strategies for your own EPM CoE: Introduction to EPM Center of Excellence.

Learn More

- Watch the Cloud Customer Connect webinar: Creating and Running a Center of Excellence (CoE) for Cloud EPM
- Watch the videos: Overview: EPM Center of Excellence and Creating a Center of Excellence.
- See the business benefits and value proposition of an EPM CoE in *Creating and Running an EPM Center of Excellence*.

2

Creating a FreeForm App

FreeForm apps use an open dimensional cube construct allowing you to create cubes with any dimension combination you need.

Related Topics

- Understanding FreeForm
- FreeForm App Sources
- Important Considerations for FreeForm Apps
- Creating a FreeForm App
- Managing Application Ownership
- Frequently Asked Questions (FAQ) About FreeForm

Understanding FreeForm

FreeForm is a Reporting, Analysis, and Planning application. FreeForm enables you to create an application with the cubes and dimensions of your choice without being constrained by the cube and dimension limitations imposed by standard applications. With FreeForm, you can model and build your own cubes while preserving the ability to leverage business process functionalities.

FreeForm can be accessed using Oracle Smart View for Office or Oracle Enterprise Performance Management Cloud screens; they support Groovy scripts for custom functions, and business rules for calculations.

You can create applications with multiple cubes or you can create an application with a single cube and then add more cubes later (up to 12 cubes total).

Table 2-1 FreeForm Features at a Glance

What is FreeForm?	What can I create with FreeForm?	What use cases can I solve with multi-cube FreeForm?	What's special about multi-cubeFreeForm?
FreeForm is an EPM application with no dimension prerequisites, to create multiple cubes with full flexibility in dimensions based on business use case needs.	Create up to 12 cubes that can use any of the 29 dimensions in any combination for each cube based on use case needs. (26 custom dimensions plus three account, period, and entity dimension types.)	• Unified Reporting • Essbase SaaS Migration • Excel model migration • Financial and Operational Reporting or Planning across any industry	• Only place in EPM Cloud where you can create a reporting application with only ASO cubes • Only place in EPM where you can import Oracle Essbase files (using Migration or outline load) to create an application

Table 2-1 (Cont.) FreeForm Features at a Glance

What is FreeForm?	What can I create with FreeForm?	What use cases can I solve with multi-cube FreeForm?	What's special about multi-cubeFreeForm?
Oracle EPM Cloud Platform provides the technology framework guardrails and features to help customers succeed.	Create an application that is all ASO reporting or all Hybrid BSO planning cubes or a combination of ASO and BSO.	• Financial Reporting • Operational Reporting • Revenue Planning • Expense Planning • Demand Planning • Supply Planning • And so on	Gives full control to customers to work with Sales to create a best fit solution that best meets their requirements for reporting or planning

Note:

The following features are not available in FreeForm because these features require the provided Scenario, Version, Period, and Year dimensions to be present:

- Predictive Planning
- Auto-Predict
- IPM Insights

In FreeForm, the Scenario, Version, Period, and Year dimensions can be defined as custom dimensions.

In addition, the following features are not available in FreeForm:

- Sandboxes
- Approvals
- Task Manager

About Multi-Cube FreeForm Apps

- The maximum number of custom dimensions allowed in a FreeForm app is 26.
- Cubes can share dimensions or have standalone dimensionality.
- All dimensions can be custom; or native account, period, and/or entity dimensions can be inherited by the cubes.
- Cross-cube data maps, Smart Push, and Copy Data support are available for FreeForm app types.

Multi-cube FreeForm apps are constrained by these existing business process assumptions:

- Members must be unique across all the cubes within FreeForm.
- Governors are applicable at the application level and not at the cube level.
- The input outlines must not be enabled to allow duplicate member names.

Videos

Your Goal	Learn How
Get an introduction to FreeForm.	Introduction to FreeForm Apps in Cloud EPM
Create FreeForm apps from on-premises Essbase outline (OTL) files and snapshots.	Creating FreeForm applications from on-premises Essbase outline files and snapshots
Create multi-cube FreeForm apps with an Essbase outline	Creating Multi-Cube FreeForm Apps from Essbase Outlines
Create FreeForm apps with a Hybrid BSO cube	Creating FreeForm Applications with a Hybrid BSO Cube
Create FreeForm apps with an ASO or reporting cube	Creating FreeForm Applications with a Reporting/ASO Cube
Create multi-cube FreeForm apps with a BSO or ASO cube	Creating Multi-Cube FreeForm Apps from BSO and ASO Cubes
Learn how to create FreeForm apps using an Excel template. Using the Excel template, you learn how to define application properties, manage cubes, create dimensions and members, attributes, access permissions, and load data into the FreeForm application.	Creating Multi-Cube FreeForm Applications with an Excel Template - Part 1
Learn how to update FreeForm apps using an Excel template and verifying changes in the web interface. After creating your FreeForm application using an Excel template, learn how you can use the same template to make modifications to dimensions, members, substitution variables, and security. Then, in the Planning web interface, assign Application Management options, review the application properties and definition, and the data you imported.	Creating Multi-Cube FreeForm Applications with an Excel Template - Part 2

FreeForm App Sources

You can build FreeForm apps by creating dimensions using the application creation wizard. Alternatively, you can use an outline (OTL) file or snapshot from an on-premises release 11.1.2.4.xxx (or later) single cube Oracle Essbase application to create a FreeForm app.

When you use an Essbase cube OTL file or Essbase application snapshot as the source for a FreeForm app, Account, Period (time), and Entity (country) dimensions are created and mapped automatically. Additionally, Version and Scenario dimensions of the Essbase application are created as custom dimensions. As a result, out of the box Workflow functionality is not supported in FreeForm apps.

Service Administrators build a FreeForm app using the application creation wizard or these sources:

- **An Outline file from on-premises release 11.1.2.4.xxx (or later) single cube Essbase application**

  You may use an OTL file from an on-premises release 11.1.2.4.xxx (or later) single cube Essbase application to create the structure of the FreeForm app. New cubes may be added or created for this FreeForm app at a later time.

  Because the OTL file name is assigned to the cube created for the FreeForm app, the file name must be eight characters or less. Longer file names will cause the process to fail.

  Generally, the OTL file is available in the `EssbaseServer/essbaseserver1/app/<app_Name>` directory within your on-premises Essbase deployment.

  Because the OTL file does not contain application data, you must extract Essbase data to a file. Use MaxL, Essbase Studio, or another tool to export data from an Essbase application. The data must be exported in an Essbase data file format.

  After creating the application, import the extracted data, selecting Essbase as the source type. You can import from a locally stored data file or from a file that was uploaded to the Oracle Enterprise Performance Management Cloud inbox. For detailed instructions, see Importing Data.

- **An application snapshot from an on-premises release 11.1.2.4.xxx (or later) single cube Essbase application**

  This option simplifies application creation by migrating an existing Essbase application snapshot (ZIP file) that was created using Migration. This process automates dimensions loading, substitution variables creation, calculation scripts conversion as graphical rules, and data load. New cubes may be added or created for this FreeForm app at a later time.

Important Considerations for FreeForm Apps

> **Note:**
>
> Do not attempt to create FreeForm apps by importing Oracle Essbase Migration snapshots into an Oracle Enterprise Performance Management Cloud environment. Importing snapshots using Migration is supported only for snapshots created from a previous FreeForm app.

- Data maps and other multi-cube features are unavailable if you create a single cube FreeForm app using the application creation wizard.
- Essbase snapshots with Custom Defined Functions (CDF) and CDF references cannot be used to create FreeForm apps.
- Make sure that the OTL file or Essbase application snapshot that you are using does not contain objects with system restricted names; for example, a member named `FY02` in the Year dimension. You must rename or remove such restricted names before generating the OTL file or snapshot that you plan to use to create the application. See Naming Restrictions.
- Ensure that the OTL file or Essbase application snapshot that you are using does not have member and alias names that are longer than 80 characters. If member and alias names

are longer than 80 characters, you must remove or shorten them or the import will not be successful. See Naming Restrictions.

- If the OTL file that you are using contains Year Total and Beginning Balance members, the Beginning Balance member must be the first child under the Period dimension, and Year Total must be the second child under Period dimension. These members require special handling and can be moved to appropriate positions by modifying the .csv files in the Migration snapshot before restoring the snapshot.
- If you plan to use an OTL file or snapshot stored in your EPM Cloud environment as the source, upload the file before starting the application creation process.

  Use the `uploadFile` EPM Automate Utility command or Migration to upload the OTL file or snapshot to an EPM Cloud environment.
- Imported data may not be editable in FreeForm apps. At times, the data is set as Read-only.
- Member outline must be unique in the application.
- If you create a FreeForm app by importing an OTL or snapshot, you can add or create new cubes for this FreeForm app at a later time.
- If you create a FreeForm app by adding new cubes in the application creation wizard, an Essbase cube OTL or Essbase application snapshot cannot be imported into this FreeForm app at a later time to create new cubes.

Do not use the following reserved words to name dimensions and members in FreeForm:

Table 2-2 Reserved words that should not be used as member names in FreeForm

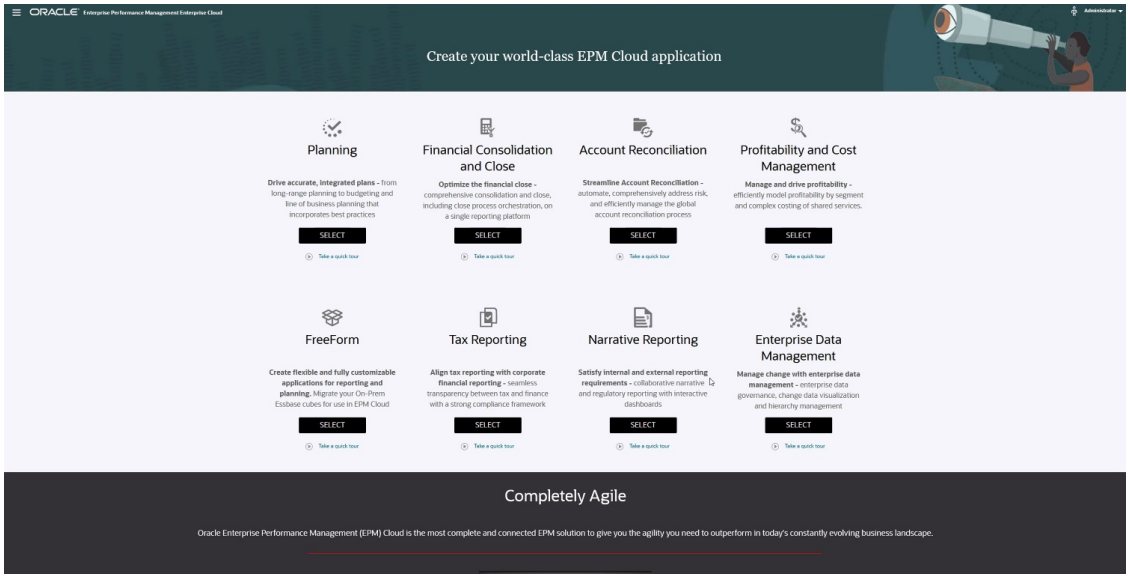
Reserved words			
BU Version_1	Users	Attribute Dimensions	HSP_Entity
ConsolidatedData	Groups	Default	HSP_Period
BaseData	Calendars	Task Lists	HSP_Version
SandBoxData	Currencies	Menus	HSP_XCRNCY
Super User	Predefined	CalcMgrRules	HSP_Years
Strategic Planner	FX_Tables	CalcMgrRulesets	HSP_View
Service Administrator	Forms	CalcMgrVariables	HSP_Metric
CalcMgrTemplates	Aliases	HSP_Rates	
Root	Cubes	HSP_Scenario	
Dimensions	Planning Units	HSP_Account	

Creating a FreeForm App

Before you create a FreeForm app, review the important considerations and the reserved words that should not be used as member names in FreeForm apps. See Important Considerations for FreeForm Apps.

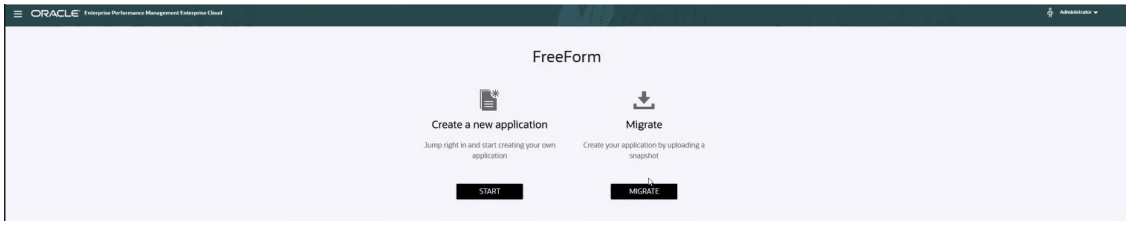
The Landing Page

The landing page is your starting point for creating a FreeForm app and for viewing overview videos that help you get started.



FreeForm App Creation Options

To create a FreeForm app, click **SELECT** under the FreeForm description. You'll see the following options:



- **Create a new application**: Click **START** to manually create a FreeForm app based on your own models using custom dimensions. During this process, you can map the custom dimensions that you want to use as Entity, Period, and Account dimensions, and you can also add other custom dimensions to support your model. See Creating a FreeForm App Using the Application Creation Wizard.
- **Migrate**: Click **MIGRATE** to create a FreeForm application from a snapshot that you previously uploaded to the environment. Only one snapshot is allowed. See Creating a FreeForm App Using an Outline File or Snapshot.

Creating a FreeForm App Using an Outline File or Snapshot

To continue creating a FreeForm app using an outline file or snapshot:

1. Select the location of the source OTL file or the Essbase application snapshot:
   - Select **Local** to access the source OTL file or snapshot from the computer from which you are currently accessing Oracle Enterprise Performance Management Cloud.

     For **Import File**, click **Choose File** and then select the source OTL file or snapshot.
   - Select **Inbox** to access the source OTL file or snapshot from your EPM Cloud environment.

     From **Select One**, select the source OTL file or snapshot.
2. Click **Next**.

3. Review application information and then click **Create**.

   When application creation is complete, EPM Cloud Home page is displayed.

4. Make sure that errors were not reported during application creation.
   - Open the Jobs console by clicking **Application**, and then **Jobs**
   - Verify that the **Create Database** and **Process Outline** activities finished without errors. Correct any reported errors.

5. **Optional:** If you created the application using an OTL file as the source, import application data. See Importing Data.

   If you used a snapshot as the application source, the application creation process automatically imports data.

6. Create application users in identity domain and assign predefined roles as needed. See Managing Users and Roles in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

7. Set up access permissions as needed. See Setting Up Access Permissions.

Creating a FreeForm App Using the Application Creation Wizard

To continue creating a FreeForm app using the application creation wizard:

1. For **No. Of Cubes**, select the number of cubes you'd like to use to create your FreeForm app. You can add up to 12 cubes now, or you can start with one cube and then add more cubes later on the Application Overview page.

   > **Note:**
   >
   > If you create a single cube FreeForm app, data maps and other multi-cube features are unavailable until you add cubes to the FreeForm app. Also for a data map to work there needs to be at least one block storage (BSO) cube, and the source for the data map needs to be a BSO cube.

2. In **Cube Name**, enter a name for each cube, maximum eight characters.

3. Select **Is ASO**, if the cube is an aggregate storage cube. If the **Is ASO** checkbox is cleared, then the cube is a Hybrid block storage cube.

   > **Note:**
   >
   > Reporting applications that use ASO cubes, generally, store data in aggregate views and can handle a higher number of dimensions than BSO without adversely affecting performance.

4. Click **Next**.

   The **Create Dimensions** page is displayed. From this screen, you can create up to three dimensions and assign one each to Account, Entity, and Period default dimension types. Additional dimensions can be added later in the web interface.

> **Note:**
>
> FreeForm apps do not require standard dimensions. You can create an application with only custom dimensions containing members of your choice. The dimensions you create and its structure are not governed by the constraints imposed on standard application dimensions.

5. In **Create Dimensions**, create and map default application dimensions. Complete a step:
   - To create dimensions and map them to default dimension types, select the **Enabled** check box and then type in dimension names.

     > **Note:**
     >
     > Enabling dimensions is not automatic. Just like custom applications, you must manually enable dimensions for FreeForm apps. You can do this now, or you can enable them later.

   - To create a dimension and assign a dimension type, select the check box in a row and then type in a dimension name.
   - To create a shell application, do not change anything in this screen.

     You can add and map dimensions to shell applications after the application creation process is complete.
6. Click **Next**.
7. In **Review**, verify the settings that you selected and then click **Create**.

   The application creation process may take a few minutes.
8. In the Application Creation Status screen, click **OK**.

   The Home page is displayed.
9. **Optional:** Create custom dimensions, if needed.
   a. Click **Application**, then **Overview**, and then select the **Dimensions** tab.
   b. For **Cube**, select a cube that you created for your FreeForm app or select **All**.
   c. Click **Create**.
   d. Complete the dimension details for each dimension you want to add. For descriptions of the dimension properties, see Editing Dimension Properties.

      For FreeForm applications, you can select the dimension type. Dimension types can include:

      - Account
      - Entity
      - Period
      - Custom

      Notes about dimension type:

      - A dimension type is not listed if you've already created it (for Account, Entity, Period) or if you've exceeded the maximum number of dimensions (for Custom).

- You can't edit the dimension type after you've created the dimension.

Select **Enabled** to indicate whether the dimension is to be used in this cube.

> **Note:**
>
> Enabling dimensions is not automatic. Just like custom applications, you must manually enable dimensions for FreeForm apps.

e. Click **Done**.

f. Click **Action** and then **Refresh Database** to refresh the cube.

10. **Optional:** Import application data. See Importing Data.
11. **Optional:** If you created a shell application with a view to import a snapshot, access Migration, then delete the application, and then complete the import process.
    a. Upload a snapshot from a FreeForm app to your environment. See Uploading Archives to the Service in *Administering Migration for Oracle Enterprise Performance Management Cloud* for detailed instructions.
    b. Import the snapshot. See "Importing a Backup to Create a Clone of Another Environment" in Importing Artifacts and Application from a Snapshot in *Administering Migration for Oracle Enterprise Performance Management Cloud* for detailed instructions.
12. Create application users in identity domain and assign predefined roles as needed. See Manage Users and Roles in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.
13. Set up access permissions as needed. See Setting Up Access Permissions.

Managing Application Ownership

The Service Administrator who creates the application is auto-assigned the Application Owner function. When a Service Administrator with the Application Owner function is deleted or when the user's predefined role assignment is downgraded (for example, from Service Administrator to Power User), the Application Owner function is reassigned to the next available Service Administrator user (in alphabetical order) for the application, and then role of the current Application Owner user is deleted or changed to sync up with Access Control.

If there are no available Service Administrators, then the Service Administrator user is not deleted or the role is not changed. Oracle will contact you with next steps to resolve the issue.

Any Service Administrator can use **System Settings** to take over or reassign the Application Owner role to any available Service Administrator.

See What Application and System Settings Can I Specify?.

Frequently Asked Questions (FAQ) About FreeForm

Related Links

General Questions:

- What is the difference between Essbase and an EPM Cloud FreeForm app?
- When will current Essbase 21c versions be adopted in EPM Cloud?

- What is the impact of having EPM Cloud + Essbase 21c? How will they work together? Data movement, reporting, Smart View connections, and so on.
- What is the recommendation for customers for their Essbase cubes when migrating from On-premise EPM to EPM Cloud?
- What are the default hardware settings that will limit processing?
- What EPM license do I need to have to get multi-cube FreeForm?
- Where can I find helpful resources for frequently asked questions about administrative tasks in EPM Cloud?

FreeForm App Details Questions:

- Are all FreeForm apps considered custom applications?
- Does multi-cube FreeForm app mean that I can have 12 cubes, each with only 2 or 3 dimensions?
- What kind of Administration Interface is available in FreeForm apps?
- Does Cube Refresh refresh all cubes at once?
- What about size thresholds?
- Will slow changing attributes be supported in FreeForm apps?
- Is there direct access to FreeForm cubes through Smart View and Oracle Analytics Cloud and Data Visualization?
- Are hybrid cubes supported with FreeForm apps?
- Do FreeForm apps offer a repository of multiple applications?
- Is load of metadata through Data Management available with FreeForm apps?
- Do we lose any Planning Custom App Type related functionality in FreeForm apps?
- Can you use Groovy if you select FreeForm apps?
- Can you remove a dimension after adding it?
- Can you use Planning Migration files to create an application similar to the Essbase OTL or the Migration zip files?
- Can you import multiple outline or Migration files?
- Is FreeForm only available with the Enterprise version of EPM (vs. what was previously called PBCS)?
- How do you upload dimensions (and hierarchies) from a legacy Oracle Hyperion Planning application into EPM using FreeForm?
- Can we refresh the database from Smart View?
- Are partitions enabled?
- How can we configure the security filters?
- Is the Task Manager feature similar to Financial Consolidation and Close where it can be integrated, or is it just basic tasks? Can you please comment on this feature?
- How do you manage dimensions for FreeForm?
- Can I take a backup of a FreeForm app and then migrate it to a Financial Consolidation and Close or Planning Modules Cloud application?
- What about the business rules script, will they be migrated properly? And the configurable consolidation rule with regard to the Financial Consolidation and Close application?

- Does migration of an OTL work with all versions of Essbase?
- Is there no restriction for Data Integration in FreeForm that needs Scenario/Time period dimensions?
- Can we import data if the application is already created?
- Could you provide information about how the EPM Integration Agent feature is being used in Data Exchange?
- Is there the capability to create custom dimensions and business rules?

Oracle Essbase Migration Technical Questions:

- What tuning capabilities, if any, will be made available for FreeForm apps?
- Is partitioning supported? Transparent, Linked, Replicated?
- What about the MaxL Language for Admins, Automation, Shell integration?
- Are the MDX functions going to be supported?
- Aside from Data Integration/ETL how else can we recreate rules files in FreeForm apps? This could be cumbersome in Data Integration.
- For large scale ASO databases, what is the preferred data load mechanism in FreeForm apps?
- Any plans to create a CDF to Groovy migration tool?
- What Essbase artifacts can I migrate into FreeForm apps and how?
- What objects are skipped when Essbase outline files or Migration files are imported into FreeForm apps?
- How do Essbase features map to FreeForm apps in EPM Cloud?
- How do I deal with Active and Passive Essbase cubes in FreeForm when migrating these cubes from Essbase?

General Questions:

What is the difference between Essbase and an EPM Cloud FreeForm app?

Essbase as a solution can be bought as an on-premise solution or for deployment through Oracle Cloud Infrastructure (OCI). It is deployed as an OCI solution with the customer having full deployment control of the cubes. FreeForm apps are a SaaS solution offered by Oracle Enterprise Performance Management Cloud that allows Essbase cubes to be imported into EPM Cloud and deployed standalone. FreeForm apps have a hybrid architecture with a relational component beyond the Essbase cubes. FreeForm apps can have up to 12 cubes and up to 29 total dimensions across all cubes. (26 custom dimensions plus three account, period, and entity dimension types.)

When will current Essbase 21c versions be adopted in EPM Cloud?

Adoption of the latest Essbase versions into EPM Cloud is based on the impact on our customers – functional and performance. We are currently evaluating 21c for EPM Cloud adoption, and FreeForm apps will be one of the early adopters.

What is the impact of having EPM Cloud + Essbase 21c? How will they work together? Data movement, reporting, Smart View connections, and so on.

Essbase 21c on OCI is an IaaS deployment. FreeForm is a SaaS deployment. Data will need to be extracted from either environment or moved to the other. Oracle Smart View for Office shared connections can connect within the same platform so you can connect multiple

FreeForm apps using one shared EPM connection. Essbase 21c on OCI would be a private connection when FreeForm is a shared connection in Smart View. Any data exchange between FreeForm apps and Essbase 21c on OCI or on-premise will be an export-import of data.

What is the recommendation for customers for their Essbase cubes when migrating from On-premise EPM to EPM Cloud?

On-premise customers migrating their on-premise EPM instances to EPM Cloud should migrate their Essbase reporting or planning cube instances into FreeForm apps in EPM Cloud. This will ensure consistent access to EPM business processes all within SaaS EPM Cloud. This also avoids data latency and data movement across environments. Also, it will ensure there is one place to maintain access rights and security. Customers can leverage all the latest features in theEPM Cloud platform and future enhancements on an ongoing basis.

What are the default hardware settings that will limit processing?

We operate based on named user licensing and the Cloud hosting policy guides performance expectations. EPM Cloud uses a standardized Cloud hardware configuration, and customers can make change requests using the Service Request process.

What EPM license do I need to have to get multi-cube FreeForm?

FreeForm apps are available with EPM Enterprise Cloud.

Where can I find helpful resources for frequently asked questions about administrative tasks in EPM Cloud?

See Frequently Asked Questions in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

FreeForm App Details Questions:

Are all FreeForm apps considered custom applications?

Yes. All FreeForm apps are fully custom with no out-of-box content. You have full flexibility in dimensions and members. You can create cubes of any dimension combination as long as the total cubes are less than 26 custom dimensions. You can map Account, Period, or Entity dimensions if they are part of the cube dimensionality to the default dimension types. We automatically map these dimensions if they exist in the source outline or Migration file.

Does multi-cube FreeForm app mean that I can have 12 cubes, each with only 2 or 3 dimensions?

Up to 12 cubes are allowed in a FreeForm app, in any combination of BSO and ASO cubes. Each cube can have the exact needed dimensions without requiring any required dimensions or members.

What kind of Administration Interface is available in FreeForm apps?

Essbase admin console features to the extent relevant in SaaS are available through Calculation Manager or the dimension editor in the FreeForm app. Since it is a web-based application there are other admin interfaces in EPM web applications that allow administrators to manage EPM applications.

Does Cube Refresh refresh all cubes at once?

FreeForm deployments are cubes contained within one FreeForm app. You can have up to 12 cubes. Refresh is by application across all cubes. There is no concept of cube-specific refresh in EPM Cloud.

What about size thresholds?

There are no file size limits in the new EPM Enterprise Cloud where FreeForm is available. Dimension size and data volumes are handled on a case-by-case basis. There is a standard well established support protocol for upscaling the thresholds.

Will slow changing attributes be supported in FreeForm apps?

Some customers have asked for this, but the demand for it is not high enough. We will investigate it based on customer connect feedback in Cloud Customer Connect.

Is there direct access to FreeForm cubes through Smart View and Oracle Analytics Cloud and Data Visualization?

Smart View connections to FreeForm apps goes through the standard Smart View EPM Cloud interface. FreeForm apps can be accessed from Oracle Analytics Cloud and using Data Visualization using the direct connect capability in Oracle Analytics Cloud.

Are hybrid cubes supported with FreeForm apps?

FreeForm apps support creation of Hybrid BSO cubes and ASO cubes.

Do FreeForm apps offer a repository of multiple applications?

Repository is by application but can be connected into from another application. Data across applications can be blended in web and Smart View.

Is load of metadata through Data Management available with FreeForm apps?

Yes. Data Management metadata load is supported where there is at least one BSO cube. We will be supporting FreeForm apps with only ASO cubes through Data Management in the near term. The import of metadata using the Outline Load Utility (OLU) is an option.

Do we lose any Planning Custom App Type related functionality in FreeForm apps?

Common platform functionality which does not depend on required Planning dimensions is all available. Open dimensional approvals is on the roadmap.

Can you use Groovy if you select FreeForm apps?

Yes. All platform features are available with FreeForm apps. All business rules, including Groovy rules, are supported with FreeForm apps.

Can you remove a dimension after adding it?

You can unselect the dimension from the cube and the next cube refresh will remove the dimension association from that cube.

Can you use Planning Migration files to create an application similar to the Essbase OTL or the Migration zip files?

FreeForm apps are meant for Essbase outline (OTL)/Migration imports or as build-from-scratch applications. On-premise Planning applications map to custom application types in EPM Enterprise Cloud.

Can you import multiple outline or Migration files?

FreeForm only accepts one outline or Migration file to create an application. Additional cubes can be created in the interface and associated to this outline. We don't import multiple outline or Migration files into a single application.

Is FreeForm only available with the Enterprise version of EPM (vs. what was previously called PBCS)?

FreeForm apps are available with EPM Enterprise Cloud.

How do you upload dimensions (and hierarchies) from a legacy Oracle Hyperion Planning application into EPM using FreeForm?

Options are to load using an Essbase outline file (OTL), dimension extracts from source and import, manage using web interface/SmartView extension, or manually create.

Can we refresh the database from Smart View?

Yes, a refresh from Smart View is possible. Right-click on **Dimensions**, and then **Refresh Databases**.

Are partitions enabled?

No. Partitions are not required within FreeForm since it is a SaaS deployment in EPM Cloud where alternate solutions are available. Data maps and Smart Push can be used to feed data across cubes. Partitioning requires both Disk I/O and network access, both of which are not allowed by Oracle SaaS Cloud. In Cloud, the disk and network have been virtualized. With a mixture of ASO and BSO cubes in the same FreeForm app and with use of Hybrid BSO, data maps, Smart Push, and Groovy, partitioning is not necessary.

How can we configure the security filters?

Security is controlled through access controls, users are granted access to an application and then objects from there. In addition, cell-level security exists in FreeForm.

Is the Task Manager feature similar to Financial Consolidation and Close where it can be integrated, or is it just basic tasks? Can you please comment on this feature?

FreeForm uses task lists. The Task Manager feature is now available in Planning applications, but not in FreeForm.

How do you manage dimensions for FreeForm?

Dimension management can be completed in a number of ways:

- Manually in the solution using the web interface and Smart View
- Integrated from a source solution using Data Management and import dimension jobs
- Using an integration tool such as Oracle Fusion Cloud EPM Infrastructure (OCI)
- From Oracle Data Relationship Management (DRM) with integration
- Using Oracle Enterprise Data Management Cloud, part of the EPM Enterprise solution, which provides a full enterprise data governance and control capability

Can I take a backup of a FreeForm app and then migrate it to a Financial Consolidation and Close or Planning Modules Cloud application?

No, FreeForm is a different business process and, as such, is not portable to other business processes.

What about the business rules script, will they be migrated properly? And the configurable consolidation rule with regard to the Financial Consolidation and Close application?

No, FreeForm is a different business process and, as such, is not portable to other business processes.

Does migration of an OTL work with all versions of Essbase?

You can use an outline (OTL) file or snapshot from an on-premises release 11.1.2.4.xxx (or later) single cube Essbase application to create a FreeForm app. For more information, see FreeForm App Sources.

Is there no restriction for Data Integration in FreeForm that needs Scenario/Time period dimensions?

To use Data Integration in FreeForm you must have a minimum of Account, Period, and Scenario dimension types defined. For more information, see Loading Data to a Free Form Application in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

Can we import data if the application is already created?

Data can be imported using the data import capability. For more information, see Importing Data. Migration-based data imports can only be used during application creation.

Could you provide information about how the EPM Integration Agent feature is being used in Data Exchange?

The EPM Integration Agent is a fully unified solution for extracting and transforming data and metadata from your on-premises or any system such as a third-party cloud, and delivering it to EPM Cloud.

You can connect to and load data from on-premises data sources using custom SQL queries or pre-packaged queries to import data from sources such as EBS and the PeopleSoft General Ledger.

For more information, see EPM Cloud to On-premises Connectivity Using the EPM Integration Agent in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

Is there the capability to create custom dimensions and business rules?

Yes. FreeForm, by design, is a completely open business process and, as such, dimensions and business rules would all be custom built.

Essbase Migration Technical Questions:

What tuning capabilities, if any, will be made available for FreeForm apps?

Since FreeForm apps are a SaaS application, tuning is managed by Oracle. To the extent needed, database properties are available within the Calculation Manager interface.

Is partitioning supported? Transparent, Linked, Replicated?

No. Partitions are not required within FreeForm since it is a SaaS deployment in EPM Cloud where alternate solutions are available. Data maps and Smart Push can be used to feed data across cubes. Partitioning requires both Disk I/O and network access, both of which are not allowed by Oracle SaaS Cloud. In Cloud, the disk and network have been virtualized. With a mixture of ASO and BSO cubes in the same FreeForm app and with use of Hybrid BSO, data maps, Smart Push, and Groovy, partitioning is not necessary.

What about the MaxL Language for Admins, Automation, Shell integration?

Direct scripting is prevented for SaaS Cloud security reasons. Given it is a SaaS platform, we don't allow open ended scripts. There are other alternatives available in EPM Cloud for MaxL. EPM Automate, EPM Agent, and Groovy are good alternatives.

Are the MDX functions going to be supported?

MDX is already supported to some extent for ASO in FreeForm apps. MDX scripting for custom calc and allocation is exposed since 20.05 through Groovy scripts.

Aside from Data Integration/ETL how else can we recreate rules files in FreeForm apps? This could be cumbersome in Data Integration.

Currently we support import of Essbase format data into FreeForm apps. See Defining a Data Integration and Loading Data to a FreeForm Application in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

For large scale ASO databases, what is the preferred data load mechanism in FreeForm apps?

Use the Essbase file format or use Data Integration.

Any plans to create a CDF to Groovy migration tool?

There are no plans for CDF migration to Groovy. Given Java security considerations, CDF migration cannot be done. Groovy is a good modern alternative.

What Essbase artifacts can I migrate into FreeForm apps and how?

Essbase Artifact	FreeForm Import File?	Notes
Application and cube metadata	Yes	Cubes and outline metadata are imported.
Calculation scripts	Yes	Imported into Calculation Manager and used as business rules
Data	Yes	Level zero import from source or using Migration
Disk volumes	NA	
Drill through definitions	No	Managed through Data Integration
Excel workbooks and files	NA	Remapped with EPM Connections in Smart View
Filters	No	Metadata security is maintained in relational schema.
Linked Reporting Objects (LROs)	No	Attachments and comments are managed separately in EPM.
Location aliases	NA	

Essbase Artifact	FreeForm Import File?	Notes
Log files	No	
Outlines and formulas	Yes	
Partitions	No	Alternatives with data maps and Smart Push across cubes are available.
Report scripts	No	Other export alternatives are available.
Rule files, text files, .csv files	Yes	
Scenarios	NA	
Substitution variables	Yes	
Users	No	Managed separately in IDM in EPM SaaS Cloud
User roles	No	Managed separately in IDM in EPM SaaS Cloud

What objects are skipped when Essbase outline files or Migration files are imported into FreeForm apps?

Essbase Artifact	Reason object was skipped during import
Custom Defined Function (CDF)	CDFs are possible, but given Java security consideration, cannot be done now. Groovy is a good alternative.
Custom Defined Macro (CDM)	CDMs in Essbase are old artifacts. Using a Calculation Manager design time prompt on a rule is a better implementation of that same concept.
Location Alias	Location Alias is a short-cut name that allows you to reference a physical Essbase database through its server name, app name, database name, username, and password. FreeForm apps are self-contained within a single application with multiple databases. Data maps and Smart Push can be used within the application for this.
Report Scripts	Alternatives for data import and export are available in EPM Cloud through Groovy and EPM Automate.
Load Rules	Data Integration offers the ability to pick up load rules, which can be done separately.

How do Essbase features map to FreeForm apps in EPM Cloud?

Native Essbase 11g Feature	FreeForm Apps in EPM Cloud
Active/Passive Cubes	Can be supported using implementation setup
Calc Scripts	Business Rules
Custom Defined Functions/Macros	Groovy
Database Versioning	Automation with Migration snapshots. Additional environments.
Oracle Essbase Administration Services Outline View/Edit	Dimension Editor
Essbase Cell-Level Security	Cell-Level Security in EPM Cloud applications
Linked Reporting Objects	Cell Comments, Doc Attachments, Supporting Details
Load Rules	Data Integration Rules, Groovy, Dimension/Data Import, OLU Export, Level 0 (automated) exports

Native Essbase 11g Feature	FreeForm Apps in EPM Cloud
MaxL scripts	EPM Automate, EPM Agent, and Groovy
Partitioning	Cross-Cube Data Maps/Smart Push with Groovy is an effective alternative
Report Scripts	Export rules, other query/automation solutions in EPM Cloud
Text List	Smart List across all dimensions

How do I deal with Active and Passive Essbase cubes in FreeForm when migrating these cubes from Essbase?

- Active cubes are cubes that are actively in use and are used frequently by many users.
- Passive cubes are cubes not actively used and are used only once in a while. Passive cubes are often version snapshots of data to represent a view of the data at a point in time for comparative decisions, depending on looking back or as archives of data for audit. Passive cubes are only needed occasionally and only by few users.
- Typically there are fewer active cubes than passive cubes. Since active cubes are used frequently and have a larger number of uses, customers should migrate all active cubes into active applications. Cubes which have common dimensions should, as much as possible, be collated into a single application to optimize the number of applications deployed.

For passive cubes, the recommended practice is as follows:

1. Request additional environments (typically 2-5) as part of the licensing process. The exact number of additional environments will depend on the number of users, and the number of passive cubes.
2. Create an import and export automation routine using EPM Automate that will Import Oracle Essbase outline files/Migration files for each of the passive cubes into a spare environment, and then export them out as a FreeForm Migration file. Export FreeForm Migration files for each of the passive cube applications into a local/remote cloud location each month to ensure that there is always a current version of the Migration file for each passive cube application readily available for import. Ensure that the file names of the application containing these passive cubes are intuitive and representative of the functional nature of the cubes they represent.
3. If there's a need to work with or view data in a passive cube, the Migration file for that application can be imported into the spare EPM Cloud instance. Since the Migration file version is current, it will import into the current version of EPM Cloud for FreeForm.

3

Getting Started

Related Topics

- About the Home Page
  After you create the business process, the Home page is the launch point for accessing your business process tasks.

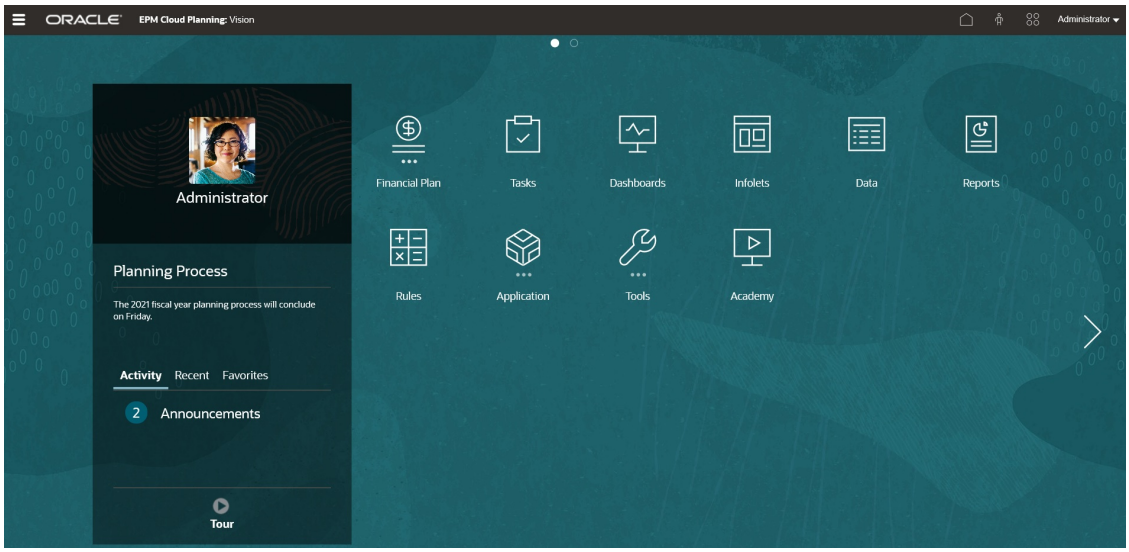
About the Home Page

After you create the business process, the Home page is the launch point for accessing your business process tasks.

The interface provides an intuitive user experience and an overview for quick access to commonly used functions. For example, users can access their tasks, work with data, approve budgets, view reports, and control settings. Service Administrators can manage and customize the business process, create forms, dashboards, and infolets, import and export data and metadata, schedule jobs, define valid intersections, make announcements, and create cross-environment connections.

All newly created or re-created Oracle Enterprise Performance Management Cloud services, business processes, and applications use the Redwood Experience as the default theme. You can change the general look and feel of your EPM Cloud environment on the **Appearance** page. On the **Appearance** page, you can enable the Redwood Experience or choose a classic theme. Each theme provides different background colors, icon styles, and so on. You can also add a branding logo and background images to the Home page. To change the general look and feel of your environment, see Customizing Your Display.

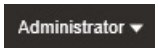
Example Home Page Displaying the Redwood Experience



Global Header

The global header is the area that stretches across the top of the user interface. It contains navigation icons as well as access to accessibility settings and the **Settings and Actions** menu. You can also switch between navigation flows from the global header.

Parts of the global header from left to right:

Global Header Part	Description
	The Navigator icon opens the **Navigator** menu, which serves as a sitemap of the business process and displays links to all of the business process pages to which you have access. **Note:** Some of the links in the **Navigator** menu are available only if you're accessing the business process from the desktop.
	Click the Oracle logo to return to the Home page while working elsewhere in the business process. You can display a custom logo instead of the Oracle logo by selecting a **Logo Image** on the **Appearance** page.
	The name of the current business process. You can hide the business process name by selecting **No** for the **Display Business Process Name** option on the **Appearance** page.
	The Home icon refreshes the Home page or returns you to the Home page while working elsewhere in the business process.
	Click the Accessibility Settings icon to enable accessibility features.
	If you belong to multiple groups or if a navigation flow is assigned to a role, click the navigation flow icon to switch navigation flows at runtime.
	Click your user name to access the **Settings and Actions** menu.

Work Area

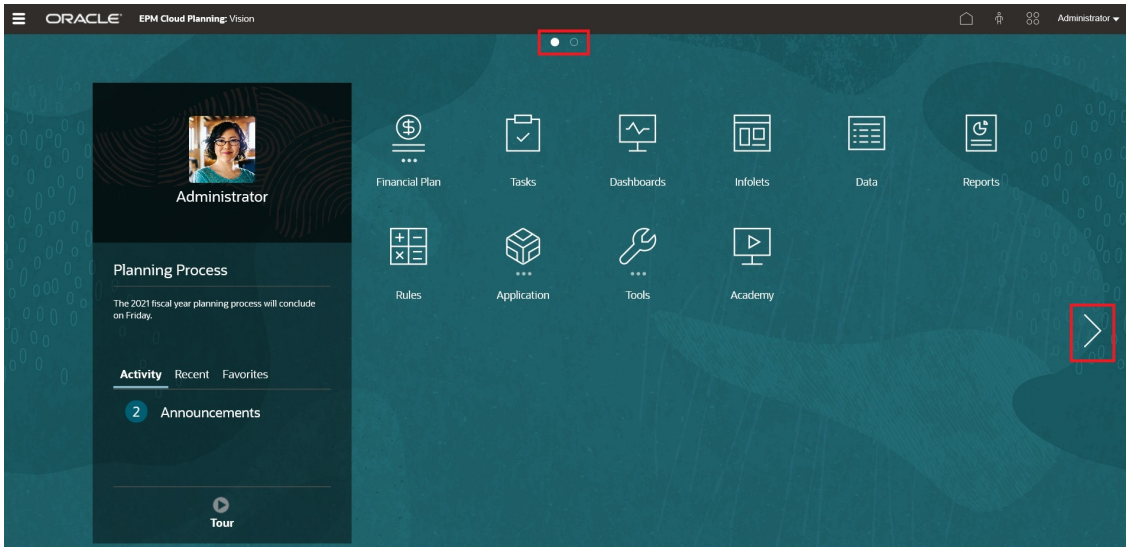
The work area on the Home page displays either a theme-based background image or a custom background image. The icons that display in the work area link you to each functional area of the business process to which you have access; for example, **Financials**, **Dashboards**, and **Data**. The three dots above an icon label denotes that the icon opens a grouping of sub-icons, called a cluster.

The **Academy** icon links you to a variety of resources about using the business process.

To replace the theme-based background image in the work area with a custom background image, select **Background Image** on the **Appearance** page.

Infolet Navigation

If your business process uses infolets to show high-level, essential information, you can explore them by clicking the dots that appear beneath the global header. Arrows are also available on the sides of the Home page to help you navigate easily between the Home page and infolet dashboard pages.



Announcements Panel

The Announcements Panel displays your user name and profile picture (if set), any system announcements entered by the Service Administrator, and helps you track your activity.

- You can set your profile picture to display at the top of the Announcements panel of the Home page. To set your profile picture, click **Tools**, and then **User Preferences**.

  For more information, see Setting Your Profile Picture in *Working with FreeForm*.
- **Activity**: Summarizes system announcements (the most recent announcement, sorted by effective date, appears at the top) and lists your open tasks. Click **Announcements** to flip the panel and view announcements. Click **Tasks Due Today** to flip the panel and view your tasks.
- **Recent**: Displays a list of links to recently visited forms and dashboards (up to 15 items). Clicking a link will launch the item in a secondary window. Clicking the star next to a link will tag it as a favorite.
- **Favorites**: Displays a list of links to forms or dashboards that were tagged as favorites, and prevents them from being overwritten. Click a link in **Favorites** to launch the item in a

secondary window. To add items to **Favorites**, click **Recent** to view your recent user activity, and then click the star to the right of the item.

-  : Click the **Tour** icon to launch a video about key features in the business process.

Videos

Your Goal	Watch This Video
Explore the default Redwood Experience theme.	 Overview: Announcing EPM Cloud's new Redwood Theme
Learn how to customize the interface to streamline workflow.	 Overview: Customizing Workflow in Enterprise Performance Management Cloud

4

Setting Up Access Permissions

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

Related Topics

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

About User and Role Management

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

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

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

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

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

Application Artifacts That Can Be Assigned Permissions

You can assign permissions to:

- Dimensions, including user-defined dimensions

> **Note:**
>
> You can assign permissions to members by selecting the dimension property **Apply Security**. If you omit or clear the **Apply Security** setting, all users can access the dimension's members.
>
> See Editing Dimension Properties.

- Navigation flows

- Launch privileges to rules
- Rule folders
- Forms
- Dashboards
- Infolets
- Reports, Books, and Bursting Definitions
- Form folders
- Dashboard folders
- Infolet folders
- Reports and Documents folders
- Task lists
- Groovy templates

Types of Access Permissions

Access permissions include **Read**, **Write**, **None**, and **Display**.

For descriptions of these options, see Accessing Ancestor Members in Ad Hoc Grids.

You can also set who can launch which rules.

- **Launch**: Allow launch privileges

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

- **No Launch**: Disallow launch privileges

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

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

You can use groups to provide access permissions to your application artifacts such as forms, rules, and dashboards. Oracle Enterprise Performance Management Cloud recognizes three types of groups:

- **Predefined**: These groups are automatically created for each predefined role. All users are assigned to a predefined group based on their predefined role (for example, Power User).
- **EPM**: These are the groups that you create in **Access Control** in **Tools**.
- **IDCS**: These are the groups that you create in the Oracle Cloud Identity Console. You can view them in **Access Control** and assign them to application roles and EPM groups.

For more information, see Manage Groups in *Administering Access Control for Oracle Enterprise Performance Management Cloud* .

Inheriting Permissions

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

Table 4-1 Options for Inheriting Access Permissions

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

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

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

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

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

Managing Permissions to Artifacts

Related Topics

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

About Assigning Permissions to Artifacts, Rules, and Folders

Service Administrators can assign permissions to artifacts (forms, dashboards, infolets, reports, books, and bursting definitions), rules, and folders.

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

Principles:

- **Artifacts (Forms, Dashboards, Infolets, Reports, Books, and Bursting Definitions):**
  - Users and Power Users can view or enter data only into forms to which they have permissions (and can work only with members to which they have permissions).
  - Service Administrators and Power Users can design artifacts.
  - Power Users can access artifacts they created or to which a Service Administrator assigned them permissions.
  - Service Administrators have Write permissions to all dimension members and to all artifacts.
- **Rules**:
  - Users and Power Users can see and launch only rules to which they are assigned Launch permission.
  - Rulesets inherit launch permissions from the rules included in the ruleset.
- **Folders:**
  - Users who are assigned permissions to a folder can access the items in that folder, unless they are assigned more specific permissions. Likewise, Users have Launch permission to the Calculation Manager rules in folders to which they are assigned permissions, unless they are assigned more specific permissions.
  - When you assign permissions to a folder, all folders under it inherit that permission.
  - If you assign specific permissions (for example, None or Write) to a folder, that permission takes precedence over its parent folder's permissions. For example, if a user has Write permission to Folder1 that contains Folder2 to which the user has None permission, the user can open Folder1, but doesn't see Folder2.
  - If you assign specific permissions (for example, Launch) to a Calculation Manager folder, that permission takes precedence over its parent folder's permissions. For example, if a user has Launch permission to RulesFolder1 that contains RulesFolder2 to which the user has No Launch permission, the user can open RulesFolder1, but doesn't see RulesFolder2.
  - If a user has None permission to a folder called Folder1 that contains a form called Form1 to which the user has Write permission, the user can see Folder1 and Form1.

- If a user has No Launch permission to a Calculation Manager folder called RulesFolder1 that contains a rule called Rule1 to which the user has Launch permission, the user can see RulesFolder1 and Rule1.

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

Adding, Changing, and Removing Permissions to Artifacts and Folders

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

To assign permissions to artifacts and folders:

1. Navigate to the artifact (Data, Dashboards, Infolets, Reports, or Documents) listing page.
2. Click the Actions icon next to the artifact or folder, and then click **Assign Permission**.

> **Note:**
>
> You can assign permissions to only one artifact or folder at a time.

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

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

To assign permissions to rules and rules folders:

1. Click **Rules**.
2. Select the rule or rule folder, click  , and then click **Assign Permission**.

> **Note:**
>
> You can assign permissions to only one rule or folder at a time.

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

Reporting on Access Permissions

You can view current access permissions and print reports.

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

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

Table 4-2 Access Report Types

Report Type	Description	Options
Assigned Access	Summarizes access permissions that Service Administrators assign	Specify whether access permissions are 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**: Shows access permissions inherited by users in a group.

Table 4-2 (Cont.) Access Report Types

Report Type	Description	Options
Effective Access	Summarizes access permission assignments as the application evaluates them (for example, by member selection relation, such as children, or group membership). This is useful if there are conflicts in access permissions. **Note:** Effective Access reports can't be generated for groups.	Describe the origin of the effective access permissions by selecting **Show effective access origin**. For example, a user named JSomebody may be assigned Write access permissions to Entity1 and belong to a group named Sales that is assigned Read access permissions to Entity1. This setting would show that JSomebody has Write access permissions to Entity1 because individual assigned access permissions supersede access permissions inherited by group membership.

7. Click **Create Report**.

Working with Access Permissions Reports

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

Retrieving Changed Information from Access Control

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

To synchronize information after modifying group properties:

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

5

Managing Applications

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

Related Topics

- Application Overview
  Start here to learn about managing applications, dimensions, and dimension members.
- Refreshing the Application
  Refresh the application whenever you make changes to the application structure.
- Managing Cubes
- Managing Dimensions
  Start here to learn how to manage dimensions and dimension members.
- Setting User Preferences
- Administering Variables
- Viewing Activity Reports
  Monitor user activities with activity reports.
- Importing and Exporting Data and Metadata
  Move data and metadata in and out of the application using FreeForm.
- Validating the Essbase Outline
  Pre-validate the Essbase outline to ensure that your application is compatible with a Hybrid Essbase version.
- Creating and Refreshing Application Databases
  Things you need to know before you refresh the application database.
- Adding an Aggregate Storage Outline to an Application
  Understand what aggregate storage is and how to use it in an application.
- Removing an Application
  Proceed with caution when removing an application.
- Setting the Daily Maintenance Process Start Time
  Understand how to determine and set the most convenient time to start the hour-long daily maintenance process.
- Uploading and Downloading Files Using the Inbox/Outbox Explorer
  Use the Inbox/Outbox Explorer to upload files to the server or download them to your computer.

Application Overview

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

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

To create an application, see Creating a FreeForm App.

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

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

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

Refreshing the Application

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

Changes you've made to the application are not reflected to users performing data entry tasks until you refresh the application. For example, if you modify properties of a dimension member, or add a member, or change access permissions, users will not see the updates you've made until you refresh the application.

To refresh the application:

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

> **Note:**
>
> The application is automatically refreshed during an application update. Information about the refresh is captured in a job called Content Update. To view the progress and details of the content update process, use the Jobs console, which you can access by clicking **Jobs** in the **Application** cluster in the Home page. See Viewing the Status of Jobs in this guide for information on how to view jobs, and Setting Content Update Start Time in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators* for information about setting the Content Update start time.

Managing Cubes

Related Topics

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

Viewing and Managing Cubes

To view and manage cubes:

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

   For cube details, see Adding Cubes.

> **Note:**
>
> You can rename a cube only during application creation. After a cube is added, you can't rename or delete it.

Adding Cubes

For FreeForm, you can add up to 12 total cubes with any combination of aggregate storage and block storage. See the chart below for a comparison of cubes by application type. See Understanding FreeForm.

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

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

Cube Comparison by Application Type

Planning Application Types

Cube Comparisons

EPM SKUs	App type	Open Cubes		Module Cubes		TOTAL
		BSO	ASO	BSO	ASO	
EPM Cloud Enterprise	Modules Based	3	4	5	2	14
	Custom	6	6	0	0	12
	Free Form*	12*	12*	0	0	12
	Sales Planning	0	1	3	2	6
	SWP	0	1	2	1	4
	Predictive Cash Forecasting (PCF)	0	1	2	1	4
EPM Cloud Standard	Modules	1	1	5	2	9
Legacy	EPBCS	3	4	5	2	14
	PBCS	3	4	0	0	7

** Total of 12 Hybrid BSO or ASO*

Clearing Cubes

Related Topics

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

About Clearing Cubes

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

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

Creating Clear Cube Jobs

To create a Clear Cube job:

1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then **Clear Cube**.
3. On the **Clear Cube** page, click **Create**, complete the job name, select the cube, then select a clear option:
   - **Clear All**: For both block storage and aggregate storage cubes, delete all data within a cube that is associated with the entities you select:
     - **Supporting Details**
     - **Comments**
     - **Attachments**
     - **Essbase Data**
   - For block storage cubes only:
     - **Upper-level blocks**: Delete data in upper-level blocks
     - **Non-input blocks**: Delete data in non-input blocks
     - **Dynamic blocks**: Delete data in dynamic blocks
     - **Empty blocks**: Delete blocks with #MISSING values
   - For aggregate storage cubes only:
     - **All Aggregations**: Delete all aggregate views
     - **Partial Data (Basic Mode)**: Clear the data in the specified region. You can select the point of view using the member selector  .

        - **Supporting Details**
        - **Comments**
        - **Attachments**
        - **Essbase Data Logical**: In which the input cells in the specified region are written to a new data slice with negative, compensating values that result in a value of zero for the cells you want to clear. The process for logically clearing data completes in a length of time that is proportional to the size of the data being cleared. Because compensating cells are created, this option increases the size of the database.
        - **Essbase Data Physical**: In which the input cells in the specified region are physically removed from the aggregate storage database. The process for physically clearing data completes in a length of time that is proportional to the size of the input data, not the size of the data being cleared. Therefore, you might typically use this method only when you need to remove large slices of data.
    - **Partial Data (Advanced Mode)**: Clear Essbase data only by using an MDX expression in the text box provided.
        - **Essbase Data Logical**: In which the input cells in the specified region are written to a new data slice with negative, compensating values that result in a value of zero for the cells you want to clear. The process for logically clearing data completes in a length of time that is proportional to the size of the data being cleared. Because compensating cells are created, this option increases the size of the database.
        - **Essbase Data Physical**: In which the input cells in the specified region are physically removed from the aggregate storage database. The process for physically clearing data completes in a length of time that is proportional to the size of the input data, not the size of the data being cleared. Therefore, you might typically use this method only when you need to remove large slices of data.

4. Click **Save and Close**.
5. To view the job on the **Clear Cube** listing page, click **Refresh**.

To start the Clear Cube job right away or to delete a job, see Starting and Deleting Clear Cube Jobs.

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

> **Tip:**
>
> To streamline the job creation process for subsequent **Clear Cube** jobs, you can duplicate an existing job using the **Save As** option, then update it. See Duplicating Jobs.

Starting and Deleting Clear Cube Jobs

To start a Clear Cube job right away or to delete a job:

1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then **Clear Cube**.

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

Scheduling Clear Cube Jobs

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

Improving Cube Performance

Jobs You Can Schedule to Improve Cube Performance

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

- **Restructure Cube**: Performs a full restructure of a block storage cube to eliminate or reduce fragmentation. This will also remove empty blocks. Running this job won't push any changes from the business process to Essbase.
- **Compact Outline**: Compacts the outline file of an aggregate storage cube. Compaction helps keep the outline file at an optimal size. Compacting the outline doesn't clear the data. Running this job won't push any changes from the business process to Essbase.
- **Merge Data Slices**: Merges incremental data slices of an aggregate storage cube. Fewer slices improve a cube's performance. You can merge all incremental data slices into the main database slice or merge all incremental data slices into a single data slice without changing the main database slice. You can optionally remove cells that have a value of zero.
- **Optimize Aggregation**: Generates optimized views based on collected query tracking information in an aggregate storage cube. This job has two actions:
  - **Enable Query Tracking**: Records a meaningful set of queries, and then uses the recorded query data to select the most appropriate set of aggregate views to materialize for that database. You must enable query tracking and allow it sufficient time to collect data before executing the aggregation process based on query data.

    Once enabled, query tracking continues until query tracking is disabled, the cube is reloaded or refreshed, the outline is compacted, the application is shut down, or additional aggregate views are materialized for the database. Query tracking does not resume automatically after these actions occur.

> **Note:**
>
> The following actions clear the stored query tracking information from the aggregate storage cube, but they do not disable query tracking:
>
> * Merging incremental data slices
> * Clearing data
> * Clearing existing aggregations
> * Clearing partial data
> * Loading data into the aggregate storage cube

– **Execute Aggregation Process**: Calculates aggregations for aggregate storage databases that contain data and to which you are granted Calculation permission. To perform an aggregation, you use system recommended views. The selection of views and aggregation processes are combined into one, non-configurable operation performed by the server. You can optionally specify the maximum disk space for the resulting files, base the view selection on user querying patterns, and include rollup hierarchies in the view selection. Select one or more of the following options:

  * **Based on query data?**: Aggregate the views the server selects, based on collected user querying patterns. This option is only available if query tracking is turned on.
  * **Include rollup option?**: Include secondary hierarchies (with default level usage) in the view selection process.
  * **Include growth size option?**: Aggregate the views the server selects, until the maximum growth of the aggregated database exceeds limits you specify. Enter the size (in ratio) beyond which the server should stop the aggregation.

You can start the jobs right away or schedule them to run later.

> **Note:**
>
> Users can stay logged in during these actions.

Related Topics

- Scheduling Jobs
- Optimize BSO Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide*
- Optimizing Aggregate Storage Option Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide*

Managing Dimensions

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

Dimensions categorize data values.

Dimensions on the **Dimensions** page are listed in order of precedence. The order of dimensions is critical for the structure and performance of an application. The order of your

dimensions determines how your data calculations will perform. **Evaluation Order** enables you to specify which data type prevails when a data intersection has conflicting data types.

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

To manage dimensions:

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

Tutorials

Tutorials provide instructions with sequenced videos and documentation to help you learn a topic.

Your Goal	Learn How
Get an introduction to Dimensions and Members and learn how to manage them in Oracle Enterprise Performance Management Cloud business processes.	Managing Dimensions in Cloud EPM Business Processes

Filtering the Dimension View by Cube

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

To filter the dimension view by cube:

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

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

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

- Order of precedence
- Dimension density

> **Note:**
>
> The **Dense** column is hidden for applications that contain only ASO cubes. If an application contains a BSO cube or a BSO and an ASO cube, the **Dense** column displays for both cubes.

- Evaluation order

For information about editing dimensions, see Editing Dimensions in the Simplified Dimension Editor.

Creating Dimensions

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

To create a dimension:

1. Click **Application**, click **Overview**, and then click **Dimensions**.
2. Click **Create**.
3. On the **Create Dimension** page, enter or select values for the properties that are displayed.

   For FreeForm applications, you can select the dimension type. Dimension types can include:

   - Account
   - Entity
   - Period
   - Custom

   Notes about dimension type:

   - A dimension type is not listed if you've already created it (for Account, Entity, Period) or if you've exceeded the maximum number of dimensions (for Custom).
   - You can't edit the dimension type after you've created the dimension.

   For descriptions of the properties, see Editing Member Properties.

   For information about editing dimensions, see Editing Dimensions in the Simplified Dimension Editor.

Setting User Preferences

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

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

To learn more about application defaults and system settings, see Managing Application and System Settings.

Administering Variables

Related Topics

- Working with Substitution Variables
- Working with User Variables

Working with Substitution Variables

Related Topics

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

About Substitution Variables

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

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

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

Creating and Assigning Values to Substitution Variables

To create a substitution variable and assign values to it:

1. From the Home page, click **Tools**, and then click **Variables**.
2. Select the **Substitution Variables** tab.
3. Click  .

   A new variable is added to the bottom of the list.
4. Scroll to the bottom of the list, click the down arrow next to the **Cube** name for the newly added variable, and then select the cube.
5. For **Name**, enter the name of the substitution variable (up to 80 characters).

6. For **Value**, enter a value for the substitution variable (up to 255 characters).

> **Note:**
>
> - You can specify a range of values when defining substitution variables; for example, `FY16:FY18`.
>
> You can define a substitution variable range for base members and parent members. For base members, you can use a single colon (`:`) or a double colon (`::`) as delimiters; for example, `SunEnT 110::111`. However for parent members, you can only use a single colon (`:`).
> - If the substitution variable will be used in data export job definitions for block storage cubes and the member name is numeric only, the member name must be placed in double quotation marks in the **Value** field; for example, `"1100"` or `"000"`. The double quotation marks aren't necessary for alphanumeric member names; for example, `a1110`. However if this variable will be selected in data export job definitions for aggregate storage cubes, you can't place the member name in double quotation marks in the **Value** field. Therefore if you need to use the same member name in substitution variables for data export jobs defined for both types of cubes, you must define two different substitution variables; one with quotes and one without

7. Click **Save**.

Deleting Substitution Variables

To delete substitution variables:

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

Working with User Variables

Related Topics

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

About User Variables

User variables act as filters in forms, enabling users 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, users must first select values in preferences for the variable before opening forms. Afterward, users can change the variable on

the form only if it's a dynamic user variable. Otherwise, they must continue to set the variable in preferences. For example, if you create a user variable called Division, users must select a division before working in the form.

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

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

Creating User Variables

To create user variables:

1. From the Home page, click **Tools**, and then click **Variables**.
2. Select the **User Variables** tab.
3. Click  .

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

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

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

   Afterward, users can select members for the user variable. See *Working with FreeForm*.

Managing User Variables

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

The typical sequence of steps:

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

   See Creating User Variables.
3. When designing the form, associate the user variable with the form.

   See Selecting User Variables as Members.
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 FreeForm*.

Setting Limits for User Variables on Forms

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

To set up user variable limits:

1. During form design, select **Enable Dynamic User Variables** to allow dynamic user variables in the form.

   See Setting Form Precision and Other Options.
2. Create a user variable.

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

   To select members by function, click  and select from the available options.
4. In the form designer, assign the user variable to the page axis.

   See Defining Form Page and Point of View.

Deleting User Variables

You can delete user variables if they are not associated with a form. If you attempt to delete a user variable that is used in a form, you'll see an error message saying it cannot be deleted. The message will also provide the names of the forms where the user variable is used.

To delete user variables not associated with forms:

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

Viewing Activity Reports

Monitor user activities with activity reports.

The Activity Report enables Service Administrators to understand application usage. It also helps streamline application design by identifying user requests, calculation scripts, forms, reports, and so on. Two versions of the report; an HTML version and a JSON version, are available.

Oracle Enterprise Performance Management Cloud retains Activity Reports for the last 60 days only. Oracle recommends that you download and create backup copies on a local computer to analyze them for historical trends.

The Activity Report is automatically generated in these situations:

- Every day during daily maintenance of the service
- Each time you submit a Provide Feedback submission
- Every time you execute the resetService EPM Automate command to restart an environment.

For details about the information that is provided in activity reports, see Activity Report Contents in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

To view activity reports:

1. Click **Application**, and then click **Overview**.
2. Click **Activity Reports**.
3. To view the report details, click **View** next to the name of the report that you want to view.

Importing and Exporting Data and Metadata

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

Related Topics

- Importing Metadata
  You can import metadata from a flat file in a comma-delimited, tab-delimited, or other format.
- Exporting Metadata
  You can export metadata to a flat file in a .csv (comma-delimited) or .txt (tab-delimited or other delimiter character) format.
- Importing and Exporting Data

Importing Metadata

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

These artifacts are supported in the imports:

- Dimensions
- Smart Lists
- Exchange rate tables

To import metadata, perform these tasks:

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

> **Note:**
>
> - You cannot rename members of attribute dimensions during a metadata import. Renaming of members will be ignored.
> - You cannot delete attribute dimensions while importing metadata.

Videos

Your Goal	Watch This Video
Learn how to import metadata.	 Importing Metadata in Cloud EPM

Creating the Metadata Import File

Related Topics

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

About the Metadata Import File

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

1. A header record, the first line in the file:
   - Lists the dimension and any member properties used by subsequent metadata records; the header record and subsequent records don't need to include all

properties; properties that are not included are inherited from the corresponding parent's default property value

- Is case sensitive
- Can list properties in any order, as long as the subsequent metadata records are in the same order
- Can use comma or tab delimiters. Other delimiter characters can be used if they are supported and the same delimiter character is used throughout the file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.

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

> **Note:**
>
> Only characters that are included in the UTF-8 character set are supported in import or export files. Characters that aren't included in the UTF-8 character set are considered invalid. For dimension name restrictions, see Naming Restrictions for Dimensions, Members, and Aliases.
>
> In order to delete UDAs and Attributes in metadata load files, replace the UDAs and attributes with `<none>` instead of leaving the property value blank.

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

Example Entity Dimension Import File

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

```
Entity, Parent, Data Storage
e1, Entity,
```

```
e2, ,
e1, e2, Shared
```

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

```
Entity
e1
e2
    e1(Shared)
```

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

Other Supported Delimiter Characters

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

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

- brackets ([ ])
- at sign (@)
- period (.)

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

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

Import and Export Metadata Delimiter Exceptions

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

Table 5-1 Import and Export Metadata Delimiter Exceptions

Delimiter Character	Reason for Exception
double quotation mark ("")	Creates an empty file
plus (+) minus (-) forward slash (/) percent sign (%)	Causes an error if the metadata import file contains consolidation properties that use these characters
angle brackets (< >)	Causes an error if a property uses the value <none>

> **Note:**
>
> Any character that conflicts with a character in a member name will cause an error.

Import and Export Data Delimiter Exceptions

Table 5-2 Import and Export Data Delimiter Exceptions

Delimiter Character	Reason for Exception
apostrophe (')	Causes an error if used in a cube name
parentheses ( )	Causes an error if used in a data import file
double quotation mark ("")	Creates an empty file
hyphen-minus (-)	Causes an error if used in a data import file

Loading the Metadata Import File

To load the metadata import file:

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

2. Click **Application**, and then click **Overview**.
3. Click **Dimensions**, and then click **Import**.
4. On the **Import Metadata** page, click **Create**.
5. Select the location of the metadata import file or files:
   - **Local**: Loads the import file or files from a location on your computer. For **Import File**, click **Browse** to select the import file on your computer for the artifact you're importing.
   - **Inbox**: Loads the import file or files from the server. Enter the name of the file in **Import File**. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.
6. For **File Type**, select **Comma delimited**, **Tab delimited**, or **Other**.

   If **Other** is selected, enter the delimiter character that is used in the import file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.
7. Select **Clear Members** to delete members not explicitly specified in the load file before performing the import.

   > **Note:**
   >
   > - Any member not specified is deleted from the application outline after importing the dimension unless it's an ancestor of a member that was specified, or it's a base member of a shared member that was specified.
   > - If **Clear Members** isn't selected, the import will only add or update existing members. Oracle recommends loading the metadata file without selecting **Clear Members** to make sure the file loads successfully. Then, select **Clear Members** and execute the import again
   > - **Clear Members** is unavailable in certain dimensions where members must exist and are critical to the application.
   > - Members that can't be deleted in the dimension editor for any reason (for example, because they're used in a form or in a validation rule) won't be deleted.
8. **Optional:** If the location selected is **Inbox**, click **Save as Job** to save the import operation as a job, which you can schedule to run right away or at a later time. You can't select this option if **Local** is selected. See Managing Jobs.

> **Note:**
>
> - Saving an import operation as a job is useful to batch a load sequence; for example, import metadata, then import data, then run rules when the data load is complete.
> - You can run up to five import jobs at one time.
> - If you want to review the data that was rejected during the metadata import job, you can specify an **Error File** on the **Save as Job** dialog. This file will provide information about the metadata records that were not imported for each dimension. If an error zip file is specified, a separate error file is created for each dimension, then the error files are zipped together, and the zip file is stored in the Outbox where you can download the file using Inbox/Outbox Explorer. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.

> **Tip:**
>
> To streamline the job creation process for subsequent **Import Metadata** jobs, you can duplicate an existing job using the **Save As** option, then update it. See Duplicating Jobs.

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

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

If the import fails, a **Failed** status will display in the **Last Validate/Import** column. Click **Failed** to view the detailed status. If the import is successful, a **Completed** status will display in the **Last Validate/Import** column. To view details about the successful import, click **Completed**, and then in the **Show** drop-down, select **All**.

When performing metadata loads as a part of cloning environments or importing snapshots using Migration or EPM Automate, Oracle Enterprise Performance Management Cloud uses the multi-pass solution. This solution automatically initiates another load of the same metadata input file if the previous metadata load resulted in rejected records.

Exporting Metadata

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

These artifacts are supported in the exports:

- Dimensions
- Smart Lists
- Exchange rates

The application creates an export file for each artifact (.txt or .csv, depending on the file type), and all export files are consolidated into one zip file. You must extract the .csv or .txt files from

the zip file if you want to use the files as import files (for example, when importing into another application).

> **Note:**
>
> Only characters that are included in the UTF-8 character set are supported in export or import files. Characters that aren't included in the UTF-8 character set are considered invalid. For dimension name restrictions, see Naming Restrictions for Dimensions, Members, and Aliases.

Videos

Your Goal	Watch This Video
Learn how to export metadata.	 Exporting Metadata in Cloud EPM

To export metadata to a flat file:

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

     Click **Save as Job** to save the export operation as a job, which you can schedule to run right away or at a later time. You can't select this option if **Local** is selected.

     > **Note:**
     >
     > – You can run up to five export jobs at one time.
     > – When scheduling export jobs, you can specify a unique output file name for each job.
     > – For more information about scheduling jobs, see Managing Jobs.

     > **Tip:**
     >
     > To streamline the job creation process for subsequent **Export Metadata** jobs, you can duplicate an existing job using the **Save As** option, then update it. See Duplicating Jobs.
5. Select the artifact or artifacts to export.
6. Select an option:
   - **Comma delimited**: Creates a comma-delimited .csv file for each artifact.

- **Tab delimited**: Creates a tab-delimited .txt file for each artifact.
- **Other**: Creates a .txt file for each artifact. Enter the delimiter character that you want to use in the export file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.

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

Importing and Exporting Data

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

Related Topics

- Importing Data
- Driver Member Casting Errors
- Exporting Data
- Viewing Data Import and Export Status

Importing Data

Before importing data, you must:

- Define or load metadata and refresh the database. See Importing Metadata.
- Prepare the data load file. Source data must be in a business process-specific data file format or in an Oracle Essbase data file format.
- Ensure the data load files do not contain any thousands separators, quotes, and so on, around data values. Up to one single . (decimal point) character is supported.
- Ensure the data import file type is .csv, .txt, or .zip.

You can create a data import file based on a template, or you can generate one based on the source system. To use a template, run the export data process in your business process and use the exported file as a template. See Exporting Data.

> **Note:**
>
> When importing data, no validation is performed on intersections to determine which are valid, and data is loaded to all intersections. To review data that has been loaded into invalid intersections, run the Invalid Intersection report before importing data to see and clear invalid intersections. See Working With Invalid Intersection Reports.

To import data:

1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then click **Import Data**.
3. Click **Create**.
4. Select the location of the data import file:
   - **Local**: Imports the data import file from a location on your computer.

- **Inbox**: Imports the data import file from the server. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.

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

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

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

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

     For **Delimiter**, specify the delimiter for the Essbase data; options are **Space** or **Tab**.

     > **Note:**
     >
     > **Stop In Case of Error** is selected by default. If this option is selected and any import errors occur during an Essbase data load, the import will quit and no data will be loaded. If this option is cleared, data will continue to load even if there are errors.

6. Enter or select the source file:
   - If you selected **Local**, click **Browse** to navigate to the source file.
   - If you selected **Inbox**, enter the name of the file in **Source File**.
7. **Optional:** Clear **Include Metadata** to improve data import performance and to prevent members from changing position in the outline during data import. Regardless of this option setting, the required metadata must already exist in the business process and in the Essbase outline before you can load the application data.
8. Click **Save as Job** to save the import operation as a job, which you can schedule to run later.

> **Note:**
>
> - **Save as Job** is available only when **Inbox** is selected.
> - Saving an import operation as a job is useful to batch a load sequence, for example, import metadata, then import data, then run rules when the data load is complete. See Managing Jobs.
> - You can run up to five import jobs at one time.
> - If you want to review the data that was rejected during the data import job, you can specify an **Error File** on the **Save as Job** dialog. This file will provide information about the data records that were not imported. If an error zip file is specified, the zip file is stored in the Outbox where you can download the file using Inbox/Outbox Explorer. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.

> **Tip:**
>
> To streamline the job creation process for subsequent **Import Data** jobs, you can duplicate an existing job using the **Save As** option, then update it. See Duplicating Jobs.

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

> **Note:**
>
> There is no import option in this business process to clear data before import. Therefore, the import will not overwrite existing data values with #MISSING values. However, you can run a clear cube job prior to import, or clear data with a script/rule or using the REST API:
>
> - You can run a Clear Cube job that enables you to clear specific business process data within input and reporting cubes. See Clearing Cubes.
> - You can write a business rule that uses the CLEARDATA Essbase command to clear specific data values. See CLEARDATA in *Technical Reference for Oracle Analytics Cloud - Essbase*.
> - You can use the Clear Data Slices REST API to clear business process and Essbase data for a specified region. See Clear Data Slices in *REST API for Enterprise Performance Management Cloud*.

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

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

Driver Member Casting Errors

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

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

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

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

```
Entity, aDate, aSl, aText, aCurr, aPercent, Point-of-View,Data Load Cube Name
e1, exampleBadDateValue, sl1, text11, 888, 0.99,
"FY15,Current,""BUVersion_1"",Local,u1,Jan",Plan1
```

Error messages displayed:

- After loading the data, the status window will read:

  ```
  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 Data

To export data to a file:

1. Click **Application**, and then click **Overview**.

2. Click **Actions**, and then click **Export Data**.
3. Click **Create** and make your **Export Data** selections:

Table 5-3 Data Export Options

Data Export Option	Description
Location	Select the target environment of the data export file: • **Local**: Saves the data export file to a location on your local computer. • **Outbox**: Saves the data export file to the server. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.
Cube	Select a cube from the drop-down list. **Note:** The driver/column dimension in the Export Data job definition needs to be dense.
File Type	Select from the following types: • **Comma delimited**: Creates a comma-delimited .csv file for each artifact. • **Tab delimited**: Creates a tab-delimited .txt file for each artifact. • **Other**: Creates a .txt file for each artifact. Enter the delimiter character that you want to use in the export file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.
Smart Lists	Specify **Export Labels** or **Export Names**.
Dense Dynamic Members (for BSO cubes only)	For a block storage (BSO) cube, select whether to **Include** or **Exclude** dense dynamic calculation members during export. Note the following: • Excluding dynamic calculation members from exported data improves performance during data exports. • Sparse dynamic members are not included in the export even if the **Include** option is selected.
Dynamic Members (for ASO cubes only)	For an aggregate storage (ASO) cube, select whether to **Include** or **Exclude** dynamic calculation members during export. **Note:** Excluding dynamic calculation members from exported data improves performance during data exports.

Table 5-3 (Cont.) Data Export Options

Data Export Option	Description
Decimals	Specify the number of decimal positions (0-16) to format the data after export, or select the default **None** to use the default precision formatting. For example, specifying a decimal value of **3** in the **Decimals** field will result in the exported data displaying three digits to the right of the decimal point, wherever applicable. **Note:** Selecting **None** formats the decimal number as specified in the default precision settings, whereas selecting **0** formats the data to display a whole number. For example, if the data value you are exporting is `123.45678`, selecting **0** will export the data as `123` and selecting **None** will export the data as `123.45678`.
Slice Definition	Select the slice of data to be exported. **Note:** If you're exporting data from an aggregate storage cube and you're exporting multiple members simultaneously, make sure that at least one of the members you select is a non-shared member. You can export shared members individually because it's equivalent to selecting a base member. However, selecting level 0 descendants of a parent member whose descendants are all shared members is equivalent to not selecting any members from a dimension and the export will not run. Therefore, you must ensure that one of the members selected is a non-shared member.

4. **Optional:** Click **Save as Job** to save the export operation as a job, which you can schedule to run immediately or at a later time.

Note:

- **Save as Job** is available only when **Outbox** is selected.
- You can run up to five export jobs at one time.
- When scheduling export jobs, you can specify a unique output file name for each job.
- For more information about scheduling jobs, see Managing Jobs.

> **Tip:**
>
> To streamline the job creation process for subsequent **Export Data** jobs, you can duplicate an existing job using the **Save As** option, then update it. See Duplicating Jobs.

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

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

Troubleshooting

For help with troubleshooting issues if you get an Oracle Essbase query limit error when exporting a large number of data cells from ASO cubes, see Handling Issues Related to Large Data Export from ASO Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Viewing Data Import and Export Status

To view the status of a data import and export:

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

The status page displays messages for **Errors**, **Warnings**, **Info**, and **All**. If there are errors and you specified an error file name while defining the job, you can download the generated error file using the **Error File** link.

Validating the Essbase Outline

Pre-validate the Essbase outline to ensure that your application is compatible with a Hybrid Essbase version.

As part of an effort to upgrade all environments to an Oracle Essbase version that supports Hybrid Block Storage Option (BSO) cubes, Oracle has added a process to pre-validate Essbase outlines to ensure a smooth transition when your environment is upgraded to a Hybrid Essbase version.

> **Note:**
>
> For information about Essbase versions, see About Essbase in EPM Cloud in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

To ensure that your application is compatible with a Hybrid Essbase version, we've implemented a new utility which verifies the member formulas in your environment and

provides a report so you can fix any issues. Use the following options under the **Essbase Outline Validation** menu option in the **Actions** menu on the application's **Overview** page:

- **Pre validate Outline**: Performs a validation of your application
- **Outline Pre-validation Report**: Displays a list of member formulas that need to be fixed to be compatible with Hybrid Essbase version

> **Note:**
>
> - The **Essbase Outline Validation** menu option only displays if your environment is on a non-Hybrid Essbase version. If your environment is on a Hybrid Essbase version, then the **Essbase Outline Validation** menu option is no longer available.
> - Another way to check whether your environment is on a Hybrid or non-Hybrid Essbase version is to view the value of **Essbase Version supports Hybrid Block Storage Option** in the **Activity Report**. If the value is **Yes**, it means that your environment is on a Hybrid Essbase version. If the value is **No**, then your environment is on a non-Hybrid Essbase version.

To validate the Essbase outline in your environment and to fix any errors found in member formulas:

1. From the Home page, click **Application**, and then click **Overview**.
2. Click **Actions**, select **Essbase Outline Validation**, then select from the following options:
   - To perform the validation, select **Pre validate Outline**.
   - To view and download a report of errors found during validation, select **Outline Pre-validation Report**.

After you have fixed the member formulas, you can use the **Pre validate Outline** option again to make sure that all issues are resolved.

Creating and Refreshing Application Databases

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

Related Topics

- Creating Application Databases
- Before Refreshing the Database
- Refreshing Application Databases
- Validating Metadata

Creating Application Databases

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

To create a FreeForm application, see Creating a FreeForm App.

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

Before Refreshing the Database

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

Before refreshing the database:

- Back up your application and export data from all databases.
- Decide if you want to enable all users or just the current Service Administrator to use the application in administration mode during the refresh
- Decide if you want to log off all users
- Terminate any active application requests

Refreshing Application Databases

The application database stores data for each cube in the application. You must refresh the database after changing the application structure, and modifying artifacts such as dimensions and members. You can start the Refresh Database process right away or schedule it to run later as a job.

To refresh the database:

1. See Before Refreshing the Database.
2. Click **Application**, and then click **Overview**.
3. Click **Actions**, and then **Refresh Database**.
4. On the **Refresh Database** page, click **Create**, and then select from the following options:

   Before Refresh Database

   - **Enable use of the application for**: Lets **All users** or **Administrators** (or the current logged in Service Administrator) access the application in administration mode during the refresh
   - **Log off all users**: Logs off all application users before starting the refresh
   - **Stop all active requests**: Terminates any active requests in the Essbase application before starting the refresh

   After Refresh Database

   - **Enable Use of the Application for**: Lets **All users** or **Administrators** use the application after the refresh
   - **Validate Metadata**: By default, this option is not selected. When this option is selected, the system scans for member identity (ID) mismatches between the business process and Oracle Essbase and then automatically resolves them after the refresh process. Selecting **Validate Metadata** might add additional time to the refresh process. For more information, see Validating Metadata.

     The **Validate Metadata** option is available for cubes with these storage options running on the Oracle Essbase version that supports Hybrid Block Storage Option (BSO) cubes: Hybrid BSO, ASO, and BSO. This option is not available for cubes running on the legacy Essbase version that does not support Hybrid BSO cubes.

5. To refresh the database now, click **Refresh Database**. To run the refresh later as a job, click **Save as Job**. All of the options you've selected on the **Refresh Database** page, including **Validate Metadata**, will persist to the saved job. See Managing Jobs.

> **Tip:**
>
> To streamline the job creation process for subsequent **Refresh Database** jobs, you can duplicate an existing job using the **Save As** option, then update it. See Duplicating Jobs.

Troubleshooting

For help with troubleshooting database refresh issues, see Troubleshooting Database Refresh Issues in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Validating Metadata

About Validate Metadata

Use the **Validate Metadata** setting (formerly called **Autocorrect Errors**) as part of the database refresh process to scan for and automatically resolve member identity (ID) mismatches between the business process and Oracle Essbase.

> **Caution:**
>
> Selecting **Validate Metadata** might add additional time to the refresh process.

In certain cases, the identity of a member changes in a way that causes it to be not fully recognized in the multi-dimensional database used by the application. When a member's identity is not fully correctly established within all parts of the system, it could sometimes cause hierarchy issues like incorrect member ordering, or it could miss children underneath that member, or miss some siblings, or miss certain properties, and so on. Selecting **Validate Metadata** detects such members and automatically resolves their identity, allowing these members as well as their hierarchies along with properties and siblings, to be fully and correctly recognizable and usable within all the parts of the system.

Viewing the Report Detailing Member Movement and Deletion

After making member ID corrections, members might be deleted or moved. A report file detailing any member movement (to a different parent) or deletion is generated and placed in the Outbox in a csv (comma-delimited) file format. Review the report to reaggregate some of the stored parent members where the child hierarchy may have changed.

For example:

Action	Member	Parent	Dimension	Cube	New Parent
Delete	Member1	Parent1	Scenario	Plan1	
Move	Member2	Parent2	Account	Plan2	Parent3

The system generates a CSV file to the Outbox folder, and the filename is automatically generated with the application name followed by `_autocorrect.csv`; for example, `vision_autocorrect.csv`. This file is overwritten on each run.

After running the **Refresh Database** job with the **Validate Metadata** setting, you can download the export file from the Outbox.

To download files:

1. Click **Application**, and then click **Jobs**.
2. Under **Recent Activity**, click the job.
3. At the top of **Job Details**, click the export file option to select a download location.

Adding an Aggregate Storage Outline to an Application

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

Related Topics

- About Aggregate Storage
- Aggregate Storage Outline Cube Characteristics
- Process for Adding an Aggregate Storage Database to an Application

About Aggregate Storage

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

Aggregate Storage Outline Cube Characteristics

- The application doesn't generate XREFs on aggregate storage databases. XREFs can only be generated on block storage databases.
- 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.

Troubleshooting

Performance of aggregate storage reporting cubes is governed by many factors, including the number of slices in the cube, the outline of the cube, and the type of dimensions in the cube. See Optimizing Aggregate Storage Option Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Aggregate Storage Outline Cube Characteristics

- The application doesn't generate XREFs on aggregate storage databases. XREFs can only be generated on block storage databases.
- Dynamic time series members are not applicable for the Period dimension in an aggregate storage application.
- Creating and refreshing security filters are not applicable for aggregate storage databases.

Process for Adding an Aggregate Storage Database to an Application

To add an aggregate storage database to an application:

1. Create an aggregate storage cube. Perform one task:
   - Create an aggregate storage cube during application creation.

     To create a FreeForm application, see Creating a FreeForm App.
   - Add a new cube using the cube editor.

     See Managing Cubes.
2. Add dimensions to the aggregate storage cube.

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

Removing an Application

Proceed with caution when removing an application.

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

To remove an application:

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

Setting the Daily Maintenance Process Start Time

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

A business process instance requires one hour every day to perform routine maintenance. Service Administrators can select (and change) the most convenient time to start the hour-long daily maintenance process.

For information about daily maintenance operations and scheduling the maintenance start time for an environment, see Managing Daily Maintenance in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

> **Note:**
>
> There is a system setting called **Export EPM Cloud Smart List textual data during daily maintenance for incremental data import** that, depending on if this setting is enabled, could impact the duration of the export. For more information and for recommendations about using this setting, see the following topics:
>
> - Exporting Smart List Textual Data During Daily Maintenance for Incremental Data Import in *Oracle Enterprise Performance Management Cloud Operations Guide*
> - What Application and System Settings Can I Specify?

To prevent automatic backup from failing due to certain jobs running during daily maintenance, EPM Cloud disallows certain jobs from starting while the daily maintenance process is running.

The following jobs are prevented from starting during daily maintenance:

- Import Data
- Import Metadata
- Export Data
- Export Metadata
- Refresh Database
- Clear Cube
- Restructure Cube
- Compact Outline
- Merge Data Slices
- Optimize Aggregation

These jobs are prevented from starting whether they are jobs that are scheduled or jobs that are started on an ad hoc (unscheduled) basis.

For unscheduled jobs, this means that if you attempt to start a job during the daily maintenance process using the **Run Now** option on the **Schedule Jobs** page or if you submit a job from the listing page of your job definitions, the job will be prevented from starting. If the system prevents an ad hoc job from starting, an error message will display.

For example, if you view the jobs listed on the **Clear Cube** page while daily maintenance is running and then you click **Actions** and then **Submit**, an error message will display and the job will not start. Similarly, if you create a **Clear Cube** job on the **Schedule Job** page during daily maintenance and select **Run Now**, the system will prevent the job from starting and you'll see an error message.

If the system prevents a job from starting, the reason will be stated in the Job Details. If you've enabled email notifications for the Job Console, you'll receive an email notification when a job does not start. If you have a job that is scheduled to start during the daily maintenance process, it is recommended that you reschedule your job to start outside of the daily maintenance window. See Scheduling Jobs.

Uploading and Downloading Files Using the Inbox/Outbox Explorer

Use the Inbox/Outbox Explorer to upload files to the server or download them to your computer.

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

To upload or download files:

1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then **Inbox/Outbox Explorer**.
3. **Optional:** Filter by name, refresh the listing, and upload files from the server.
4. To upload a file from the server, click **Upload**. In the **Upload File** dialog box, click **Browse**, select the file to upload, and then click **Upload File**.
5. To download a file from the server, select the desired file, then click •••, and then click **Download File**.
6. Click **Close** to close the **Inbox/Outbox Explorer** and return to the **Overview** tab.

> **Note:**
>
> There is an upload file size limit of 2GB. If your upload file exceeds the size limit, either select a smaller file or use EPM Automate to upload the larger file. For information about EPM Automate, see *Working with EPM Automate for Oracle Enterprise Performance Management Cloud*.

6

Using the Member Selector

Use the member selector to select members and use wildcards in searches. You can also select substitution variables, user variables, and user-defined attributes (UDAs).

Related Topics

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

Working with Members

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

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

> **Note:**
>
> Forms 2.0, Dashboards 2.0, and ad hoc grids use a different version of the member selector. See Using the Member Selector in *Administering and Working with Enterprise Profitability and Cost Management*.

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

If drop-down member selectors are defined for row dimensions on a form, you can select members directly from the list, enter data, and add rows to a form.

See Entering Data for Suppressed or Excluded Members in *Working with FreeForm*.

Making Selections

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

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

To make selections:

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

     > **Note:**
     >
     > The search isn't case-sensitive. You can search for a word, multiple words, or wildcard characters. See Using Wildcards in Searches.

   - To change display options such as viewing variables and attributes, showing alias names, showing member counts, sorting alphabetically, refreshing the member list, or clearing selections, click  next to **Search**, and then select from the list of display options.
   - To filter the members that are displayed in the member list, under  next to **Search**, select **Add Filter**, and then select from the list of filter options.

     > **Note:**
     >
     > Android users can click  to view and select filter options.

3. Make selections by clicking a member in the member list.

   To understand how related members are selected, see Member Relationships.

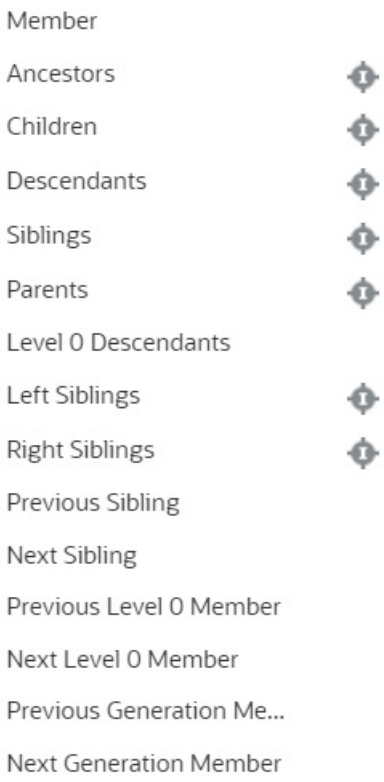
Note:

- Selected members display a check mark and are moved to the **Selections** pane, if applicable.
- To expand a parent member to see its child members, click the expansion icon to the right of the parent member name. Clicking the expansion icon will not select the parent member.
- To clear selections you have made, click  and then select **Clear Selection**.

  At times, you may find that a point of view and page axis member you want to select is suppressed. This occurs because a dimension selection in the point of view and page axis has invalidated other dimensions due to valid intersections that were applied. To resolve this issue, use the **Clear Selection** option to clear the point of view and page axis members you previously selected. Then you can use the point of view and page axis again to select members that were previously suppressed.
- To show all the members that are suppressed due to valid intersection rules, click  and then select **Show Invalid Members**. Invalid members are displayed but are unavailable for selection.

4. **Optional:** Perform these tasks:

   - To further refine which related members are selected in the **Selections** pane, click f_x to the right of the member to display the member relationships menu:



Click the relationship name to select the related members, excluding the selected member. Click the include icon to the right of the relationship name to select the related members, including the selected member.

For descriptions of the relationships, see Member Relationships.

- If substitution variables or attributes are defined, below the member selection area, click next to **Members**, and then select **Substitution Variables** or **Attributes** to select members for substitution variables or attributes. Members are displayed as children. Only members to which the user has read access are displayed in forms.
- To move or remove members in the **Selections** pane, click next to **Selections**.
- To highlight the location of a selected member within the dimension hierarchy, double-click the member name in the **Selections** pane.

> **Note:**
>
> To highlight the location of a selected member on a mobile device, tap the member name in the **Selections** pane, and then tap **Locate** under next to **Selections**.

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

Member Relationships

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

Table 6-1 Member Relationships

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

Table 6-1 (Cont.) Member Relationships

Relationship	Members Included
Left Siblings	The members that appear before the selected member with the same parent
Left Siblings (inc)	The selected member and its left siblings
Right Siblings	The members that appear after the selected member with the same parent
Right Siblings (inc)	The selected member and its right siblings
Previous Sibling	The member that appears immediately before the selected member with the same parent
Next Sibling	The member that appears immediately after the selected member with the same parent
Previous Level 0 Member	The previous level zero member that appears before the selected member Examples: PrevLvl0Mbr(Jan) returns BegBalance; PrevLvl0Mbr(Jul) returns Jun; PrevLvl0Mbr(BegBalance) returns an empty result
Next Level 0 Member	The next level zero member that appears after the selected member Examples: NextLvl0Mbr(Mar) returns Apr; NextLvl0Mbr(Dec) returns an empty result
Previous Generation Member	The member that appears immediately before the selected member within the same generation
Next Generation Member	The member that appears immediately after the selected member within the same generation
Relative	Returns a member relative to the specified member at the same generation with the specified offset. Examples: Relative("Jan", 4) returns May, which is January plus 4 months in a standard monthly Period dimension; Relative("Jan", -2) returns Nov, which is January minus 2 months in a standard monthly Period dimension.
Relative Range	Returns all of the members from the starting member to the offset member when going forward or from the offset member to the starting member when going backwards. Examples: RelativeRange("Jan", 4) returns Jan, Feb, Mar, Apr, May; RelativeRange("Jan", -2) returns Nov, Dec, Jan.

Note:

Member relationships work for any dimension, not just the Period dimension. We've used the Period dimension in our examples because it's easier to explain as time is linear.

Using Wildcards in Searches

You can use these wildcard characters to search for members.

Table 6-2 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].

Selecting Attribute Values as Members

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

Example: Numeric attribute

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

```
Size

Small

1

2

Medium

3

4

Large

5
```

```
6
```

Table 6-3 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 6-4 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.

Selecting Members for Forms

When selecting members for forms:

- To filter members from certain users, restrict their access permissions to members, and then refresh the plan.
- The order of members in the **Selected Members** list determines the order on forms. To change the order, select a member and click the Up or Down Arrow above the selected members list.

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

- In the **Layout** tab of the **Form Management** dialog box, you can open the **Member Selection** dialog box by clicking the member selection icon, or by right-clicking a row or column and selecting **Select Members**.
- To select different sets of members across the same dimension, see Creating Asymmetric Rows and Columns.
- For forms with multiple dimensions in a row or column, you can set member selection options for a dimension by selecting that dimension from the **Dimensions** drop-down list that is displayed in the Member Selection dialog box for multiple dimensions in a row or column.

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

  This feature is available for single members only, not for members selected with functions, such as Children (inc). For example, if you select Q/IChildren for Column A and select **Place Selection in Separate Columns**, the form layout isn't changed.
- All settings except Count are retained after the Member Selection dialog box is closed, and members in the Member Selection dialog box are displayed based on user-defined settings. Members displayed on the Layout tab don't inherit the display settings defined in Member Selection dialog box. Instead, they are displayed using the member name.
- To define different sets of members for a dimension, see Creating Asymmetric Rows and Columns.
- To set display, functionality, and printing options, see Setting Form Precision and Other Options.

Selecting Substitution Variables as Members

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

Note:

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

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

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

To specify substitution variables in forms:

1. Create the form (see Creating Forms).
2. In **Member Selection**, click the **Variables** tab, expand **Substitution Variables**, and then select substitution variables the same way you select members, to move substitution variables to and from **Selected Members**.

   When selected, a substitution variable is preceded by an ampersand (&). For example:

   ```
   &CurrentScenario
   ```
3. Click **OK**.

Selecting User Variables as Members

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

When you create forms with user variables, users must select values for the variable before opening forms. For example, if you create a user variable called Division, users 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 Forms).
2. On **Member Selection**, click the **Variables** tab, expand **User Variables**, and then select user variables the same way you select members, using the arrows to move user 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**.

Selecting UDAs as Members

You can select members for forms based on a common attribute, which you have defined as a user-defined attribute (UDA). Before you can associate the UDA with a form, you must create the UDA. See Working with UDAs.

When you create forms with UDAs, any members that are assigned to the UDA are dynamically added to the form. For example, if you create a UDA called New Products and assign this UDA to the new products in the Product dimension hierarchy, the form will automatically display the new products at runtime.

To select UDAs for forms:

1. Create the form (see Creating Forms).
2. On **Member Selection**, click the **Variables** tab, expand the **UDAs**, and then select the UDAs the same way you select members, using the arrows to move UDAs to and from the **Selected Members**.

   UDAs are displayed for the current dimension only. When selected, a UDA is preceded by `UDA`. For example:

   ```
   UDA(New Products)
   ```

3. Click **OK**.

7

Connecting Environments in EPM Cloud

Learn how to connect environments so users who have access across EPM Cloud can use a single access point with one login. Understand use cases, scenarios, and considerations, and learn how to connect to external web services and use direct URLs to integrate environments.

Related Topics

- About Connecting EPM Cloud Environments
- Considerations for Migrating EPM Cloud Connections
- Creating, Editing, and Deleting Connections to Other EPM Cloud Environments
- Connecting to External Web Services
- Specifying Advanced Options for External Connections
- Navigating Across EPM Cloud Environments
- Customizing Navigation Flows to Access Other EPM Cloud Environments
- Using Direct URLs to Integrate Connected Environments

About Connecting EPM Cloud Environments

Overview

Service Administrators can connect multiple EPM Cloud environments of the following types:

- Account Reconciliation
- Enterprise Profitability and Cost Management
- Financial Consolidation and Close
- FreeForm
- Narrative Reporting
- Planning
- Planning Modules
- Profitability and Cost Management
- Sales Planning
- Strategic Workforce Planning
- Tax Reporting

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

> **Note:**
>
> You can also connect directly Oracle Analytics Cloud Enterprise Edition or Professional Edition 5.6 to EPM Cloud Platform, provided you have both services. When you've configured the connection, you can visualize data from EPM Cloud business processes in Oracle Analytics Cloud. You no longer have to model EPM data in a metadata repository (RPD) file to create visualizations and dashboards in Oracle Analytics Cloud. For more information, see the Oracle Analytics Cloud documentation.
>
> Additionally, you can connect Oracle NetSuite Account Reconciliation and Oracle NetSuite Planning and Budgeting.

Which environments can I connect?

The source environment is the environment from which you're creating the connection. The target environment is the environment to which you're connecting from the source environment.

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

- Enterprise Profitability and Cost Management
- FreeForm
- Financial Consolidation and Close
- Oracle NetSuite Planning and Budgeting
- Planning
- Planning Modules
- Sales Planning
- Strategic Workforce Planning
- Tax Reporting

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

- Account Reconciliation
- Narrative Reporting
- Oracle NetSuite Account Reconciliation
- Profitability and Cost Management

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

- Toggle between the source environment and the target environment on the **Navigator** menu. See Navigating Across EPM Cloud Environments.
- Customize navigation flows in the source environment to access clusters, cards, and artifacts in other target environments from the Home page. See Customizing Navigation Flows to Access Other EPM Cloud Environments.
- Use direct URLs to seamlessly integrate connected environments. See Using Direct URLs to Integrate Connected Environments.

Considerations

- Only Service Administrators create cross-environment connections.

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

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

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

- Service Administrators cannot configure cross-environment connections using corporate SSO (identity provider) credentials.

  If your environments are configured for SSO, ensure that identity domain credentials are maintained for the Service Administrators who configure cross-environment connections. See Enabling Sign In With Identity Domain Credentials.

- Migrating cross-environment connections between test and production environments can cause issues in certain use case scenarios. For more information, see Considerations for Migrating EPM Cloud Connections.

- Vanity URLs are not supported in cross-environment connections.

Troubleshooting

For help with information on fixing common errors while connecting environments, see Handling Issues with Navigation Flows in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Videos

Your Goal	Watch This Video
Watch this video to learn how to customize an EPM Cloud workflow.	 Overview: Configure Navigation Flows to Integrate EPM Cloud Business Processes

Considerations for Migrating EPM Cloud Connections

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

In the following scenarios, assume you have environments for Financial Consolidation and Close and Planning.

Use Case Scenario 1: Test to Production

When migrating connections from test environments to production environments, ensure that connections that were defined in the test environment are changed to point to the corresponding production environments.

For example, a Service Administrator has defined a connection between the test environments of Planning and Financial Consolidation and Close. The Service Administrator then uses this connection to build a navigation flow in Planning that refers to a card in the Financial Consolidation and Close. The snapshot that the Service Administrator creates for migrating the Planning test environment will include connections and navigation flows, including the connection to Financial Consolidation and Close test environment.

On migrating the snapshot into the Planning production environment, Planning will have an undesirable connection to the Financial Consolidation and Close test environment. You must manually change undesirable connections to point to the corresponding production environment either before or after migrating the environment.

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

This scenario doesn't have any caveats.

Use Case Scenario 3: Production to Test

In this scenario, the Service Administrator might be trying to migrate a snapshot from a production environment into a test environment to resolve an issue. Because the connections created in the test environment still points to a production environment, it is important for the Service Administrator to modify connections so that they point to a test environment. Connections in test environments that point to a production environment may inadvertently tamper with the production environment.

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

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

To create, edit, duplicate, and delete connections:

1. Login to the source environment.
2. From the Home page, click **Tools**, and then click **Connections**.
3. Choose an action:
   - To add a connection:
     a. On the **Manage Connections** page, click **Create**.
     b. On the **Select Provider to Create Connection** page, select the target environment you want to add.
     c. Enter the target environment connection details:
        – Click **Change Provider** to select a different target environment.
        – In **Connection Name** enter name for this navigation link; for example, `Consolidation Application`.

- Enter an optional description for the link.
- In **URL**, enter the URL of the target environment instance; for example, `http(s)://your-target-host-url.com`. This is the URL that you normally use to sign in to the target environment instance.
- Use **Service Administrator** and **Password** to specify the credentials of a Service Administrator.

> **Note:**
>
> * These credentials are used only to define and validate the navigation link. When a user logs in, their own role and access will be applied to access the target environment.
> * Do not prefix user names with the domain name for connections to other EPM Cloud environments. However, the domain name is still needed for connections to other external Web services. See Connecting to External Web Services.

- The **Domain** field is automatically populated based on the URL you enter. If there is no domain in the URL, then the **Domain** field is left blank.

d. Click **Validate**.

e. If the validation is successful, click **Save and Close**.

- To edit connections:

a. On the **Manage Connections** page, click the name of a connection.

b. Edit connection details.

> **Note:**
>
> If you edit the URL to connect to a new service type, you could cause navigation flows to break. If you want to connect to a different service, Oracle recommends creating a new connection instead.

c. Click **Validate**.

d. If the validation is successful, click **Save and Close**.

- To duplicate a connection:

a. On the **Manage Connections** page, in the **Action** column next to the connection you want to duplicate, click •••.

b. Click **Duplicate**.

c. Enter a name for the duplicate connection, then click **OK**.

- To delete a connection:

a. On the **Manage Connections** page, in the **Action** column next to the connection you want to delete, click •••.

b. Click **Delete**.

When target environments are connected to an EPM Cloud source environment, they are listed in the **My Connections** pane on the **Navigator** menu of the source environment. The **My Connections** pane on the **Navigator** menu is where you can navigate across environments. See Navigating Across EPM Cloud Environments.

Troubleshooting

For troubleshooting assistance, see Handling Issues with Navigation Flows in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Connecting to External Web Services

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

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

> **Note:**
>
> The **Other Web Service Provider** connection type is only available for use with those business processes that allow Groovy Rules to be created. See the Oracle Enterprise Performance Management Cloud, Groovy Rules Java API Reference.

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

To create a connection to an external Web service:

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

> **Note:**
>
> The optional advanced options enable you to specify query or header parameters when defining an external connection. See Specifying Advanced Options for External Connections.

8. Enter **User** and **Password** login credentials for the connection, if required. In some cases, such as connecting to Oracle Fusion Cloud EPM services, the domain name may need to be prefixed to the user name; for example, `<Identity Domain>.<User Name>`.

   To understand basic authentication for EPM Cloud REST APIs and for instructions on finding your identity domain, see Basic Authentication - for Classic and OCI in *REST API for Enterprise Performance Management Cloud*.

9. Click **Save and Close**.

Specifying Advanced Options for External Connections

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

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

To specify advanced options for external connections:

1. Create an external connection or open an existing external connection.

   See Connecting to External Web Services.

2. Enter connection details, and then click **Show Advanced Options**.
3. Specify query details as follows:

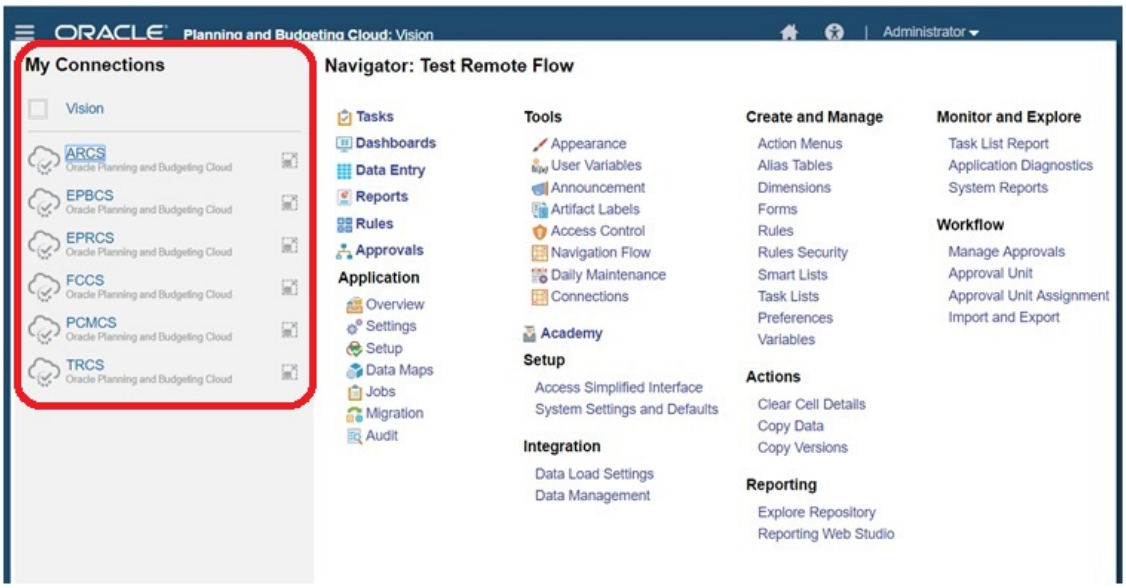
   - : Add query
   - : Delete query
   - **Type**: Select **Header** or **Parameter**.

     Header sets a default header that will be sent on every request made for this connection. **Parameter** sets a default query parameter that will be sent on every request made for this connection.
   - **Secure**: If selected, the value entered in the **Value** field will be encrypted. Clearing the **Secure** check box for a row will remove the value.

     An example header that one would secure is the Bearer Token for external Web services that supports Bearer Authentication, or the API Key query parameter for external Web services that supports API keys for authentication.
   - **Name**: Enter a name for the header or query parameter.
   - **Value**: Enter the value for the header or query parameter.

Navigating Across EPM Cloud Environments

Once a Service Administrator creates connections to other EPM Cloud environments, the connections are listed in the **My Connections** pane on the **Navigator** menu.



You can toggle between the environments from this location. You must have access to the other environments in order to open them. Artifacts are accessible based on the user's role.

To open another Oracle Enterprise Performance Management Cloud environment:

1. From the Home page, click **Navigator** .
2. If environments are connected and you have access to those environments, you'll see a list of connected environments in the **My Connections** pane. Click an environment to open it.

> **Note:**
>
> Click the icon to the right of the environment name to open the environment in a new window.

Customizing Navigation Flows to Access Other EPM Cloud Environments

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

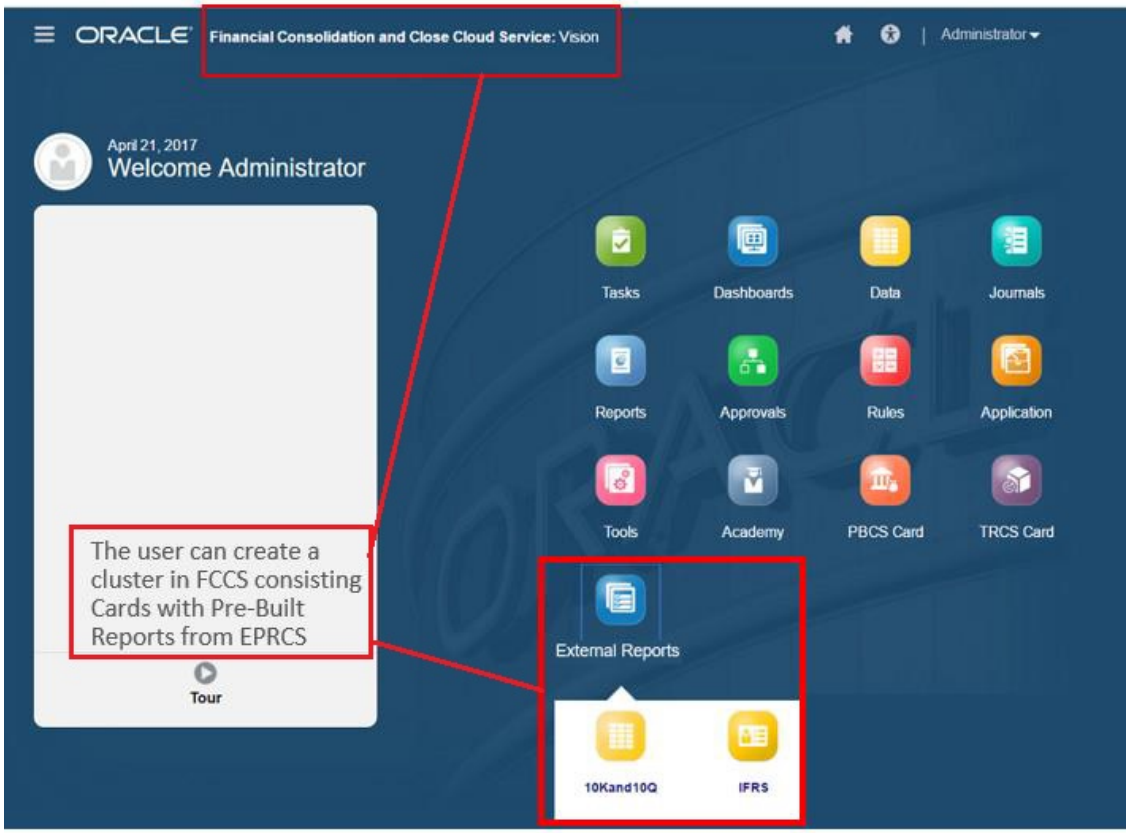
These two use cases describe in detail how to customize navigation flows to access other EPM Cloud environments:

- Grouping Cards from Other EPM Cloud Environments into Clusters
- Configuring Cards with Tabs from Multiple EPM Cloud Environments

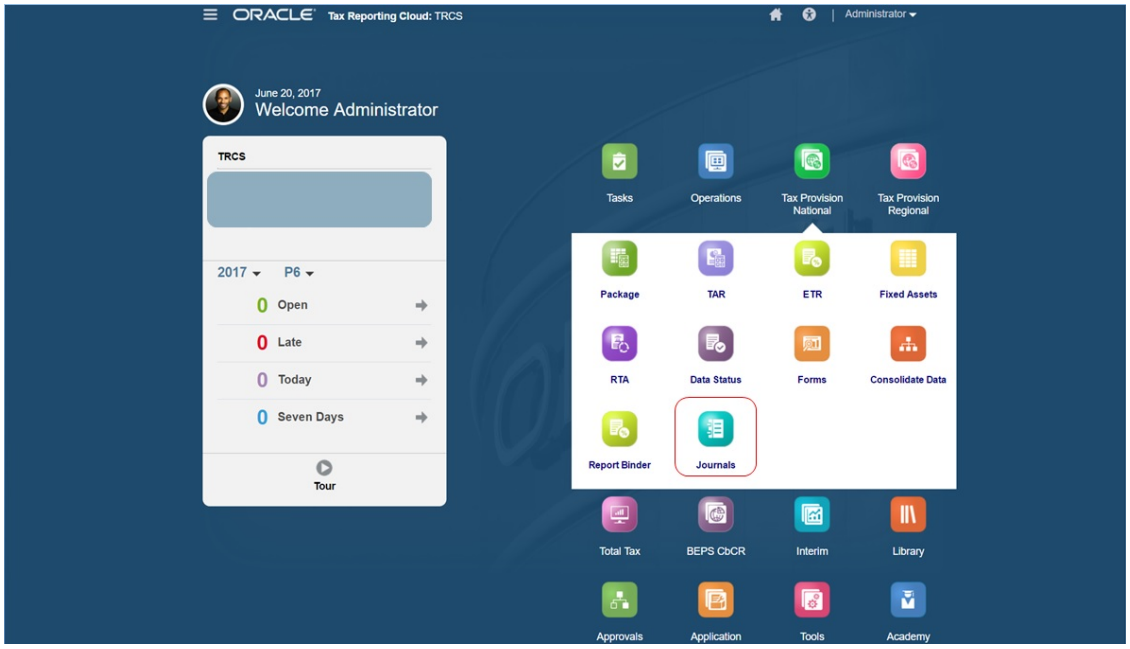
To learn more about designing navigation flows, see Designing Custom Navigation Flows.

Grouping Cards from Other EPM Cloud Environments into Clusters

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



Cards from multiple environments can also be included within the same cluster on a source environment. For example, a Tax Reporting user can launch a Journals icon from Financial Consolidation and Close without leaving Tax Reporting.



You create clusters and add cards to clusters by customizing navigation flows.

For general information about navigation flows, see Designing Custom Navigation Flows.

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

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

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

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

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

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

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

> **Note:**
>
> Editing is only possible if the navigation flow is inactive. If the navigation flow you want to edit is active, ensure you mark it **Inactive** before editing.

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

      Note the following:

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

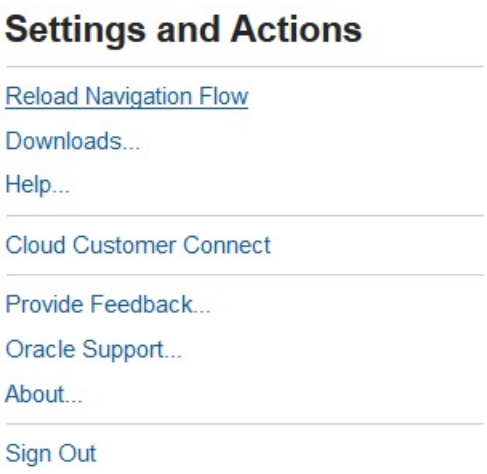
        See Specifying Artifact Labels.
3. Select the cards to include in the cluster using one of these options:
   - Navigate to the card you want to add to the cluster. If the card is within another environment, first select the environment under **My Connections**, and then navigate to the card in that environment. Assign the card to a cluster using one of these options:
     – To the right of the card that you want to move, in the **Order** column, click ›. Select the cluster, and then click **OK**.
     – Click the name of the card to view the card details, then for **Cluster** select a cluster for the card, and then click **OK**.
   - Navigate to the cluster in which you want to add the new card. If the cluster is within another environment, first select the environment under **My Connections**, and then navigate to the cluster in that environment. Right-click the cluster (or click +), click **Add Card In Cluster**, and then select an option:
     – Select **Add Existing Card**, to select an existing card or to add existing cards from another cluster to the selected cluster.
     – Select **Add Card**, and then enter card details to add a new card to the selected cluster.

> **Note:**
>
> You cannot add a card to a cluster if the card or the cluster is already referenced from another navigation flow.

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

4. Click **Save and Close**.

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

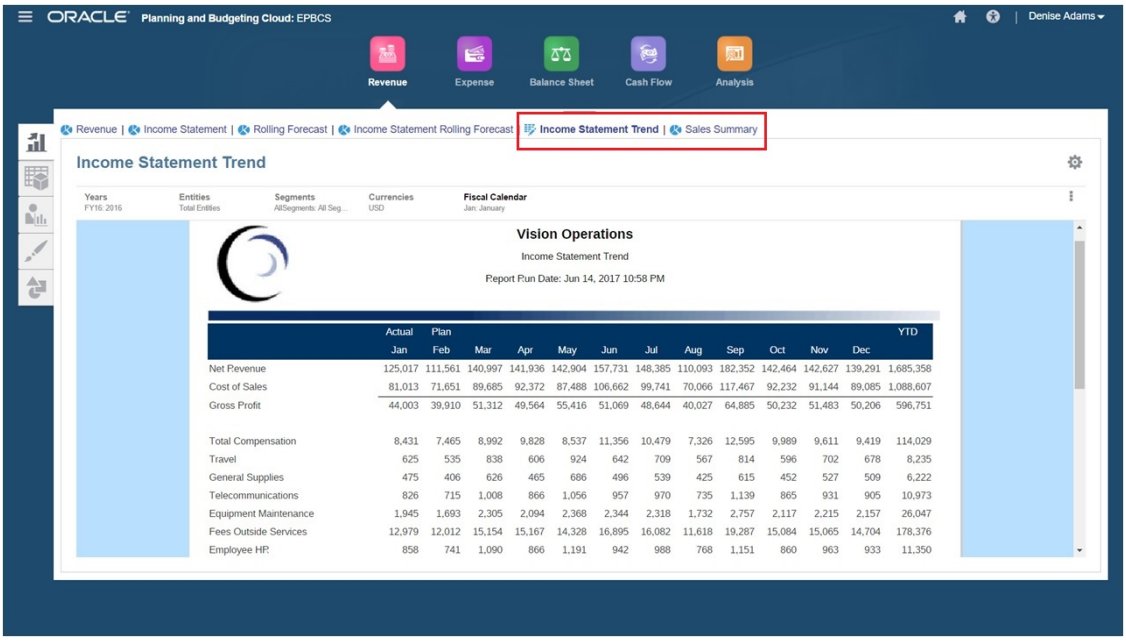


Troubleshooting

If you can't see your referenced artifacts after activating and reloading the navigation flow, see Handling Issues with Navigation Flows in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Configuring Cards with Tabs from Multiple EPM Cloud Environments

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



You create tabular cards by customizing navigation flows.

For general information about navigation flows, see Designing Custom Navigation Flows.

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

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

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

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

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

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

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

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

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

   Note the following:

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

     See Specifying Artifact Labels.

b. To add a new tabular card to the navigation flow, from the **Navigation Flow** page, right-click a card or cluster (or click), click **Add Card**, and then select details for the card:

   - **Name**: Enter a label for the card.
   - **Visible**: Select whether the card is visible to users on the Home page.
   - **Cluster**: If clusters exist, select a cluster for the card or select **None**.
   - **Icon**: Select the icon that will be displayed for the card you're creating. Choose from the available icons provided in the icon library.
   - **Content**: Select from the following options:
     - **Page Type**: Select a multiple page (tabular) format.
     - **Orientation**: Select **Vertical** or **Horizontal**.

3. Add tabs and sub tabs to the tabular card:

   a. To add an existing tab, right-click a tab, click **Add Existing Tab** (or click the **Add Existing Tab** button), and then select a tab from the Object Library.

   b. To add a new tab, right-click a tab, click **Add New Tab** (or click the **Add New Tab** button), and then edit tab details.

   c. Right-click a tab, click **Add New Sub Tab** or **Add Existing Sub Tab** (or click the **Add New Sub Tab** or **Add Existing Sub Tab** buttons), and then choose a sub tab from the Object Library or edit sub tab details.

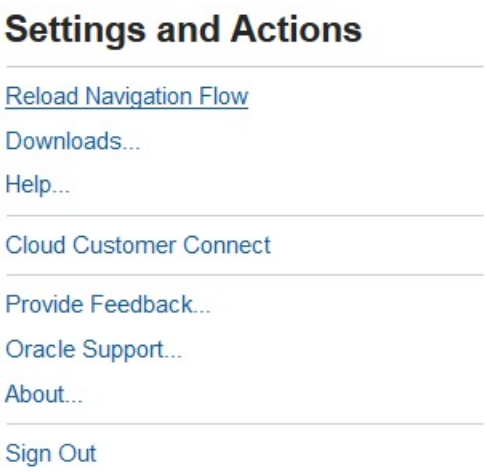
   d. For **Artifact**, click  to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include forms, dashboards, and reports. To select an artifact from another environment, select the target environment under **My Connections**, and then choose the artifact you want to add to your sub tab.

   e. Repeat adding tabs and sub tabs until the card is complete.

4. Click **Save and Close**.

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

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



If you can't see your referenced artifacts after reloading the navigation flow, see Handling Issues with Navigation Flows in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Using Direct URLs to Integrate Connected Environments

Other source systems like Oracle ERP Cloud can embed URLs to directly link to artifacts contained in cards, tabs, and sub-tabs within connected Oracle Enterprise Performance Management Cloud environments.

Other Cloud environments like Oracle ERP Cloud use direct URL links to open connected EPM Cloud content like forms, dashboards, infolets, and reports. To make the integration between EPM Cloud and other systems seamless, you can copy the unique URLs for the artifacts within a connected EPM Cloud business process. You can copy the unique URLs in one of two ways:

- Copy the individual URL for an artifact in the business process. See Copying Direct URLs for Artifacts.
- Export all of the URLs in the business process to a CSV file, then find and copy the unique URLs. See Exporting All URLs to a CSV File.

Videos

Your Goal	Watch This Video
Learn how to use direct links to embed EPM Cloud content in other systems such as ERP Cloud and NetSuite.	Overview: Embed Content Using Direct Links

Copying Direct URLs for Artifacts

Use the **Copy URL** option on the artifact listing page to copy the unique URL for an artifact (dashboards, forms, infolets, and reports) in your Oracle Enterprise Performance Management Cloud business process.

> **Note:**
>
> **Copy URL** is available to all users so that direct URLs to artifacts can be shared. However, only users with access to the targeted artifact can perform actions on it.

To copy the unique URL for an artifact:

1. From the Home page, open the listing page for the artifact.

   For example, click **Dashboards**, **Data**, **Infolets**, or **Reports**.
2. From the listing page, click • • • next to the artifact, and then select **Copy URL**.
3. The **Copy URL** dialog displays the unique URL for the artifact. Copy the URL.

Copied URLs can, at times, become invalid. Possible reasons for invalid URLs include:

- The artifact is deleted.
- User access for the artifact is revoked.
- The artifact is renamed, which breaks the URL.
- The artifact is moved to a new folder, resulting in a new URL, which breaks the URL.

Exporting All URLs to a CSV File

Use the **Export URLs** option to create a CSV file that provides the unique URLs for each card, tab, or sub-tab within a connected Oracle Enterprise Performance Management Cloud business process. URLs are grouped by navigation flow and cluster, so the URLs are easier to find within the CSV file. You can open the CSV file with a text editor or Microsoft Excel and embed the relevant URL within the source system pages to serve as a launch point into EPM Cloud.

To export EPM Cloud URLs to a CSV file:

1. Log into an EPM Cloud environment.
2. From the Home page, click the down arrow next to the user name (upper right corner of the screen).
3. On the **Settings and Actions** menu, click **Export URLs**, and then click **Save**.

The system saves a CSV file to the default download folder on your local machine, and the filename is automatically generated with the current date and time of the server; for example, `19_Feb_2021 13_15_38 Navigation Flow URLs.csv`. Find the file in your download folder, and open it with a text editor or with Microsoft Excel.

Viewing the Exported URLs File

The CSV file lists all of the URLs in the business process. Each card, tab (vertical tab), and sub-tab (horizontal tab) has a unique URL. When viewed in a text editor like Notepad or in Microsoft Excel, it identifies the unique URL for each card, tab, and sub-tab, so the URLs for each artifact can be more easily found. URLs are grouped by navigation flow and by cluster.

> **Note:**
>
> Only cards, tabs, and sub-tabs have URLs. Navigation flows and clusters don't have URLs.

Table 7-1 Direct URLs Export File Headers

Header	Description
Navigation Flow Name	The name of the navigation flow; for example, Default or Financial Flow.
Status	Status of the navigation flow; for example, Active or Inactive.
Type	Type of entry; for example, cluster, card, tab, or sub tab
Name	The cluster, card, tab or sub-tab name where the artifact is contained. This entry will be empty for clusters or cards which do not contain an artifact directly.
Artifact Type	The type of artifact; for example, Form, Dashboards, Financial Reports, and URL type artifacts.
Artifact Name	The name of the artifact or, in the case of an URL type artifact, the direct URL of the target page. **Caution:** If a direct URL is displayed, do not confuse this URL with the unique URL you'll use to integrate the connected environments.
URL	This is the unique URL you will use to integrate connected environments.
Visible	Indicates whether the artifact in the navigation flow is visible on the Home page to users or groups; for example, Y or N.
Role/Group	The role or group who can view the navigation flow. If a navigation flow is Global, then it can be seen by all users.
Description	The description of the navigation flow, if provided.

The URLs export file provides the information separated by a vertical bar or pipe (|) delimiter character. See the following example direct URLs export file as viewed in Notepad:

```
Navigation Flow Name|Status|Type|Name|Artifact Type|Artifact Name|URL|Visible|Role/Group|Description
Dashboard 2.0 Test|Inactive|cluster|Strategic Modeling|||Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Model View|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_30-EPM_CA_158|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Consolidation View|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_30-EPM_CA_159|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Templates|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_30-EPM_CA_160|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Tasks|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_21|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Dashboards|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_1|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Infolets|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_204|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Data|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_31|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Reports|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_6|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|tab|Reports|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_6-EPM_TA_606|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|tab|Financial Reports|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_6-EPM_TA_6|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|tab|Documents|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_6-EPM_TA_633|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Rules|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_4|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Approvals|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CA_51|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|cluster|Application||||Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Overview|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_120|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Settings|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_121|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Valid Intersections|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_122|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|sub tab|Setup|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_122-EPM_TA_427-EPM_TA_91|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|sub tab|Reports|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_122-EPM_TA_427-EPM_TA_101|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Data Exchange|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_208|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|sub tab|Data Integration|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_208-EPM_TA_599-EPM_TA_558|Y|Global|Default Navigation
Dashboard 2.0 Test|Inactive|sub tab|Data Maps|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_208-EPM_TA_599-EPM_TA_111|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Jobs|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_124|Y|Global|Default Navigation Flow
Dashboard 2.0 Test|Inactive|card|Cell Level Security|||http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?EC_HL=Y&EC_NF=Dashboard+2.0+Test&EC_PAGE_ID=EPM_CL_22-EPM_CA_214|Y|Global|Default Navigation Flow
```

To view the URLs export file in Microsoft Excel:

1. Open Excel, and then click the **Data** menu.
2. Click **New Query**, then **From File**, and then click **From CSV**.
3. Find and select the CSV file you exported, and then click **Import**. A new window displays the data in the CSV file.
4. To make the first row of the CSV file the header row, click **Edit**, click **Use First Row as Headers**, and then click **Close and Load**.

The resulting Excel file will look like the following example:

	A	B	C	D	E	F	G	H	I	J
1	Navigation Flow Name	Status	Type	Name	Artifact Type	Artifact Name	URL	Visible	Role/Group	Description
2	Dashboard 2.0 Test	Inactive	cluster	Strategic Modeling				Y	Global	Default Navigation Flow
3	Dashboard 2.0 Test	Inactive	card	Model View			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
4	Dashboard 2.0 Test	Inactive	card	Consolidation View			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
5	Dashboard 2.0 Test	Inactive	card	Templates			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
6	Dashboard 2.0 Test	Inactive	card	Tasks			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
7	Dashboard 2.0 Test	Inactive	card	Dashboards			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
8	Dashboard 2.0 Test	Inactive	card	Infolets			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
9	Dashboard 2.0 Test	Inactive	card	Data			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
10	Dashboard 2.0 Test	Inactive	card	Reports			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
11	Dashboard 2.0 Test	Inactive	tab	Reports			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
12	Dashboard 2.0 Test	Inactive	tab	Financial Reports			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
13	Dashboard 2.0 Test	Inactive	tab	Documents			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
14	Dashboard 2.0 Test	Inactive	card	Rules			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
15	Dashboard 2.0 Test	Inactive	card	Approvals			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
16	Dashboard 2.0 Test	Inactive	cluster	Application				Y	Global	Default Navigation Flow
17	Dashboard 2.0 Test	Inactive	card	Overview			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
18	Dashboard 2.0 Test	Inactive	card	Settings			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
19	Dashboard 2.0 Test	Inactive	card	Valid Intersections			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
20	Dashboard 2.0 Test	Inactive	sub tab	Setup			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
21	Dashboard 2.0 Test	Inactive	sub tab	Reports			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
22	Dashboard 2.0 Test	Inactive	card	Data Exchange			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
23	Dashboard 2.0 Test	Inactive	sub tab	Data Integration			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
24	Dashboard 2.0 Test	Inactive	sub tab	Data Maps			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
25	Dashboard 2.0 Test	Inactive	card	Jobs			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow
26	Dashboard 2.0 Test	Inactive	card	Cell Level Security			http://slcar261.usdv1.oraclecloud.com:10151/HyperionPlanning?E	Y	Global	Default Navigation Flow

Find and copy the unique URL (found in the URL column) for the card, tab, or sub-tab that you wish to have integrated into the other connected environment. Only the URL target will open and users with access to the targeted artifact can perform the same actions as if they're working within the target business process.

8

Designing Custom Navigation Flows

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

Related Topics

- Understanding Navigation Flows
- Viewing and Working with Navigation Flows

Understanding Navigation Flows

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

Videos

Your Goal	Watch This Video
View the highlights of customizing workflows for connected environments.	 Overview: Configure Navigation Flows to Integrate EPM Cloud Business Processes

Related Topics

- What Can Be Customized in the Business Process Interface?
- Navigation Flow Customization Categories
- Navigation Flow Permissions
- Predefined Navigation Flows
- Viewing and Working with Navigation Flows

What Can Be Customized in the Business Process Interface?

Navigation flows are made up of cards. Each card contains content with information that is displayed as one or more tabbed pages. The content included on cards can be URLs or artifacts such as forms, dashboards, reports. Cards can be grouped into clusters.

> **Note:**
>
> If Dashboard 2.0 and Forms 2.0 are enabled in your business process:
>
> - Navigation flows support the features included in Dashboard 2.0 and Forms 2.0, including the ability to edit a form from within the runtime dashboard, and editing it with or without data. Dashboard 2.0 component actions such as **Open Form** and **Edit Form** also work directly within the navigation flow.
> - If a dashboard contains multiple components, when you edit a dashboard from within a navigation flow, the focus will be set to the last component in the dashboard.
>
> See Working with 2.0 Dashboards in *Working with FreeForm*.

Navigation flows can be customized in the following ways:

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

See Viewing and Working with Navigation Flows.

Navigation Flow Customization Categories

Navigation flows are categorized as follows for customization:

1. Global: Navigation flows are seen by all users
2. Role: Navigation flows are seen only by users in a specific role; for example, Service Administrator or Power User
3. Group: Navigation flows are seen only by users belonging to a specific group; for example, Sales

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

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

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

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

Navigation Flow Permissions

The business process offers three levels of permissions for navigation flows:

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

Predefined Navigation Flows

The business process comes with one predefined navigation flow, called Default. The Default navigation flow is read only; therefore, you can't make any modifications to it.

These are the operations you can and cannot perform on the Default navigation flow:

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

Understanding Navigation Flow Status after Import

When you import a snapshot containing an active navigation flow into an environment containing an active navigation flow of the same category (role, group, or global), then the new navigation flow being imported is deactivated.

For example, the following navigation flows are currently active on your system:

- NF Sales Q2 (Sales group)
- NF Administrator (Service Administrator role)
- NF Custom Default (global)

Let's say you import a navigation flow called NF Sales Q3, which uses the same Sales group as NF Sales Q2. The NF Sales Q3 navigation flow will be imported as inactive and the existing NF Sales Q2 will remain active for the Sales group. If you want the new NF Sales Q3 to be active, you must manually activate it (which will inactivate the old NF Sales Q2). We deactivate the imported Q3 navigation flow because the import might occur ahead of the Q3 start date, and we want to ensure the Q3 navigation flow isn't activated before the Q3 start date.

If a new application is created from a Migration snapshot that includes an active custom global navigation flow (for example, NF Custom), the NF Custom global navigation flow will become inactive after Migration import. The same is true if the application is cloned. The Default global navigation flow will become the active global navigation flow and you'll need to ensure you activate the global NF Custom navigation flow after the new application is created. If all users are using the NF Custom global navigation flow, they might experience a service interruption until the NF Custom navigation flow is activated.

When designing navigation flows:

- Create navigation flows for specific groups or classes of users. Then associate the navigation flow to corresponding user groups.
- Avoid creating one global navigation flow for all users.

Viewing and Working with Navigation Flows

Users assigned the Service Administrator role can view all navigation flows, including the predefined navigation flow, on the Navigation Flows page. Users assigned to other roles and groups will only see the navigation flows to which they have access.

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

Viewing Navigation Flows

To view the Navigation Flow page:

1. On the Home page, click **Tools**.
2. Click **Navigation Flows**.

Working with Navigation Flows

To work with a navigation flow, perform an action:

- For design best practices and considerations, see Navigation Flow Design Best Practices and Naming Considerations.
- To create and duplicate navigation flows, see Creating and Duplicating Navigation Flows.
- To edit a navigation flow, see Editing a Navigation Flow.
- To activate or deactivate a navigation flow, see Activating and Deactivating Navigation Flows.
- To validate navigation flows and to learn how to find and resolve navigation flow elements with missing artifacts, see Using Validate to Find Missing Artifacts in Navigation Flows.
- To resolve a navigation flow in the listing that is displaying a warning icon  , see Resolving Navigation Flows that Display a Warning Icon.
- To rename cards and tabs, see Customizing Labels for Cards, Tabs, and Clusters.
- To customize the graphics used for cards and tabs, see Customizing Icons for Cards and Vertical Tabs.
- To hide and unhide cards and tabs, see Hiding and Unhiding Clusters, Cards, and Tabs.
- To change the display order of cards on the Home page, see Changing the Display Order of Cards on the Home Page.
- To add cards, see Adding Cards.
- To add tabs, see Adding Tabs to a Tabular Page.
- To remove navigation flows, cards, and tabs, see Removing Navigation Flows, Cards, and Tabs.
- To group cards into clusters, see Grouping Cards into Clusters.

Troubleshooting

For help with resolving navigation flow issues, see Handling Issues with Navigation Flows in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Navigation Flow Design Best Practices and Naming Considerations

Design Best Practices

To provide an optimal user experience and to avoid excessive scrolling on the Home page and within cards and tabs, you must adhere to the following guidelines when designing navigation flows:

- Keep the top level items (cards and clusters) to no more than 16 visible items.
- Add no more than 16 visible cards to a cluster.
- Add no more than 10 visible vertical tabs within a card.
- Add no more than 20 visible sub tabs (horizontal tabs) within a vertical tab.
- Label names on sub tabs (horizontal tabs) display only the first 30 characters at runtime. Hovering your cursor over the tab reveals the entire label.

> **Note:**
>
> If you attempt to exceed the visibility limits, you will see a warning message telling you that you've exceeded the limit.

Naming Considerations

There are naming restrictions for navigation flows, cards, clusters, tabs, and infolets (if your business process uses infolets) in navigation flows. You cannot use these special characters:

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

Creating and Duplicating Navigation Flows

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

To create and duplicate a navigation flow:

1. Open the **Navigation Flow** page. See Viewing and Working with Navigation Flows.
2. In the upper right-hand corner of the page, click  , and then select **Create Copy**.
3. Enter a name for the navigation flow, and then click **OK**.

> **Note:**
>
> Ensure that you adhere to the navigation flow naming restrictions outlined in Navigation Flow Design Best Practices and Naming Considerations.

4. Edit details for the navigation flow. See Editing a Navigation Flow.

> **Note:**
>
> New flows are marked **Inactive** until they are activated by the Service Administrator. To activate a navigation flow, see Activating and Deactivating Navigation Flows.

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

Editing a Navigation Flow

To edit a navigation flow:

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

> **Note:**
>
> The predefined navigation flow isn't editable. However, you can create a copy of a predefined navigation flow and edit the copy. See Predefined Navigation Flows.

You'll see a page listing the cards and clusters in the navigation flow. On this page you can edit the role or goup assignation, designate which clusters and cards are visible on the Home page, change the order in which the navigation flow clusters and cards are displayed, add cards to clusters or remove them, and remove clusters and cards from the navigation flow.

- **Assign To**: Click  to assign the navigation flow to a group of users or to a role.
- **Visible**: Edit the visibility of the navigation flow clusters and cards on the Home page by selecting or deselecting them in the **Visible** column.

> **Note:**
>
> Ensure that you adhere to the navigation flow guidelines for visibility outlined in Navigation Flow Design Best Practices and Naming Considerations.

- **Order**: Clusters and cards are listed in the order in which they are displayed on the Home page, if visible. Selecting an up or down arrow option repositions the clusters and cards in the listing and changes the display order of the clusters and cards on the Home page. Selecting the right arrow moves a card into a cluster.

- **Remove**: Removes the cluster or card from the navigation flow.

3. Click a cluster or card to edit details. For descriptions of card details, see the following topics:
   - Adding Cards
   - Adding Tabs to a Tabular Page

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

Activating and Deactivating Navigation Flows

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

> **Note:**
>
> Each business process requires one active global navigation flow. To make a different global navigation flow active, select another global navigation flow and activate it.

For information about categories, see Navigation Flow Customization Categories.

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

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

To activate or deactivate a navigation flow:

1. Open the **Navigation Flow** page. See Viewing and Working with Navigation Flows.
2. In the **Active** column, click **Active** or **Inactive**. An active flow will be marked inactive. Conversely, an inactive flow will be marked active.

Using Validate to Find Missing Artifacts in Navigation Flows

While viewing the Navigation Flow listing, you might see navigation flow nodes or artifacts displaying an error icon  . This error occurs because artifacts that were associated with the navigation flow were either renamed or removed and they are now considered missing. You'll need to edit the navigation flow to associate it with either a renamed artifact or with a different artifact before you can activate the navigation flow. You might not realize that artifacts are considered missing, so it's recommended that you validate your navigation flows before setting them to Active status.

> **Note:**
>
> To resolve a navigation flow displaying a warning icon  , see Resolving Navigation Flows that Display a Warning Icon.

To validate navigation flows to find missing artifacts and reassociate them:

1. Open the **Navigation Flow** page. See Viewing and Working with Navigation Flows.
2. If they aren't already deactivated, set the navigation flows you want to validate to **Inactive** status. See Activating and Deactivating Navigation Flows.
3. Highlight the row or rows of the navigation flows you want to validate.
4. Click  , and then select **Validate**.

   If artifacts are missing, you'll see an error message indicating which navigation flows are referencing artifacts that cannot be found.
5. Click the name of each navigation flow with the error, and then expand the nodes which display the error icon  until you reach the Manage page which displays the missing artifact.
6. For **Artifact**, click  to select the renamed artifact or a different artifact in the Artifact Library.
7. Click **Save and Close**.
8. Repeat the validation on your navigation flows and drill down to make corrections, as needed, until you see a message indicating the navigation flows are valid.
9. Activate the navigation flows. See Activating and Deactivating Navigation Flows.

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

Resolving Navigation Flows that Display a Warning Icon

While viewing the Navigation Flow listing, you might see a navigation flow displaying a warning icon  . This occurs because the group that was associated with the navigation flow was deleted. You'll need to edit the navigation flow to associate it with a group or role before you can activate the navigation flow.

> **Note:**
>
> To resolve navigation flows displaying an error icon  , see Using Validate to Find Missing Artifacts in Navigation Flows.

To resolve the navigation flow:

1. Open the **Navigation Flow** page. See Viewing and Working with Navigation Flows.
2. Click the name of the navigation flow displaying the warning icon  .

3. For **Assign To**, click  to assign the navigation flow to a group of users or to a role, then click **Save and Close**.
4. Activate the navigation flow. See Activating and Deactivating Navigation Flows.

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

Customizing Labels for Cards, Tabs, and Clusters

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

> **Note:**
>
> Do not update the labels for cards, tabs, or clusters against the Default navigation flow. Only customize labels for custom navigation flows.

To customize labels for cards, tabs, and clusters:

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

   > **Note:**
   >
   > - You can edit the label here. However, if the label is defined in the **Artifact Labels** page on the **Tools** cluster, that definition will take precedence and will display during runtime. To change a label permanently, redefine it in the **Artifact Labels** page.
   >
   >   See Specifying Artifact Labels.
   > - Ensure that you adhere to the naming restrictions outlined in Navigation Flow Design Best Practices and Naming Considerations.

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

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

Customizing Icons for Cards and Vertical Tabs

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

To customize the icons for cards and vertical tabs:

1. Open the **Navigation Flow** page and click the name of the navigation flow you want to edit. See Viewing and Working with Navigation Flows.
2. If customizing the icon for a card:
   a. Click the name of the card you want to edit.
   b. Click the icon for the card, select a new icon from the library, and then save it.
3. If customizing the icon for a tab:
   a. Click the name of the icon you want to edit.
   b. In the tab listing on the **Manage Tab** page, click the name of the tab you want to edit.
   c. Click the icon for the tab, select a new icon from the library, and then save it.

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

Hiding and Unhiding Clusters, Cards, and Tabs

You cannot hide the following navigation elements:

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

Ensure that you adhere to the navigation flow guidelines for visibility outlined in Navigation Flow Design Best Practices and Naming Considerations.

To hide and unhide clusters, cards, and tabs:

1. Click the **Navigation Flow** icon and click the name of the navigation flow you want to edit. See Viewing and Working with Navigation Flows.
2. Edit the Home page visibility of the navigation flow clusters and cards by selecting or deselecting them in the **Visible** column.
3. If hiding or unhiding a tab:
   a. Click the name of the card you want to edit.
   b. In the tab listing on the **Manage Tab** page, select or deselect the check box in the **Visible** column.

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

Changing the Display Order of Cards on the Home Page

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

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

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

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

Adding Cards

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

To ensure an optimal user experience, review the navigation flow design best practices. See Navigation Flow Design Best Practices and Naming Considerations.

You can also group cards into clusters. See Grouping Cards into Clusters.

To add cards to a navigation flow:

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

   Note the following:

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

     See Specifying Artifact Labels.
   - A reference card is a card that is already referenced from another navigation flow. References to already referenced cards are not supported in navigation flows and will not be available for selection in the Object Library when adding an existing card; for example:
     - A card referring to a remote artifact or remote tab will not be available from the Object Library when adding an existing card.
     - A card referring to a tab from another navigation flow will not be available from the Object Library when adding an existing card.

   A card is added to the list as a sibling under the card or cluster that is currently selected. To add a card to a cluster, see Grouping Cards into Clusters.

   > **Note:**
   >
   > Cards that are added without first selecting a card or cluster are added to the end of the list.

3. To add a new card to the navigation flow, right-click a node in the list (or click ), click **Add Card**, and then select details for the new card:

Table 8-1 New Card Details

Label	Description
Name	Enter a label for the card. Ensure that you adhere to the naming restrictions outlined in Navigation Flow Design Best Practices and Naming Considerations.
Visible	Select whether the card is visible to users on the Home page.
Cluster	If clusters exist, select a cluster for the card or select **None**.
Icon	Select the graphic that will be displayed for the card you're creating. Choose from the available graphics provided in the graphics library.
Page Type	Select **Single Page** or **Tabular Page** format.
Content Source	If you selected the **Single Page** format, select **Artifact** or **URL**: • For **Artifact**, click  to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include forms, dashboards, and reports. To select an artifact from another environment, select the target environment under **My Connections**, and then choose the artifact you want to add. • For **URL**, enter a complete URL; for example, a URL to embed an Oracle Analytics Cloud dashboard in a card, then click **Preview** to validate the URL in a popup window. Insert only external site URLs starting with the `https://` security protocol. Don't use internal or relative URLs or URLs for unconsenting third-party sites. See About Using URLs to Embed Third-Party Pages in EPM Cloud Applications.
Orientation	If you selected the **Tabular Page** format, select **Vertical** or **Horizontal**, and then add new or existing tabs and sub tabs. See Adding Tabs to a Tabular Page.

A card is added to the list as a sibling under the card or cluster that is currently selected. To add a card to a cluster, see Grouping Cards into Clusters.

> **Note:**
>
> Cards that are added without first selecting a card or cluster are added to the end of the list.

4. Click **Save and Close**.

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

Adding Tabs to a Tabular Page

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

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** and **Reports** tabs are at the bottom of the page.

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

To ensure an optimal user experience, review the navigation flow design best practices. See Navigation Flow Design Best Practices and Naming Considerations.

To add tabs to a tabular page:

1. Open the **Navigation Flow** page and click the name of the navigation flow you want to edit. See Viewing and Working with Navigation Flows.
2. Edit an existing card by clicking the name of the card you want to edit, or add a new card by right-clicking a card (or clicking), and then clicking **Add Card**.
3. On the **Manage Card** page, select the following options:
   - For **Page Type**, select **Tabular Page**.
   - For **Orientation**, select **Vertical** or **Horizontal**.

   A tab listing displays at the bottom of the **Manage Card** page.
4. To edit an existing tab, click a tab name from the tab listing, and edit tab details.
5. To add a new or existing tab:
   a. To add an existing tab, right-click a tab in the listing at the bottom of the **Manage Card** page, click **Add Existing Tab** (or click the **Add Existing Tab** button), select a tab from the Object Library, and then click **OK**.

      > **Note:**
      >
      > A reference tab is a tab that is already referenced from another navigation flow. References to already referenced tabs are not supported in navigation flows and will not be available for selection in the Object Library when adding an existing tab; for example:
      >
      > - A tab referring to a remote artifact or remote sub-tab will not be available from the Object Library when adding an existing tab.
      > - A tab referring to a sub tab from another navigation flow will not be available from the Object Library when adding an existing tab.

   b. To add a new tab, right-click a tab, click **Add New Tab** (or click the **Add New Tab** button), and then edit tab details.
   c. Select the content for the new tab:
      - For **Artifact**, click to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include forms, dashboards, and reports. To select an artifact from another

environment, select the target environment under **My Connections**, and then choose the artifact you want to add.

- For **URL**, enter a complete URL; for example, a URL to embed an Oracle Analytics Cloud dashboard in a tab, then click **Preview** to validate the URL in a popup window.

  Insert only external site URLs starting with the `https://` security protocol. Don't use internal or relative URLs or URLs for unconsenting third-party sites. See About Using URLs to Embed Third-Party Pages in EPM Cloud Applications.

A tab is added to the list as a sibling under the tab that is currently selected.

> **Note:**
>
> Tabs that are added without first selecting a tab are added to the end of the list.

6. To add new or existing sub tabs to a tab:
   a. Click the name of a tab in the in the tab listing.
   b. For **Page Type**, select **Tabular Page**.
   c. Right-click a tab, and click **Add New Sub Tab** or **Add Existing Sub Tab** (or click the **Add New Sub Tab** or **Add Existing Sub Tab** button), and then edit sub tab details.
   d. Select the content for the new sub tab:

      - For **Artifact**, click  to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include forms, dashboards, and reports. To select an artifact from another environment, select the target environment under **My Connections**, and then choose the artifact you want to add.
      - For **URL**, enter a complete URL; for example, a URL to embed an Oracle Analytics Cloud dashboard in a sub tab. Click **Preview** to validate the URL in a popup window.

        Insert only external site URLs starting with the `https://` security protocol. Don't use internal or relative URLs or URLs for unconsenting third-party sites. See About Using URLs to Embed Third-Party Pages in EPM Cloud Applications.

      A sub tab is added to the list as a sibling under the tab that is currently selected.

> **Note:**
>
> Sub tabs that are added without first selecting a tab are added to the end of the list.

7. Click **Save and Close**.

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

Note:

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

  See Specifying Artifact Labels.

About Using URLs to Embed Third-Party Pages in EPM Cloud Applications

Oracle Enterprise Performance Management Cloud uses IFrame to embed third-party URLs. IFrame requires that the page that is being embedded must give approval to the page that is embedding it. For example, if we want to embed a page from sharepoint.com into an EPM Cloud application, then sharepoint.com must allow oraclecloud.com to embed the page from sharepoint.com.

This can be achieved by adding oraclecloud.com in the Content Security Policy of the web application whose page you need to embed.

When embedding a third-party page you also must consider if the page is available to the public or if it requires a login. For example, pages from wikipedia.org do not need any authentication.

If you are embedding a page that requires authentication, you need to see if SSO can be enabled for the page. If not, then your page may not load inside IFrame. As a workaround, log in to that web application in another browser tab, and then if you access the same page from the EPM Cloud application, it will open.

This URL support feature allows you to embed the following types of pages:

- Other Oracle products (would need to enable SSO)
- Web applications owned by customer (would need to allow the EPM Cloud application by updating the Content Security Policy and enabling SSO, or likewise)
- Pages from the public domain (for example, wikipedia.org)

Note:

Though not supported by Oracle, you can also embed links to the following resources within navigation flow cards, tabs, and sub tabs:

- Google Sheets
- Files stored in Google Drive (for example, PDF and Excel)
- Files stored in Microsoft Office 365

Use the instructions provided by the third-party sites to generate a URL which can then be used in your EPM Cloud application.

Removing Navigation Flows, Cards, and Tabs

You can't remove the following navigation elements:

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

To remove navigation flows, cards, and tabs:

1. Open the **Navigation Flow** page. See Viewing and Working with Navigation Flows.
2. If removing a navigation flow:
   a. Select the navigation flow you want to remove.
   b. In the upper right-hand corner of the page, click  , and then select **Delete**.

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

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

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

Grouping Cards into Clusters

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

To ensure an optimal user experience, review the navigation flow design best practices. See Navigation Flow Design Best Practices and Naming Considerations.

To assign cards to clusters:

1. Create a new cluster or add an existing cluster:
   a. Open the **Navigation Flow** page and click the name of the navigation flow in which you want to add a cluster. See Viewing and Working with Navigation Flows.

b. To create a new cluster, right-click a card or cluster in the list (or click), click **Add Cluster**, enter or select the cluster details, and then choose a graphic for the cluster.

> **Note:**
>
> Ensure that you adhere to the visibility and naming restrictions outlined in Navigation Flow Design Best Practices and Naming Considerations.

A cluster is added to the list as a sibling under the cluster that is currently selected.

> **Note:**
>
> Clusters that are added without first selecting a card or cluster are added to the end of the list.

c. To add an existing cluster, right-click a card or cluster in the list (or click), click **Add Existing Card/Cluster**. If there is an existing cluster you want to add from another environment, right-click a card or cluster in the list (or click), click **Add Existing Card/Cluster**, select the target environment under **My Connections**, and then choose the cluster you want to add to your navigation flow.

> **Note:**
>
> - Clusters can't be directly selected from Narrative Reporting and Profitability and Cost Management using the **Add Existing Card/Cluster** option.
> - Clusters that are added from another navigation flow or from another environment will display the localized labels that were defined in the source navigation flow. To update the cluster labels in your navigation flow, on the Home page, click **Tools**, and then click **Artifact Labels**.
>
>   See Specifying Artifact Labels.
> - A reference cluster is a cluster that is already referenced from another navigation flow. References to already referenced clusters are not supported in navigation flows and will not be available for selection in the Object Library when adding an existing cluster.

A cluster is added to the list as a sibling under the card or cluster that is currently selected.

> **Note:**
>
> Clusters that are added without first selecting a card or cluster are added to the end of the list.

d. Click **Save and Close**.

2. Select the cards to include in the cluster using one of these options:

   a. Navigate to the card you want to add. If the card is within another environment, first select the environment under **My Connections**, and then navigate to the card in that environment. Assign the card to a cluster using one of these options:

      - To the right of the card in the **Order** column, click ›, select the cluster, and then click **OK**.
      - Click the name of the card to view the card details, then for **Cluster** select a cluster for the card, and then click **OK**.

   b. Navigate to the cluster in which you want to add a card, then right-click the cluster (or click +), click **Add Card In Cluster**, and then select an option:

      - Select **Add Existing Card**, to select an existing card or to add existing cards from another cluster to the selected cluster.
      - Select **Add Card**, and then enter card details to add a new card to the selected cluster.

      > **Note:**
      >
      > You cannot add a card to a cluster if the card or the cluster is already referenced from another navigation flow.

   c. Click **Save and Close**.

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

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

Reloading a Navigation Flow

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

To reload a navigation flow after making design changes:

1. From the Home page, click the down arrow next to the user name (upper right corner of the screen).
2. On the **Settings and Actions** menu, click **Reload Navigation Flow**.

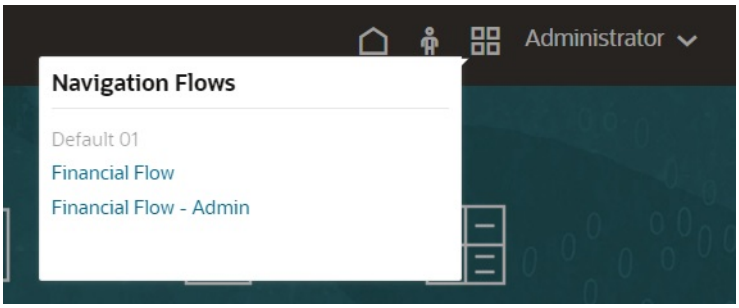
Switching Navigation Flows at Runtime

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

> **Note:**
>
> Users assigned the Service Administrator role can access all navigation flows.

To switch navigation flows at runtime:

1. From the Home page, click .
2. Select the navigation flow you want to view.



Sharing the Current Page URL for Cards, Tabs, and Sub Tabs

Copy the URL for the currently displayed card, tab, or sub tab and share it with other users who have access.

Use the **Share Current Page URL** option on the **Settings and Actions** menu to copy the URL for the currently displayed card, tab, or sub tab in your Oracle Enterprise Performance Management Cloud business process.

URLs can be copied and then shared with other users in an email, Slack, or any other messaging application. Also, copied URLs can be used in a slide show presentation such as PowerPoint. Note that users can only access cards, tabs, or sub tabs if they are authorized to access them.

When the copied URL is launched the page will be rendered with the full header, meaning they will be able to navigate to all of the pages to which they have access within the navigation flow. If the user launching the URL is not logged in, they will be redirected to a login page, then after login the URL page will open.

To copy a URL for sharing:

1. Navigate to the page (card, tab, or sub tab) in your application that you want to copy.

   For example, from the Home page, click Dashboards to view the Dashboards page.
2. Click the down arrow next to your user name (upper right corner of the screen).
3. On the **Settings and Actions** menu, select **Share Current Page URL**.

> **Note:**
>
> The **Share Current Page URL** menu option only displays if you're currently viewing a card, tab, or sub tab. If you're currently viewing the Home page, the **Share Current Page URL** menu option won't display on the **Settings and Actions** menu.

4. Select the displayed URL and copy it by pressing Ctrl+C.

   After pasting it into a browser, the URL will enable direct access to the current page in the application.

Considerations

Note the following considerations when using the **Share Current Page URL** option:

- The page specified in the URL will not open if the navigation flow was deleted, modified, or made inactive. Instead, you'll be redirected to the home page of the currently active navigation flow.
- If you're using a classic environment, you cannot share URLs. URLs are not supported for classic pages.
- URLs cannot be embedded into dashboards as the content source.
- URLs cannot be embedded into navigation flows as the content source.
- URLs cannot be shared for pages with dynamic tabs. Dynamic tabs are local and limited to the current user session, and therefore cannot be shared.
- URLs cannot be shared for infolets. URLs are not supported for infolets.

9

Editing Dimensions in the Simplified Dimension Editor

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

Related Topics

- About Editing Dimensions in the Simplified Dimension Editor
  The Simplified dimension editor displays dimensions and members in a grid format.
- Dimension Overview
- About Dimensions and Members
- About Sparse and Dense Dimensions
- About Dimension Hierarchies
- About Custom Dimensions
- Aggregation Options
- Storage Options
- About Entities
- About Accounts
- Account Types
- Saved Assumptions
- Data Type
- Accounts, Entities, Periods, and Cubes
- Accessing the Simplified Dimension Editor
- Working in the Edit Member Properties Grid
  Work with elements in the **Edit Member Properties** grid interface.
- Switching to Another Dimension
- Customizing the Column Layout
- Context Menu Options
- Viewing Ancestors
- Showing Member Usage in an Application
- Focusing Your Editing
- Finding Members
- Moving Members
- Sorting Members
- Moving Members to Another Hierarchy
- Working with Member Formulas
- Copying Member Names from Microsoft Excel

- Editing Member Properties
- Accessing Edit Member Properties
- Adding Members
- Editing Members
- Deleting Members
- Adding Shared Members
- Assigning Access to Dimension Members
- Adding, Editing, and Removing Access to Members
- About Effective Access Permissions to Shared Members
- Editing Dimension Properties
- Working with Attributes
- Deleting Attributes
- Working with UDAs
- Working with Attribute Values

About Editing Dimensions in the Simplified Dimension Editor

The Simplified dimension editor displays dimensions and members in a grid format.

> **Note:**
>
> This section documents how to edit dimensions using the Simplified dimension editor which was released in the 17.05 (May 2017) update.
>
> To use the Classic Dimension Editor, which is accessed by using the **Dimensions** link in the Navigator menu, see Administering Dimensions.

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

- The grid provides the ability to do a filtered search within the grid and to reorder columns using drag-and-drop. The expansion states of the member hierarchy in the grid are saved for the next session.
- The column selector interface enables you to select the columns that will display within the grid. The column selector groups properties according to their functionality and saves the column selections that you add to the grid for the next session. There is also an option to resize the columns so that all columns are visible on the grid without scrolling.

Dimension Overview

Related Topics

- About Dimensions and Members
- About Sparse and Dense Dimensions

- About Dimension Hierarchies
- About Custom Dimensions
- About Entities
- About Accounts
- Accounts, Entities, Periods, and Cubes

About Dimensions and Members

Dimensions are data categories used to organize business data for retrieval and preservation of values.

Dimensions contain hierarchies of related *members* grouped within them. For example, a Years dimension often includes members for each time period, such as quarters and months.

FreeForm apps can have up to 29 total dimensions across all cubes: 26 custom dimensions plus three Account, Period, and Entity dimension types.

About Sparse and Dense Dimensions

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

About Dimension Hierarchies

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

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

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

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

About Custom Dimensions

In FreeForm apps, you can have up to 26 custom dimensions. The Scenario, Version, Period, and Year dimensions can be defined as custom dimensions. See Understanding FreeForm.

> **Caution:**
>
> You can't delete custom dimensions after you create them.

- Aggregation Options
- Storage Options

Aggregation Options

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

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

Storage Options

When working with dimensions, it's important to understand the different data storage options and how to use them in the business process.

- Storage Options
- About Dynamic Calc
- Dynamic Calc Versus Dynamic Calc and Store
- About Store Data Storage
- About Shared Data Storage
- About Never Share Data Storage
- About Label Only Data Storage

Storage Options

Table 9-1 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, the application calculates data values of members, and disregards these values. As a best practice, Oracle recommends a limit of 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 doesn't significantly lower disk usage. For data values accessed concurrently by many users, use **Dynamic Calc**. Retrieval time may be significantly lower than for **Dynamic Calc and Store**.

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

About Store Data Storage

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

About Shared Data Storage

Use **Shared** to allow alternate rollup structures in the application.

About Never Share Data Storage

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

About Label Only Data Storage

Label Only members are virtual members; they are typically used for navigation and have no associated data.

Note:

- You can't assign level 0 members as **Label Only**.
- **Label Only** members can display values.
- Making dimension members **Label Only** minimizes database space by decreasing block size.
- You can't assign attributes to **Label Only** members.
- Data storage for children of **Label Only** parents is set to **Never Share** by default.

Caution:

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

About Entities

Entities typically match your organization's structure, such as geographical regions, departments, or divisions.

About Accounts

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

Related Topics

- Account Types
- Saved Assumptions
- Data Type

Account Types

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

Examples of Using Account Types

Table 9-2 Using Account Types

Account Type	Purpose
Expense	Cost of doing business
Revenue	Source of income

Table 9-2 (Cont.) Using Account Types

Account Type	Purpose
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 9-3 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 the application calculates the value of summary time periods.

Table 9-4 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
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

Table 9-4 (Cont.) Time Balance Properties

Time Balance Property	Description	Example
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's a leap year, the result is calculated: (10 * 31 + 15 * 29 + 20 * 31) / 91 = 15 Note that this time balance property is only supported for dimensions bound to a block storage cube. Aggregate storage cubes do not support the Weighted Average - Actual_Actual time balance property.	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; doesn't account for leap years. In the example, the average for Q1 is calculated: (1) Multiply each month's value in Q1 by the number of days in the month, (2) Sum these values, (3) Divide the total by the number of days in Q1. Assuming it's not a leap year, the result is calculated: (10 * 31 + 15 * 28 + 20 * 31) / 90 = 15 Note that this time balance property is only supported for dimensions bound to a block storage cube. Aggregate storage cubes do not support the Weighted Average - Actual_365 time balance property.	Jan: 10 Feb: 15 Mar: 20 Q1: 15
Custom	Spreading is disabled and the application designer is expected to add customized spreading. For instance, you can feed a value into the period total (Q1) using Groovy rules and it will calculate the summary time periods using the Flow method.	NA
Disable	Spreading is disabled and the summary time period is read-only. Disable prevents data from being entered on non-level zero periods. The Flow method is used to aggregate into the period total (Q1), but it won't spread down to the summary time periods.	NA

Note:

- You can use the Weighted Average - Actual_Actual and Weighted Average - Actual_365 time balance properties only with a standard monthly calendar that rolls up to four quarters.

  For information on how the application calculates and spreads data with the different Time Balance settings, see How Spreading Data Works in *Working with FreeForm*.

- For Custom and Disable time balance properties, the application designer needs to be aware of the storage characteristics of the member that they write to, be it aggregate storage or block storage. For example, you can only save to level zero members in aggregate storage and if you try to save to a dynamic calc member, it'll be ignored and overwritten when recalculated. Also note that spreading only happens in the grid prior to save, be it automatic or using a Groovy rule. After the grid is saved, normal Essbase behavior will take over with regard to saving and reading data (that is, normal outline math will apply, member formulas, time balance, and so on).

Account Types and Variance Reporting

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

- Expense: The actual value is subtracted from the budgeted value to determine the variance
- Non-Expense: The budgeted value is subtracted from the actual value to determine the variance

Setting Account Calculations for Zeros and Missing Values

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

Table 9-5 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
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

Table 9-5 (Cont.) Effect of Skip Options When Time Balance is Set to First

Skip Option	Description	Example
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, the application subtracts the actual amount from the budgeted amount.
- Determine the value for office floor space by using the time period's last value.
- Make an assumption about the number of product units sold at the end of the time period. Determine the final value for the summary time period by aggregating the number of units sold across time periods.

Data Type

Data type determines how values are stored in account members.

Available data types for account members' values:

- Percentage: Stores a numeric value and displays as a percent.
- Date: Displays as a date.
- Text: Displays as text.

Accounts, Entities, Periods, and Cubes

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

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

Entities and Cubes

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

Accounts and Cubes

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

Accessing the Simplified Dimension Editor

To access the Simplified dimension editor:

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

Working in the Edit Member Properties Grid

Work with elements in the **Edit Member Properties** grid interface.

Related Topics

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

Switching to Another Dimension

To switch to another dimension while viewing the **Edit Member Properties** grid:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.

2. Click the down arrow next to the dimension name at the top of the page.

> **Note:**
>
> The dimension drop-down displays all of the dimensions, regardless of the cube you are currently viewing.

Customizing the Column Layout

Each column in the grid represents a member property (**Member Name**, **Parent Member**, **Default Data Storage**, and so on). The columns that initially display on the grid can be different based on which dimension type you're editing.

You can customize the layout of columns in the **Edit Member Properties** grid by hiding, unhiding, resizing, or reordering columns. You can also display the complete set of properties (all columns) by selecting **Select All** in the **Select Columns** dialog box.

The properties listed in the column selector are not a flat list of properties. Instead, properties are grouped in the column selector according to their functionality. For example, Account-specific properties such as **Account Type**, **Variance Reporting**, and **Time Balance**, are grouped under **Account** in the column selector. When you select **Account** in the column selector, all properties in the group are selected for inclusion in the grid. Then you can clear the checkboxes for any properties that you don't want added to the grid.

To customize the column layout in the **Edit Member Properties** grid:

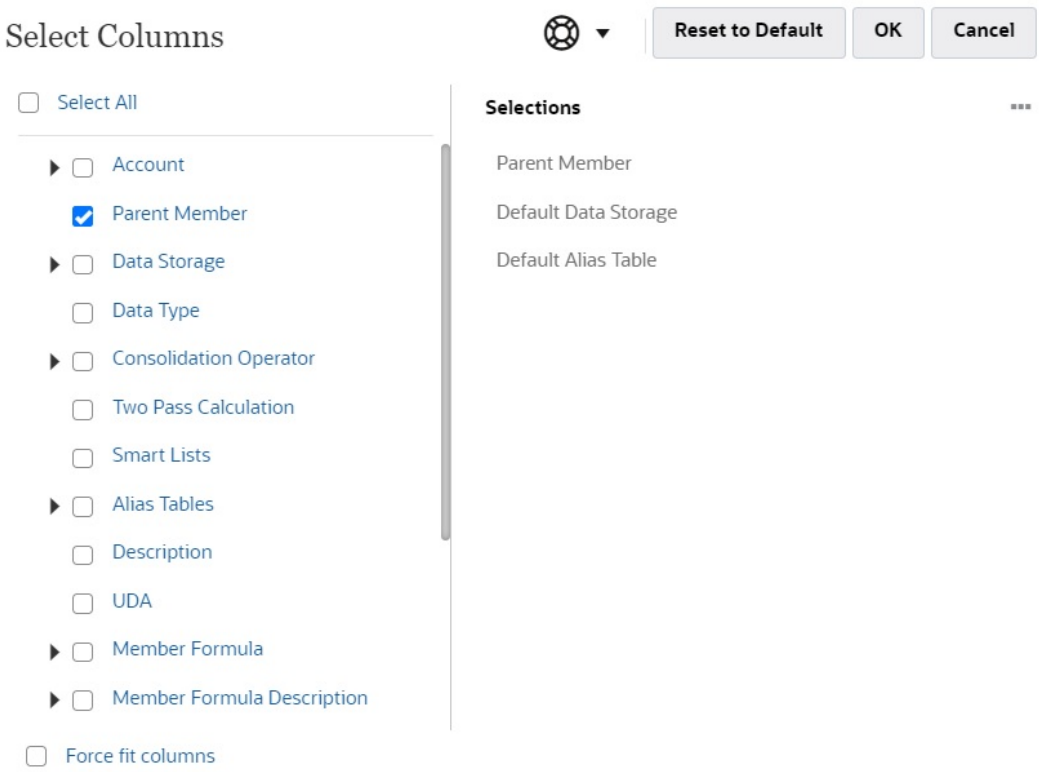
1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.

2. On the grid, right-click any column heading.

> **Tip:**
>
> Alternatively, you can display this dialog box by clicking  or by clicking **Actions**, and then **Select Columns**.

The **Select Columns** dialog box displays.



3. Select or clear the check boxes for the columns (properties) you want to add to or remove from the grid. Select a top-level group such as **Account** or **Data Storage** to select all of the properties grouped under those categories.
   - To display all property columns on the grid, click **Select All**.
   - In the **Selections** pane of the **Select Columns** dialog box, columns appear in the order they are selected, and this is the order in which they will be displayed on the grid. To reorder the columns on the grid, hover over a column in the **Selections** pane, then select the up or down arrows to move the column up or down in the order.

     > **Note:**
     >
     > You can also reorder columns by dragging and dropping them directly on the grid.

   - To remove a column from the **Selections** pane, hover over a column and click .
   - To remove all columns from the **Selections** pane, click , and then select **Remove All**.
   - To view the columns that initially displayed on the grid, click **Reset to Default**.

> **Note:**
>
> After making your selections, the columns you select will be saved for the next session.

4. To change the size of the grid or the columns that are displayed on the grid, select or clear the **Force fit columns** option. Selecting the **Force fit columns** option resizes the columns so that all columns are visible on the grid without scrolling.

Context Menu Options

In the **Edit Member Properties** grid, right-click on a cell in the grid to view the following context menu options:

- **Move** options:
  - **Move Up**: Moves the member up one position
  - **Move Down**: Moves the member down one position
- **Zoom In** options:
  - **Zoom in Next Level**: Displays all members one level below the selected member
  - **Zoom in All Levels**: Displays all descendant members below the selected member
  - **Zoom in Bottom Level**: Displays all descendant members of the selected member that have no children
- **Zoom Out**: Displays the member one level above the selected member
- **Keep Selected**: Displays only the selected row or column on the grid
- **Remove Selected**: Removes the selected row or column from the grid
- **Delete**: Deletes the selected member, its descendants, and associated shared members

> **Note:**
>
> The context menus might differ depending on the dimension you are viewing. For some members, certain commands like **Move** or **Delete** are disabled.

Viewing Ancestors

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

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

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. Select a member on the **Edit Member Properties** grid.
3. Click **Actions**, and then select **Show Ancestors**.

> **Tip:**
>
> Alternatively, you can view a member's ancestors by clicking .

Showing Member Usage in an Application

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

To view where members are used in an application:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. Select a member on the **Edit Member Properties** grid.
3. Click **Actions**, and then **Show Usage**.

> **Tip:**
>
> Alternatively, you can click to view where a member is used in the application.

Focusing Your Editing

Use zoom in, zoom out, keep selected, remove selected, and freeze to focus your editing while working in the **Edit Member Properties** grid.

To perform these operations while viewing the **Edit Member Properties** grid:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. To focus your editing on specific members in the grid, select a member, and then click one of the following zoom operations:
   - Zoom in options:
     - : **Zoom in Next level** displays all members one level below the selected member.
     - : **Zoom in All levels** displays all descendant members below the selected member.
     - : **Zoom in Bottom level** displays all descendant members of the selected member that have no children.
   - : **Zoom Out** displays the member one level above the selected member.
3. To focus your editing on specific rows or columns in the grid, select a row or column, and then choose from the following operations:

- : **Keep Selected** displays only the selected row or column on the grid.
- : **Remove Selected** removes the selected row or column from the grid.
- : **Freeze** (for columns only) keeps the selected column and all columns to the left of the selected column stationary so the column or columns can't be scrolled. Only columns to the right of the frozen column can be scrolled. For example, you can freeze the first column that includes the member name so that you can scroll and edit that member's properties and still see the member name. To unfreeze columns, click **Freeze** once again.

> **Tip:**
>
> Alternatively, you can select these operations on the **Actions** menu. You can also view some of these options by right-clicking on a cell in the grid.

Finding Members

To find dimension members in the **Edit Member Properties** grid:

1. View **Edit Member Properties** for a dimension.

   See Accessing the Simplified Dimension Editor.
2. For **Search**, select **Name**, **Alias**, or **Both**, then enter the search text (member name, alias, or partial string) for which to search.
3. Select one of the following search options:
   - : **Search Up**
   - : **Search Down**
   - : **Filtered Search** searches for all members that match the search term and returns a list of members. Click **Clear Filter** to display the unfiltered grid again.

Moving Members

To move members:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. Select the member to move.
3. To move the member up one position, click . To move the member down one position, click .

> **Tip:**
>
> Alternatively, you can click **Actions**, then select **Move Up** or **Move Down** to move a member up or down one position in the hierarchy. You can also right-click on a cell in the grid to view the **Move** options.

Sorting Members

You can sort members in ascending or descending order, by children or descendants. Sorting members affects the outline. For example, you can alphabetize a list of child members by sorting in descending order to help users quickly locate members in hierarchies in the **Edit Member Properties** grid.

To sort members using the **Edit Member Properties** grid:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. On the grid, select the member whose children or descendants you want to sort.
3. For **Sort**, select **Children** or **Descendants**.

> **Note:**
>
> - If you do not see the **Sort** options at the top of the page, click ».
> - Sorting by children affects only members in the level immediately below the selected member. Sorting by descendants affects all descendants of the selected member.

4. Click  or  .

Moving Members to Another Hierarchy

To move members to another hierarchy in the **Edit Member Properties** grid:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. In the **Parent Member** column of the grid, type a new parent name for the member you want to move.
3. Click **Save**.

Working with Member Formulas

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

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

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

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

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.

2. In the **Default Formula** column of the grid, select the member for which you want to define or edit a formula. Define or edit the formula for the member using one of the following options:

   - Click within the formula bar above the dimension editor grid, and then enter or edit the formula.
   - Click  next to the formula bar, and then enter or edit the formula.

   Note:

   - Double-clicking the formula cell used to display a multi-line popup with **OK** or **Cancel**. We've removed the popup, and instead it's a direct edit within the formula cell.
   - The formula bar displays when a formula cell is highlighted, and is hidden when you navigate away from the formula cell.
   - We recommend you edit formulas using the formula bar or using the script editor that opens when you click  .

   Tip:

   - If you don't see the **Default Formula** column, you'll need to add it to the grid. Click  to open the **Select Columns** dialog, expand **Member Formula**, select **Default Formula**, and then click **OK**. After selecting it, the **Default Formula** column will be included in the grid for all subsequent sessions.
   - To include member names in formulas, keep the focus on the formula cell in the grid. Press Ctrl while clicking the member name you want to include in the formula. The member name will display in the formula bar.

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

Copying Member Names from Microsoft Excel

To copy and paste member names from Microsoft Excel:

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

Editing Member Properties

Related Topics

- Accessing Edit Member Properties
- Adding Members
- Editing Members
- Deleting Members
- Adding Shared Members
- Assigning Access to Dimension Members

Accessing Edit Member Properties

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

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

> **Note:**
>
> The **Cube** drop-down list defaults to **All**. Click the down arrow to the right of the **Cube** drop-down list to filter the list of dimensions by cube.

3. Click the  tab.
4. To edit member properties in the **Edit Member Properties** grid, click within a grid cell to edit text or to view a drop-down menu from which you can choose member properties. You can also drag and drop member property values over to rows and columns to fill in properties that are the same. See the Member Properties table below for property options and descriptions.

   See Working in the Edit Member Properties Grid.

> **Note:**
>
> Each column in the **Edit Member Properties** grid represents a member property. The list of properties (columns) that initially displays on the grid can be different based on which dimension type you're editing. You can customize the layout of columns in the rid by hiding, unhiding, or resizing columns. You can also display the complete set of properties (all columns) by selecting **Select All** in the **Select Columns** dialog box.
>
> To customize the column layout in the **Edit Member Properties** grid, see Customizing the Column Layout.

Member names must conform to guidelines listed in Naming Restrictions.

Shared members must be consistent with Adding Shared Members.

Table 9-6 Member Properties

Property	Value
Member Name	A name that is unique across all dimension members
Parent Member	If the member has a parent in the hierarchy, the parent name.
Description	**Optional**: A description of the member (you can use the same description for multiple members) **Note:** Editing the description of a seeded member is not allowed. Descriptions for all other members added by users can be edited.
Alias Table	**Optional**: The alias table which stores the alias name. Enter an alternate name for the member in **Alias**. See Administering Alias Tables.
For Account members only: **Account Type**	Select **Expense**, **Revenue**, **Asset**, **Liability**, **Equity**, or **Saved Assumption**. For descriptions, see Account Types.
For Account members only: **Variance Reporting**	If the account type is **Saved Assumption**, select **Expense** or **Non-Expense**. Designate the saved assumption as a revenue, asset, liability, or equity account.
For Account members only: **Time Balance**	Select **Flow**, **First**, **Balance**, **Average**, **Fill**, **Weighted Average - Actual_Actual**, or **Weighted Average - Actual_365**. For descriptions, see Time Balance Property.
For Account members only: **Skip**	If the account type is **Asset**, **Equity**, or **Liability**, select **None**, **Missing**, **Zeros**, or **Missing and Zeros**. For descriptions, see Setting Account Calculations for Zeros and Missing Values.

Table 9-6 (Cont.) Member Properties

Property	Value
For Account members only: **Exchange Rate Type**	Select **Average**, **Ending**, or **Historical**. For descriptions, see Data Type.
Data Type	Select **Percentage**, **Date**, or **Text**. For descriptions, see Data Type.
For Account members only: **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	Available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be stored. For members with a stored hierarchy type, the only valid cube aggregation options are Addition or Ignore. In a stored hierarchy, the first member must be set to Addition. For members with a dynamic hierarchy type, all cube aggregation options are valid. Stored hierarchy members that are not children of Label Only members must have Addition set as the consolidation operator. Children of Label Only members can be set to Ignore.
Data Storage	The data storage property. Never Share is the default for new custom dimension members (except root members).
Two Pass Calculation	Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with **Dynamic Calc** or **Dynamic Calc and Store** properties.
Cube Consol op	Select an aggregation option for each cube: • Not used for Cube (member will be invalid for cube) • Addition • Subtraction • Multiplication • Division • Percent • Ignore • Never (member doesn't aggregate, regardless of hierarchy) You can select a source cube only if multiple cubes are valid for the member. Only cubes and aggregation options for which the member's parent is valid are available. If the parent isn't valid for a cube or aggregation option, neither is the child member. Deselecting a cube for an account or entity parent member deselects it for all descendents of that parent. For members with a stored hierarchy type, the only valid aggregation options are Addition or Ignore. You can set usage by cube for members of custom dimensions and the Period dimension, similar to the Account and Entity dimensions. **CAUTION!** Deselecting a cube for dimension members after data is entered into an application may result in loss of data when an application is refreshed. For account members, data is lost if the deselected cube is the source cube.

Table 9-6 (Cont.) Member Properties

Property	Value
For Account members only: **Source Cube**	Select the source cube for the member. A shared member is a pointer to the base member and isn't stored; this is disabled for shared members. The source cube of a shared Account member matches the source cube of the base member, even though Source Plan field is unavailable because it doesn't apply to shared members.
Smart Lists	**Optional**: Select a Smart List to associate with the member.
Solve Order	Specifies the order in which formulas are evaluated. Enter a whole number between 0 and 100000. 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. **Solve Order** is available for aggregate storage cubes and also for block storage cubes that are enabled for Hybrid. Solve Order for block storage cubes is only editable using the Simplified dimension editor.
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.
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. (A Service 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. If a Service Administrator changes these settings, they affect only future dynamic members; they don't retroactively affect dynamic members.

Adding Members

Members must conform to guidelines listed in Naming Restrictions.

To add shared members, see Adding Shared Members.

To add members in the **Edit Member Properties** grid:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.

2. Add members:

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

> **Note:**
>
> Child members inherit the dimension properties of the parent member.

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

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

3. To set or change member properties, click a cell in the **Edit Member Properties** grid and make updates. See Editing Members.
4. To undo all changes that were made since the last save, click .
5. To save your changes, click **Save**.
6. To apply your changes across the application after a save, click .
7. After creating a dimension member, you typically complete these tasks:
   - Assign access. Click **Actions**, and then click **Assign Permission**.
   - Assign custom attributes. See Editing Dimension Properties.

Editing Members

You can edit members directly in the **Edit Member Properties** grid.

To edit members:

1. View **Edit Member Properties**.

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

> **Note:**
>
> To view all property columns in the grid, right-click any heading on the grid, select the **Select All** option in the **Select Columns** dialog box, then click **OK**.

- To delete members, see Deleting Members.

3. To undo all changes that were made since the last save, click .
4. To save your changes, click **Save**.
5. To apply your changes across the application after a save, click .

Deleting Members

Each data value is identified by a set of dimension member values and a cube. Deleting dimension members or deselecting the cube results in data loss when refreshing an application.

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

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

To delete members:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. In the **Edit Member Properties** grid, select the member to delete.
3. Click **Actions**, and then click **Delete Member**.

> **Note:**
>
> Deleting a base member also deletes its shared members.

4. In the confirmation box, click **Yes**.
5. Update and validate business rules and reports.

Adding Shared Members

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

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

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

> **Note:**
>
> Shared members must have the same names as their base member. Also, base and shared member names are case-sensitive.

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

To add shared members to the **Edit Member Properties** grid:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. Add shared members:
   a. Add a child member with the same name as the base member.
   b. For **Parent Member** of the new member, type a new parent name.
   c. For **Default Data Storage** of the new member, select **Shared**.
3. Click **Save**.

Data storage properties for all other cubes will automatically be set to **Shared**.

Assigning Access to Dimension Members

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

To assign access to dimension members:

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. Select the member, click **Actions**, and then click **Assign Permission**.
3. Click **Permissions**.
4. See the following topics to add, edit, or remove permissions:
   - Adding, Editing, and Removing Access to Members
   - About Effective Access Permissions to Shared Members

Adding, Editing, and Removing Access to Members

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

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

1. View **Edit Member Properties**.

   See Accessing the Simplified Dimension Editor.
2. Select the member, click **Actions**, and then click **Assign Permission**.
3. Click **Permissions**.
4. To add access, click **Add User/Group**, then click  to select the users and groups to access the selected member.

   Click **Users** to display all users; click **Groups** to display all groups.
5. To modify the type of access:
   a. Select a **Permission** for the displayed users or groups:
      - **Read**
      - **Write**
      - **None**
      - **Display**

      For descriptions of the preceeding options, see Accessing Ancestor Members in Ad Hoc Grids.
   b. **Optional**: Select a relationship.

      For example, select **Children** to assign access to the children of the selected member.
6. To remove access, click  .
7. Click **OK**.
8. Click **Close**.

About Effective Access Permissions to Shared Members

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

The application checks access permissions at each level, first by user, then by group, based on the member's access permissions inheritance relationship. If multiple access permissions exist, the least restrictive access permission is applied (for example, Write access takes precedence over Read access).

Sample Parent and Child Entity Members

Table 9-7 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.

Editing Dimension Properties

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

1. From the Home page, click **Application**, and then click **Overview**.
2. Click **Dimensions**.
3. The **Cube** drop-down list defaults to **All**. Click the down arrow to the right of the **Cube** drop-down list to filter the list of dimensions by cube.
4. Click the name of the dimension that you want to view.
5. Select the **Edit Dimension Properties** tab  .

Dimension properties must conform to guidelines listed in Naming Restrictions.

Table 9-8 Dimension Properties

Property	Value
Dimension	Enter a name that is unique across all dimensions.
Description	**Optional**: Enter a description.
Alias Table and **Alias**	**Optional**: Select an alias table. Enter an alternate name for the dimension. See Administering Alias Tables.
Cube	Select the cubes for which the dimension is enabled. Clearing this option disables all members of the dimension for the deselected cube.
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.
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. See Assigning Access to Dimension Members.

Table 9-8 (Cont.) Dimension Properties

Property	Value
Data Storage	Select a data storage option. The default is **Never Share**. • **Store**: Stores data values of members. • **Dynamic Calc and Store**: Calculates data values of members, and stores values. • **Dynamic Calc**: Calculates data values of members, and disregards the values. • **Never Share**: Prohibits members in the same dimension from sharing data values. • **Label Only**: Has no data associated with the member. • **Shared**: Allows members in the same dimension to share data values. See 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.
Hierarchy Type	Available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be **Stored**. For members with a **Stored** hierarchy type, the only valid cube aggregation options are **Addition** or **Ignore**. In a stored hierarchy, the first member must be set to **Addition**. For members with a **Dynamic** hierarchy type, all cube aggregation options are valid. **Stored** hierarchy members that are not children of **Label Only** members must have **Addition** set as the consolidation operator. Children of **Label Only** members can be set to **Ignore**.
Custom Attributes	Click to **Create** or **Synchronize** custom attributes for a dimension.

Working with Attributes

Attributes describe characteristics of data, such as the size and color of products. You can use attributes to group and analyze members of dimensions based on their characteristics. For example, you can analyze product profitability based on size or packaging, and you can make more effective conclusions by incorporating market attributes, such as the population size of each market region, into your analysis.

You can assign attributes to sparse dimensions only. You can't assign attributes to label-only members. Attribute dimensions don't have aggregation properties because parents are dynamically calculated.

The Account dimension is usually defined as dense, so you can't assign attributes to it unless it's changed to sparse for all cubes. If you change a dimension from sparse to dense, all attributes and attribute values for that dimension are automatically deleted.

Attributes can have data types of text, date, Boolean, and numeric, as described in Understanding Attribute Data Types.

Attribute names must conform to guidelines listed in Naming Restrictions.

To create or edit attributes in the Simplified dimension editor:

1. From the Home page, click **Application**, click **Overview**, and then click **Dimensions**.
2. The **Cube** drop-down list defaults to **All**. Click the down arrow to the right of the **Cube** drop-down list to filter the list of dimensions by cube, and to see which cubes are marked as **Dense**.
3. Select a sparse dimension for the attribute.

> **Note:**
>
> Only sparse dimensions can contain attributes.
>
> See About Sparse and Dense Dimensions.

4. Select the **Edit Dimension Properties** tab  .
5. Select options:
   - To create an attribute, next to **Custom Attributes**, click **Create**. Type an attribute name, and select a data type: **Text**, **Date**, **Boolean**, or **Numeric**.

     > **Note:**
     >
     > – You can't modify the type after the attribute is created.
     > – Before working with date attributes, you must select the **Attribute Dimension Date Format** in **Application Settings** and save it.

     See Understanding Attribute Data Types.
   - To modify attributes, under **Custom Attributes**, click ••• next to the attribute you want to modify, select **Edit**, and then update the attribute name.
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.

> **Tip:**
>
> Instead of explicitly filtering by an attribute (like Red, for instance), you can create a user variable for the attribute dimension, and then use the user variable as the filter. Then you can enable the user variable as a dynamic user variable which would allow users to change the value of the filter at runtime. This is a useful technique that allows for dynamic filtering.
>
> See Managing User Variables.

Deleting Attributes

When you delete an attribute, all attribute values associated with the attribute are also deleted. Attribute values are removed from members to which they had been assigned, and the attribute is removed from dimensions to which it was assigned.

To delete attributes:

1. From the Home page, click **Application**, click **Overview**, and then click **Dimensions**.
2. Click the name of the dimension with the attribute you want to delete, and then click the **Edit Dimension Properties** tab.
3. Under **Custom Attributes**, click
   •••
   next to the attribute you want to edit, and then select **Delete**.
4. Click **Yes**.
5. Update and validate business rules and reports.

Working with UDAs

You can use user-defined attributes (UDAs), descriptive words or phrases, within calc scripts, member formulas, reports, and forms. UDAs return lists of members associated with the UDA. For example:

- 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.
- When designing forms, you can use a UDA to select members for forms based on a common attribute. When you create forms with UDAs, any members that are assigned to the UDA are dynamically added to the form. For example, if you create a UDA called New Products and assign this UDA to the new products in the Product dimension hierarchy, the form will automatically display the new products at runtime. When selected in the form designer, a UDA is preceded by `UDA`; for example, `UDA(New Products)`.
- 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 business process and create the member formula with a UDA. The syntax for this UDA is: `(UDAs: HSP_UDF)`.
- If you use the `@XREF` function to look up a data value in another database to calculate a value from the current database, you can add the `HSP_NOLINK` UDA to members to prevent the `@XREF` function from being created on all cubes that are not the source cube selected for that member.

UDAs are specific to dimensions. For example, creating a UDA for an Account member makes it available for non-shared Account members. Deleting it removes it for all Account members. To make UDAs available for multiple dimensions, create the same UDA for multiple dimensions. For example, create a UDA named New for Account and Entity dimensions to make the UDA named New available for Account and Entity members.

To work with UDAs in the **Edit Member Properties** grid:

1. From the Home page, click **Application**, click **Overview**, and then click **Dimensions**.
2. Click on the name of the dimension for whose members to associate the UDA.

3. Select the **Edit Member Properties** tab.
4. On the dimension grid, right-click any column heading, select **UDA**, and then click **OK** to make the UDA column visible in the grid.
5. Locate the member to add the UDA, and then scroll to the UDA column.
6. Click in the UDA cell to display the **UDA** management dialog.
7. Select a task:
   - To create a UDA, click
     
     , enter a name, and then click **Save**.

     > **Note:**
     >
     > Use no more than 60 characters when naming UDAs.

   - To modify a UDA, select the UDA, click
     
     , change the name, and then click **Save**.
   - To delete a UDA, select the UDA, click
     
     , and then confirm deletion.

     > **Note:**
     >
     > Deleting the UDA removes it for the whole dimension. If you delete UDAs, you must update all member formulas, calc scripts, and reports that reference them.

   - To clear UDA selections, click
     
     .
8. To add or remove UDAs for the member, use the arrows to move UDAs to and from the **Selected UDA** panel.
9. Click **OK**.

Working with Attribute Values

Attribute values provide users with another way to select dimension members when using forms. Data values for attributes are dynamically calculated but not stored.

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.

Creating Attribute Values

To create attribute values:

1. On the Home page, click **Application**, and then click **Overview**.
2. Click the **Dimensions** tab.
3. For **Cube**, select **All**.
4. Select an attribute dimension.

> **Note:**
>
> Attribute dimensions are listed at the end of the dimensions listing and at the bottom of the dimension drop-down.

5. In **Edit Member Properties**, add members:
   - To add a child member, select the parent level member, click **Actions**, and then click **Add Child**.

> **Note:**
>
> Child members inherit the dimension properties of the parent member.

   - To add a sibling member, select a member, click **Actions**, and then click **Add Sibling**.
6. To set or change member properties, click a cell in the **Edit Member Properties** grid and make updates. See Editing Member Properties.
7. To save your changes, click **Save**.

Assigning Attribute Values to Base Dimensions

When an attribute dimension is associated with a base dimension, it displays in the Simplified Dimension Editor in the same way as a regular dimension. To view attribute dimensions in the listing, select **All** in the **Cube** drop-down list on the main dimension page. Then you can select the attribute dimension to add values to the grid.

To assign attribute values to base dimensions:

1. Open the **Edit Member Properties** page for the base dimension and find the column with the name of the attribute dimension.
2. Click within the grid for a specific member to display a drop-down list with attribute values, and then select an attribute value from the list.
3. Click **Save** on the **Edit Member Properties** page to assign the value to the base member.

10

Designing Dashboards

Design dashboards that show summary data to users. Dashboards enable users to chart, evaluate, highlight, comment on, and even change key business data, as well as add and edit rules.

Related Topics

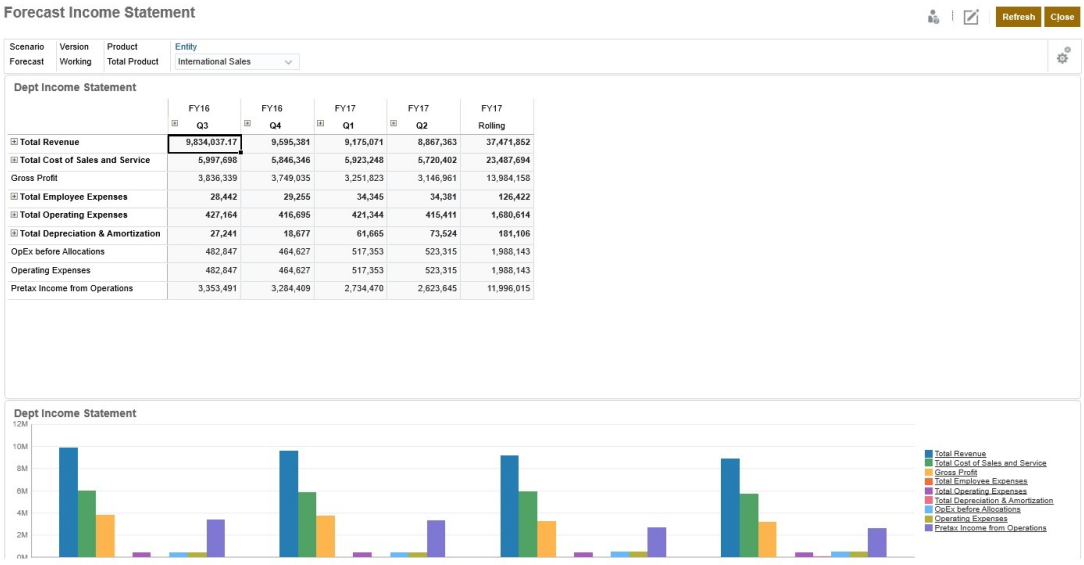
- The Power of Dashboards
- Creating and Managing Dashboards
- About Dashboard Versions
- Viewing Dashboards
- Converting Dashboard 1.0 Dashboards to 2.0
  Learn how to convert Dashboard 1.0 dashboards to Dashboard 2.0 dashboards.
- Creating and Managing Dashboards 1.0
- Concepts in Designing 1.0 Dashboards
- Creating Dashboard 1.0 Dashboards
- Creating Dashboards Containing Master Forms and Details
- About Your Dashboard's Layout - Version 1.0
- About the Gauge Chart Type
- About the Tile Chart Type
- Customizing Dashboard Colors
- About Global and Local POVs in 1.0 Dashboards
- Dashboard POVs and Valid Intersections
- Creating and Managing Dashboards 2.0
- Concepts in Designing Dashboard 2.0 Dashboards
- Object Palette
- Dashboard Toolbar
- Dashboard Workspace
- Properties Panel
- Dashboard Components
- Considerations for Dashboard 2.0
- Creating Dashboard 2.0 Dashboards
- Working with the Member Selector When Designing Dashboards 2.0
  When designing Dashboard 2.0 dashboards, you can easily search for and select new dimension members using the member selector.
- About Forms 2.0 Grids in Dashboard 2.0
- About the Geomap Chart Type

- About the Pyramid Chart Type
- About the Waterfall Chart Type
- Using Tables in Dashboards
- About the Gauge Chart Type for Dashboard 2.0
- About the Radar Chart Type for Dashboard 2.0
- About the Combination Chart Type for Dashboard 2.0
- About the Tile Chart Type for Dashboard 2.0
- About Global and Local POVs in Dashboard 2.0
- About Quick Analysis

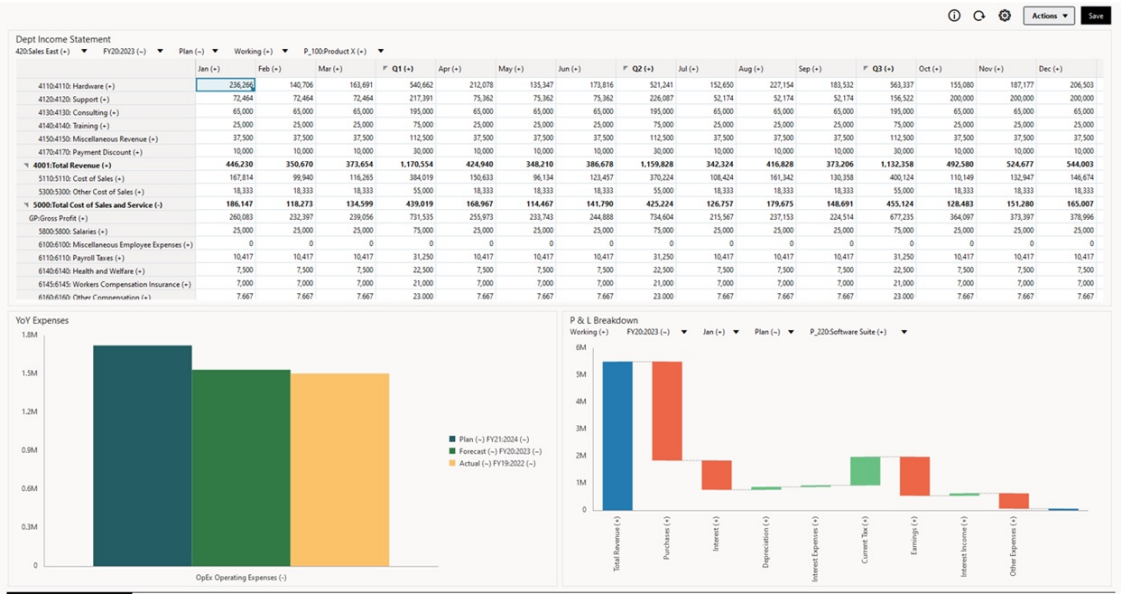
The Power of Dashboards

Dashboards typically provide an overview to users at the beginning of their planning and forecast process by showing summary data. The versatility of dashboards enables users to chart, evaluate, highlight, comment on, and even change key business data. They can change a driver in a form that's in a dashboard and immediately see its impact in other forms and charts:

Sample Dashboard 1.0



Sample Dashboard 2.0



You create dashboards by simply dragging and dropping a variety of objects from the library panel on the left to the dashboard workspace.

With dashboards, you can:

- Include multiple forms that dynamically update, including their associated charts, as users change data in the form.
- Include tiles that display a specific value from the cube. To provide the value for each tile, you can specify a form or a cell intersection as a data source.
- Switch easily between using the design environment and the runtime environment, so you can see exactly what the dashboard user will see.
- Depict data interactively with a wide variety of chart types such as area, bar, bubble, column, combination bar and line, doughnut, funnel, gauge, scatter, radar, and so on.
- Control the dashboard's layout.
- Depending on the form design, enable users to drill down into underlying detail and select which members to work with.
- Include user variables in the global POV bar and the local POV.
- In certain charts, customize the colors and line width, and show or hide gridlines.
- Add links to dynamically display external Web pages.
- Include explanations of data called *Commentary* in the dashboard.
- Tag a form as master and then filter members from the master form to detail forms or charts within the same dashboard.
- Change legend and label positions in charts.
- In Dashboard 2.0 (only available with Redwood Experience):
  - Add up to 12 components within a dashboard
  - Add up to 10 tiles in a tile chart type
  - Add hierarchical labels in charts

- Use a logarithmic scale in relevant chart types
- Add a secondary Y-axis in relevant chart types
- Render the dashboard in Default, Light, and Dark background colors using the Style property
- Use grid display with write-enabled grids (used with Forms 2.0)
- Enjoy other usability improvements such as more screen space, options to open and edit a form directly, option to edit a dashboard without data, automatically apply POV changes, and hide dimension names
- Improved Rules on Save with dashboards: with/without runtime prompts before/after Load/Save
- Migrate easily from Dashboards 1.0

> **Note:**
>
> Dashboard 1.0 dashboards are deployed with Forms 1.0, even when the **Forms Version** setting is **Forms 2.0**.

When users use a dashboard (referred to as *runtime*), they can set many aspects of the components, such as the type of chart displayed, the dashboard's title, and so on. A toolbar is available for each component. Dashboard users can change and save data, run rules, and so on. However, changes made in runtime to the chart type options aren't saved for the next session. Clicking **Save** in runtime saves the data, but not the dashboard definition.

Service Administrators create, redesign, delete, and assign permissions to dashboards and dashboard folders.

Related Topics

- About Dashboard Versions
- Concepts in Designing 1.0 Dashboards
- Concepts in Designing Dashboard 2.0 Dashboards
- Creating and Managing Dashboards

Creating and Managing Dashboards

Related Topics

- About Dashboard Versions
- Viewing Dashboards
- Converting Dashboard 1.0 Dashboards to 2.0
  Learn how to convert Dashboard 1.0 dashboards to Dashboard 2.0 dashboards.

About Dashboard Versions

Dashboards are currently available in two versions:

- **Dashboard 1.0**: Uses Oracle Application Development Framework (Oracle ADF) technology

- **Dashboard 2.0**: Uses Oracle JavaScript Extension Toolkit (Oracle JET) technology

Both dashboard versions currently coexist in the same business process at the same time. Dashboard 1.0 uses Oracle ADF technology and the behavior of the Oracle ADF-based dashboards remains unchanged. You can still create and update these dashboards in your business process as you did before. Dashboard 2.0 uses the existing Dashboard functionality and enhances it with Oracle JET technology for faster rendering, sophistication, ease of use, improved flexibility, better visualization, and additional designs. Dashboard 2.0 is available only if **Redwood Experience** is enabled.

To help you differentiate between the dashboard versions on the **Dashboards** listing page, we've provided an icon in front of each dashboard:

- 
  : Dashboard 1.0

  See Creating and Managing Dashboards 1.0.

- 
  : Dashboard 2.0

  See Creating and Managing Dashboards 2.0.

Dashboard 2.0 Designer Features

Dashboard 2.0 Designer supports:

- More chart types than Dashboard 1.0; for example, Waterfall, Pyramid, and Geomap and sub-chart types for Radar, Combination, and Gauge charts are all Dashboard 2.0 features
- Creating tables in dashboards with visual inline views of data, called Spark Charts
- Quick Analysis, which lets you add components to dashboards without having a prebuilt form

  You create an ad hoc query by typing member names in the Search bar or using the Member Selector, and then you control the layout of the query in the Layout tab of the Properties panel.
- Working directly with data in form grids if the underlying form is a Forms 2.0 form
- A URL type of dashboard component
- A Commentary type of dasboard component with a text editor
- A custom formatted dashboard title
- Background colors for URL, Commentary, and Grid component types
- Grid chart type option to render Forms 2.0 inside Dashboards 2.0
- A new **Associations** tab in the Properties panel for associating global rules
- Multiple charts for Gauge, Pie, and Donut chart types
- New menu options for directly opening and editing forms from within dashboard components
- An Information icon for forms and dashboards. Drop any form into a dashboard and see its information.
- The ability to resize the Properties panel
- The ability to set the background color for each dashboard component

- An option to remove the component title, using the saved space for visualizations
- An increased number of dashboard components (up to 12)
- An increased number of tiles (up to 10)
- The ability to edit without data
- A customized cell visualization limit of up to 10,000 cells with any row x column combination, such as:
  - 100 rows x 100 columns=10,000 cells
  - 1000 rows x 10 columns=10,000 cells
  - 250 rows x 40 columns= 10,000 cells
- The ability to select a dashboard **Style**. Options are **Default**, **Light**, and **Dark**.
- Options to show and hide borders between dashboard components and POV bars
- The new Oracle JET member selector
- A surface Designer in Dynamic Tabs

To review considerations before switching to Dashboard 2.0, see Considerations for Dashboard 2.0.

Viewing Dashboards

To view the **Dashboards** page:

1. On the Home page, click **Dashboards**.
2. To open a dashboard, click the dashboard's name.

This opens the dashboard in the *runtime* environment.

> **Note:**
>
> If you've selected a Dashboard 2.0 dashboard and you see an `Error during Dashboard 2.0 component initialization` message, the dashboard has Forms 2.0 forms associated with it. You must ensure that Redwood Experience and Forms 2.0 are enabled before you can view the dashboard.
>
> To enable Redwood Experience, click **Tools**, then **Appearance**, select **Enable Redwood Experience**, and then click **Save**.
>
> To enable Forms 2.0, click **Application**, then **Settings**, and then under **Other Options**, find **Forms Versions**, select **Forms 2.0**, and then click **Save**.

Features of the Dashboards Page

On the **Dashboards** page:

- Folders are supported. The **Dashboards** page uses the same folder hierarchy as infolets and data entry forms and all artifacts and folders reside under a root folder called **Library**. You cannot delete or rename the **Library** folder.

> **Note:**
>
> Only administrators can add files (for example, dashboards, infolets, forms, reports, and so on) to the **Library** root folder.

- To help you differentiate between the dashboard versions on the **Dashboards** page, we've provided an icon in front of each dashboard:
  - 
    : Dashboard 1.0
  - 
    : Dashboard 2.0
- You can toggle between viewing dashboards by a flat view or a tree view:
  
- The **Dashboards** page can be filtered by type. Click  , and then select from the following filter options:
  - **All Types** (default)
  - **Dashboard 1.0**
  - **Dashboard 2.0**
- To search for a specific dashboard, click
  
  to search by keywords or  to narrow the search to specific library folders.

Dashboards Page Actions Column

On the right side of the **Dashboards** page there is an **Actions** column. Clicking on the ••• icon next to the dashboard or folder, displays the following actions:

- Folder Actions:
  - **Assign Permission**: Assigns access permissions to the folder. Folder permissions apply to all items within that folder, even to items that may not be visible in the list.
  - **Create Dashboard**: Creates a Dashboard 1.0 dashboard.
  - **Create Dashboard 2.0**: Creates a Dashboard 2.0 dashboard.
  - **Create Folder**: Creates a new folder.
  - **Delete**: Deletes a folder. The folder must be empty to delete it.
  - **Rename**: Renames a folder. The folder cannot be renamed if it contains other folders.
  - **Move to**: Moves the folder to another location in the folder hierarchy.
  - **Convert All Dashboards to 2.0**: Finds any Dashboard 1.0 dashboards within the folder hierarchy, and converts them to Dashboard 2.0 dashboards.
- Individual Dashboard Actions:

- **Edit**: Opens the dashboard in the Dashboard Designer with the underlying form data. The dashboard opens in a dynamic tab.
- **Edit without Data** (Dashboard 2.0 option only): Opens the dashboard in the Dashboard Designer without the underlying form data so you can more easily perform such actions as realigning the dashboard components and tiles. The dashboard opens in a dynamic tab.
- **Rename**: Renames a dashboard.
- **Copy As**: Copies a dashboard with a new name. The dashboard is copied to the listing directly under the original dashboard.
- **Delete**: Deletes a dashboard.
- **Move to**: Moves a dashboard to another folder.
- **Assign Permission**: Assigns access permissions to the dashboard.
- **Convert to Dashboards 2.0** (Dashboard 1.0 option only): Converts the dashboard to a Dashboard 2.0 dashboard.
- **Copy URL**: Enables you to copy and share the direct URL for a dashboard. See Copying Direct URLs for Artifacts.

Converting Dashboard 1.0 Dashboards to 2.0

Learn how to convert Dashboard 1.0 dashboards to Dashboard 2.0 dashboards.

If your application currently uses Dashboard 1.0, you can convert them to Dashboard 2.0.

You convert dashboards on the **Dashboards** listing page at the individual dashboard level or at the folder level. For example, if you wish to convert all dashboards in the application from 1.0 to 2.0, you can select the top level Library folder and convert them all.

> **Note:**
>
> - After converting to Dashboard 2.0, you cannot convert your dashboards back to Dashboard 1.0.
> - The conversion process does not change the dashboard names. Navigation flows will still work as they did before the conversion.
> - After converting to Dashboard 2.0, dynamic POVs are displayed first instead of fixed POVs.

To convert Dashboard 1.0 to Dashboard 2.0:

1. On the Home page, click **Dashboards**.
2. Navigate to the folder or to the individual Dashboard 1.0 dashboard that you'd like to convert, then select from the following options:
   - To convert an individual dashboard, click ••• to the right of the dashboard, and then select **Convert to Dashboards 2.0**.
   - To convert all dashboards within a folder, click ••• to the right of the folder, and then select **Convert All Dashboards to 2.0**.

Note:

- Very old dashboards might include a dashboard definition with a component width value that is no longer supported. This could result in an unexpected dashboard rendering after converting an old dashboard to Dashboard 2.0.

  To resolve this issue, we recommend any of the following workarounds:

  – Open the original dashboard in Dashboard (1.0), edit the dashboard, and then save it. The dashboard will be automatically saved with the corrected component width values, and then it can be converted to Dashboard 2.0.

  – Create a new dashboard in Dashboard (1.0) with the same set of forms and charts and the same settings, then convert the dashboard to Dashboard 2.0.

  – Create a new dashboard in Dashboard 2.0 with the same artifacts and settings.

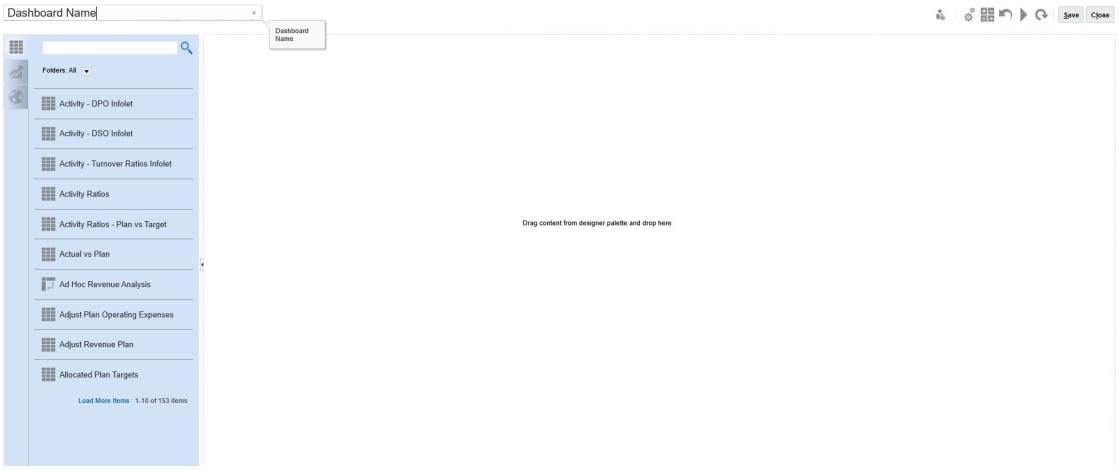
- Dashboards containing master forms and details that were converted to Dashboard 2.0 prior to the 23.11 update might display two **Apply Context** menu options in a grid's context menu. This issue was fixed in 23.11.

Creating and Managing Dashboards 1.0

- Concepts in Designing 1.0 Dashboards
- Creating Dashboard 1.0 Dashboards
- Creating Dashboards Containing Master Forms and Details
- About Your Dashboard's Layout - Version 1.0
- About the Gauge Chart Type
- About the Tile Chart Type
- Customizing Dashboard Colors
- About Global and Local POVs in 1.0 Dashboards
- Dashboard POVs and Valid Intersections

Concepts in Designing 1.0 Dashboards

Helpful information as you design 1.0 dashboards:



- On the left is the design palette. Simply drag and drop objects from the palette to the canvas.

> **Tip:**
>
> Drag an object to a border line. The drag icon changes to a plus sign when you can drop the object in an allowed space. See About Your Dashboard's Layout - Version 1.0.

- On the top right are settings for the entire dashboard:

- Click **Settings**

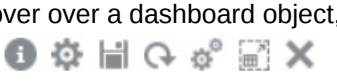
to set these general aspects of the dashboard:

Table 10-1 Dashboard General Settings

Option	Description
Use name as title	By default, this option is selected. Clear this option to give the dashboard a title with custom formatting.
Borders	When you create a new dashboard, borders are hidden by default. To show borders in newly-created dashboards, select **Show**.
Layout	Select **Fixed** (default) or **Flexible**.
POV Bars	Select **Show** (default) or **Hide**.
Global POV Bar	Select **Enable** (default) or **Disable**.

For information about POVs, see About Global and Local POVs in 1.0 Dashboards.

- When you hover over a dashboard object, a toolbar for that object displays in the upper right corner:



> **Note:**
>
> Dashboard (1.0) hover icons provide options such as **Instructions** (available only if instructions are configured for a form), **Actions**, **Save**, **Refresh**, **Settings**, and **Maximize**, depending on the type of object.

- As you create a dashboard, click **Runtime** so that you can immediately see how the dashboard looks and works to a dashboard user. To return to designer mode to continue designing the dashboard, click .
- By default, missing or suppressed data is plotted as zeros in graphs. You can clear the **Plot Missing Values as Zero** setting to ignore missing or suppressed data in certain chart types so it's no longer plotted as zeros.
- The listing page for dashboards supports folders. Folders enable you to assign permissions to all dashboards within a folder rather than assigning permissions to each individual dashboard. The dashboard listing page uses the same folder hierarchy as infolets and data entry forms and all artifacts and folders reside under a root folder called **Library**.

> **Note:**
>
> Only administrators can add files (for example, dashboards, infolets, forms, reports, and so on) to the **Library** root folder.

- On the listing page for dashboards, you can toggle between viewing dashboards by a flat view or a tree view:

  Then you can search for dashboards using **Search** . The flat view displays only the dashboards that meet the search criteria, not the folders that contain them. The tree (or hierarchical) view displays dashboards in the context of the folders that contain them.

  To search on another keyword, clear the search criteria by clicking × in the **Search** box.

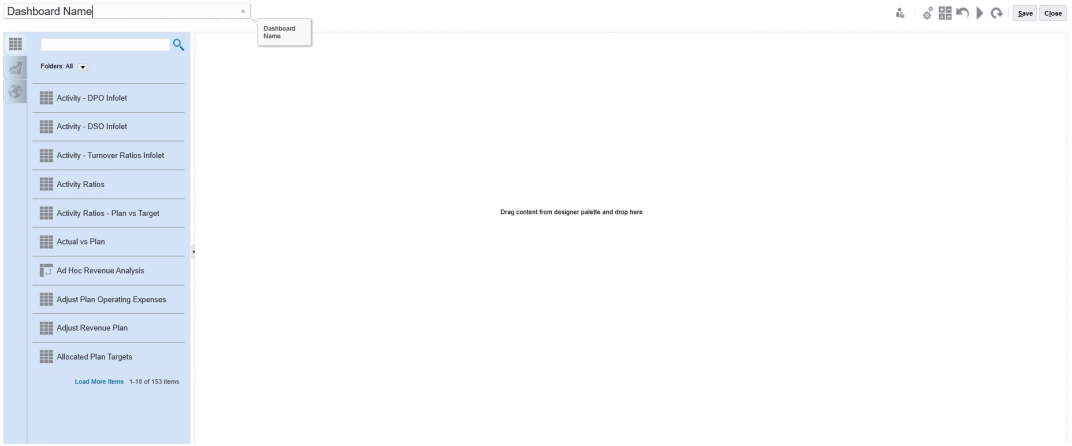
Creating Dashboard 1.0 Dashboards

Need help deciding which dashboard version to choose? See About Dashboard Versions.

> **Note:**
>
> To create a Dashboard 2.0 dashboard, see Creating Dashboard 2.0 Dashboards.

1. From the Home page, click **Dashboards**, and then click **Create**.

2. From the options provided, select **Dashboard**.



3. Enter a name by clicking the default dashboard name and entering a new name in the input box.

   You can give the dashboard a title with custom formatting, click **Settings**, clear **Use name as title**, and then enter the title and set formatting the dialog box.

4. From the design palette on the left, drag and drop objects onto the dashboard canvas.

   Select from these objects:

Table 10-2 Dashboard Objects

Object	Description
Forms	Select forms to include in the dashboard by navigating to the forms folders or by searching for them by name. To view instructions for forms after they've been added to the dashboard, hover over the form and click **Instructions**  . The access permissions set for forms are honored in dashboards.

Table 10-2 (Cont.) Dashboard Objects

Object	Description
Chart Types	Select the chart types to include in the dashboard. When first added, a selected chart has sample data. You then associate it with a form as its data source. When you link a chart to a form, users can immediately see the impact of changing data in the form on the associated charts. By default, missing or suppressed data is plotted as zeros. For selected chart types (Area, Bubble, Combination, Line, Radar, and Scatter), you can disable this setting by clearing the **Plot Missing Values as Zero** option in the chart's settings. When this option is cleared, the missing or suppressed data is ignored and is no longer plotted for those chart types. The Combination chart type alternates displaying row data with vertical bars and lines in the chart. For example, the data in row 1 of a form is displayed as a bar and the data in row 2 as a line, with alternating chart types for even and odd-numbered rows. Although the Combination chart type can display up to 20 rows of data, it's particularly useful when you want to compare two categories of data. For example, you want to compare Germany and France's average exchange rates over several years, so the form has Germany rates in row 1 of the form, and France's rates are in row 2. For information on the Gauge chart type, see About the Gauge Chart Type. The Tile chart type, sometimes called *performance tiles*, lets you select specific values from the cube to display. See About the Tile Chart Type.
Commentary	Select **External Artifacts**, and then **Commentary**. Enter text that explains the data or charts. Note that the text limit is 2000 characters, including any formatting tags that are added for rich text.
URL	Dynamic web page summary. Select **External Artifacts**, and then **URL**. Insert only external site URLs starting with the `https://` security protocol. Don't use internal or relative URLs or URLs for unconsenting third party sites such as google.com.

5. Customize the dashboard using the dashboards settings and the objects' hover toolbar, and then click **Save**.

   See About Your Dashboard's Layout - Version 1.0.

You can easily duplicate and modify a dashboard using **Copy As** on the Dashboard list page. Select the Actions icon next to the dashboard you want to copy.

Creating Dashboards Containing Master Forms and Details

You can design dashboards that have one master form and multiple detail (or target) objects (forms or charts). When you do so, the selection of members in the master form automatically filters to the members in the target objects, and the target objects show only the details that are relevant to the members highlighted in the master form.

Considerations when designing dashboards with master forms and target objects:

- Master forms must always be a form (you cannot tag a chart as a master form).
- The target objects can be forms or charts.

- The dashboard containing a master form must always contain more than one object.
- Only one form on a dashboard can be tagged as master. If you want to tag a different form on a dashboard as master, you must first untag the existing master form.

To create a dashboard containing a master form and target objects:

1. Create a dashboard with at least one form and one or more target objects.

   See Creating and Managing Dashboards.
2. Ensure the dashboard is in designer mode by clicking **Actions**, and then clicking **Edit**.
3. Select the form on the dashboard that you want to designate as master, click the Settings icon for that form, and then select **Tag as Master**.

To filter the data in the target forms or charts that is relevant to the data in a master form, right-click the relevant data in the master form and select **Apply Context**.

About Your Dashboard's Layout - Version 1.0

About setting up a dashboard layout in version 1.0:

- The first object you drag occupies the whole canvas.
- Then drag objects to the left, the right, the top, or the bottom of the existing object.
- The dashboard canvas provides two types of drop zones: One to place two objects next to each other, each occupying half the space, and the other to place three objects, each occupying a third of the space.
- You can design a dashboard with objects to display vertically, horizontally, and each can have its own size.
- To resize an object that uses **Flexible** layout, drag the object's border. You can also click an object's handle to expand it, and then click the handle again to restore the object to its original size.
- To resize an object that uses **Fixed** layout, set its width or height percentage in **Settings**.
- The form layout can be asymmetric.
- In runtime mode, if a user doesn't have access to a form or if the form is missing, then an adjacent object takes its space. In designer mode, all empty objects are displayed so that the designer can choose to remove them.

About the Gauge Chart Type

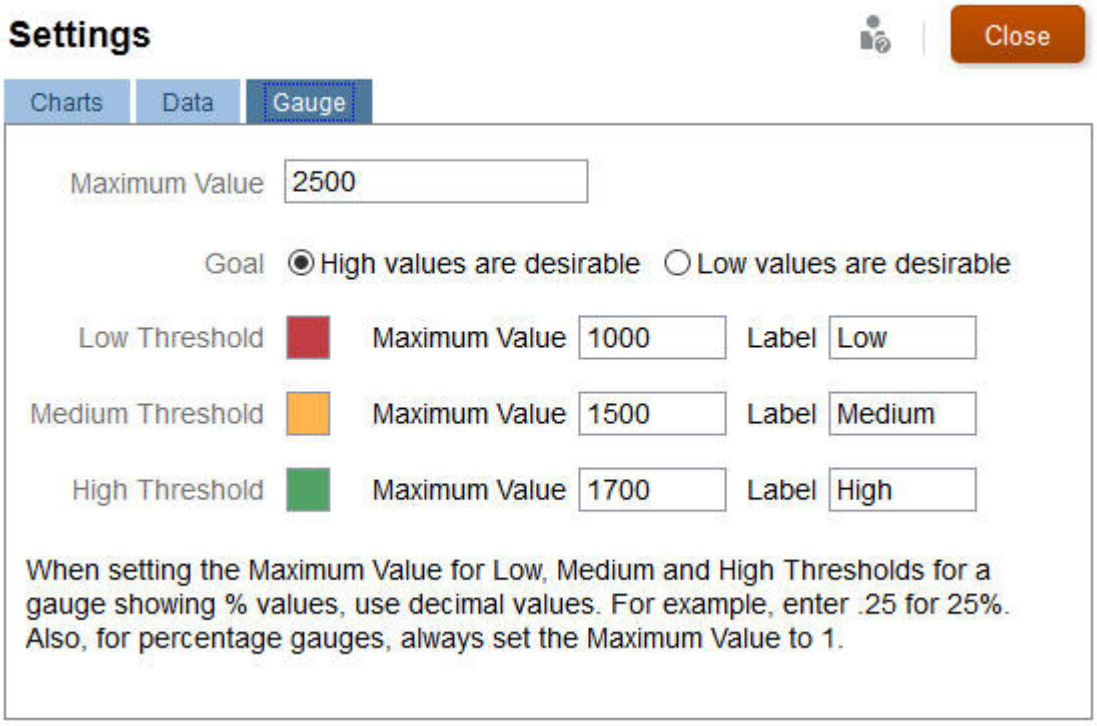
Gauge chart types are handy for showing whether data values fall within an acceptable range or not. You set the maximum value, the range maximums, and the gauge displays ranges as red, yellow, and green to help you quickly assess a current value. So, gauge chart types help you identify problems in important data points or measures. For example, you could use a gauge to display the current sales, where the thresholds are set to represent the sales targets.

If the form has multiple values, you can display multiple gauges, up to a maximum of 36 (the values in the first 6 rows and the first 6 columns in the form). The remaining values in the form are ignored. If you want the gauge chart to display only one value, then associate it with a form that has only one cell value.

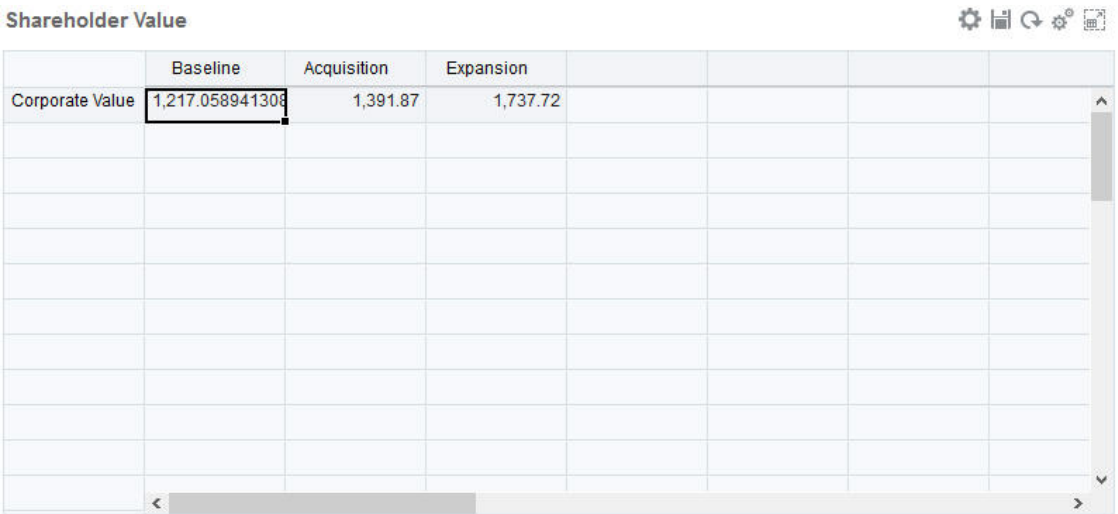
You can select either a dial gauge or a status meter gauge. You can display a status meter gauge using either horizontal or vertical bars.

Dashboard designers can set:

- **Maximum Value**: The highest value on the gauge. The dashboard designer sets the **Maximum Value** as a default, and then users can temporarily change it at runtime. If the dashboard designer doesn't specify a maximum value, the application automatically sets the maximum value as greater than the value on the gauge.
- **Thresholds**:
  - Low, Medium, and High thresholds: To visually indicate whether a measure lies in the acceptable range or not, these thresholds enable you to display the gauge in red, yellow and green based on the specified values.
  - Thresholds where low values are desirable.
  - Appropriate labels for the thresholds that are displayed when hovering over the thresholds in the gauge.

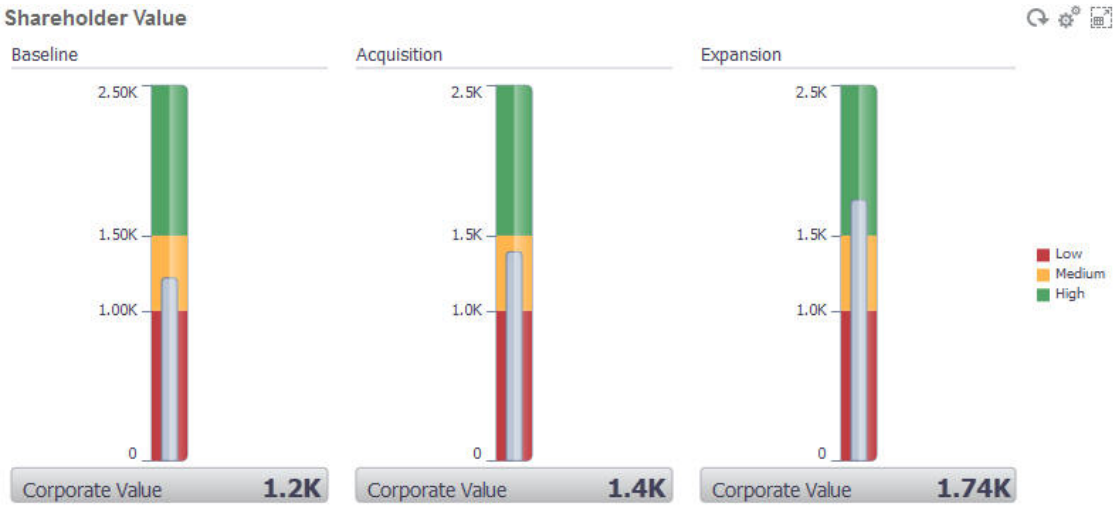


For example, here's a form's data:



Here's the resulting dial gauge:

Here's the resulting status meter gauge with vertical bars:



> **Note:**
>
> If a cell in the form is missing a value, no gauge is displayed for that cell. Also, you must specify at least 2 consecutive thresholds. The application needs the middle threshold value to calculate the chart.

About the Tile Chart Type

A tile is a chart type that lets you select specific values from the cube to display. In addition to using a form as a data source, you can directly enter a cell intersection that provides the value for a tile. You can have up to 6 tiles across, and 4 rows down in a dashboard, and give them a title. Until you associate a tile with data, it displays sample data.

With a form as the data source for a tile:

- You can have up to six tiles per object.
- The values in the first column (up to the sixth row) are used to create the tiles.

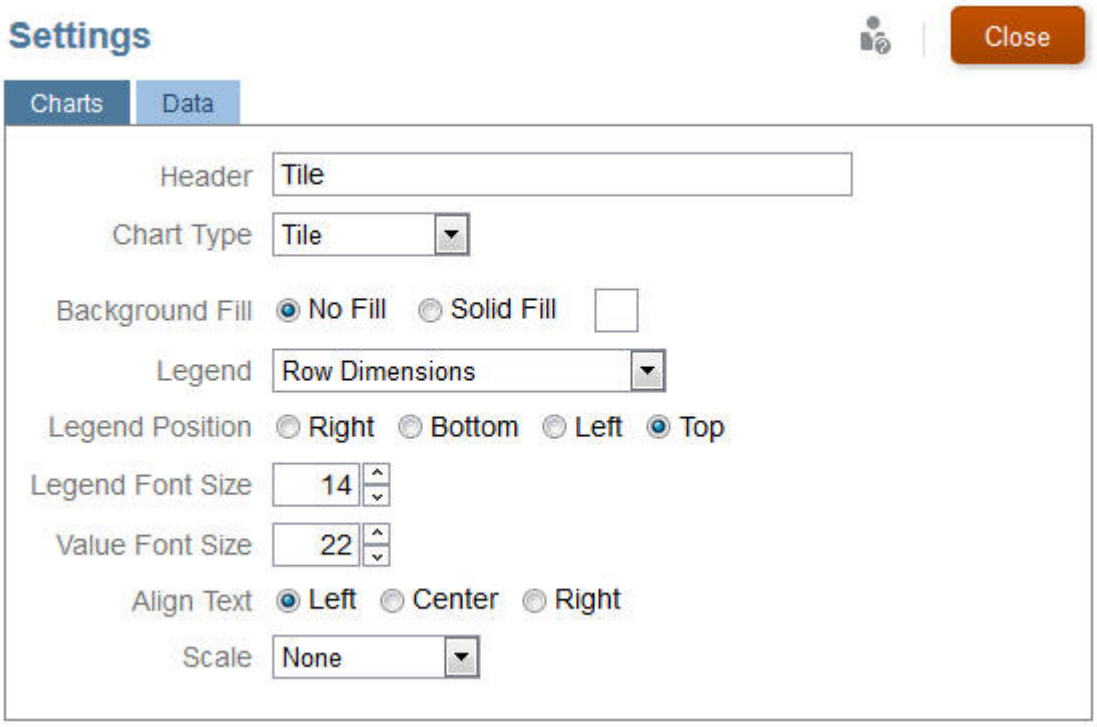
> **Note:**
>
> Sometimes in a form, the first column may be collapsed (hidden) when viewed as a grid. But the collapsed column is still considered when the tile gets its values from the form.

- The tile's title is the row's title, and it gets its values from the first column, then by row.
- You can set the tile's title, the tile's height percentage, legend, and can select which axes from the form to include. For example, if you select a form with three rows, the tile displays three values.

With a cell intersection as the data source for a tile, you can have only one tile per object.

> **Tip:**
>
> To select a tile chart type, expand the list of chart types by clicking the link at the bottom of the list.

Here are options you can set for a tile chart type. Note that you can set whether the displayed value is horizontally aligned in the tile to the left, the center, or the right.



Scaling Large Numbers

Especially useful for large numbers, you can scale how a value is displayed. For example, if the tile value is 1,689,000 and you select **K** as the scaling option, the tile displays the value as 1689K. Your scaling options:

- **None**: No scaling is applied.

- **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T.
- **K**: The value is displayed as thousands units. For example, 1689000 displays as 1689K.
- **M**: The value is displayed as millions units. For example, 123,456,789 displays as 123M.
- **B**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B.
- **T**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.

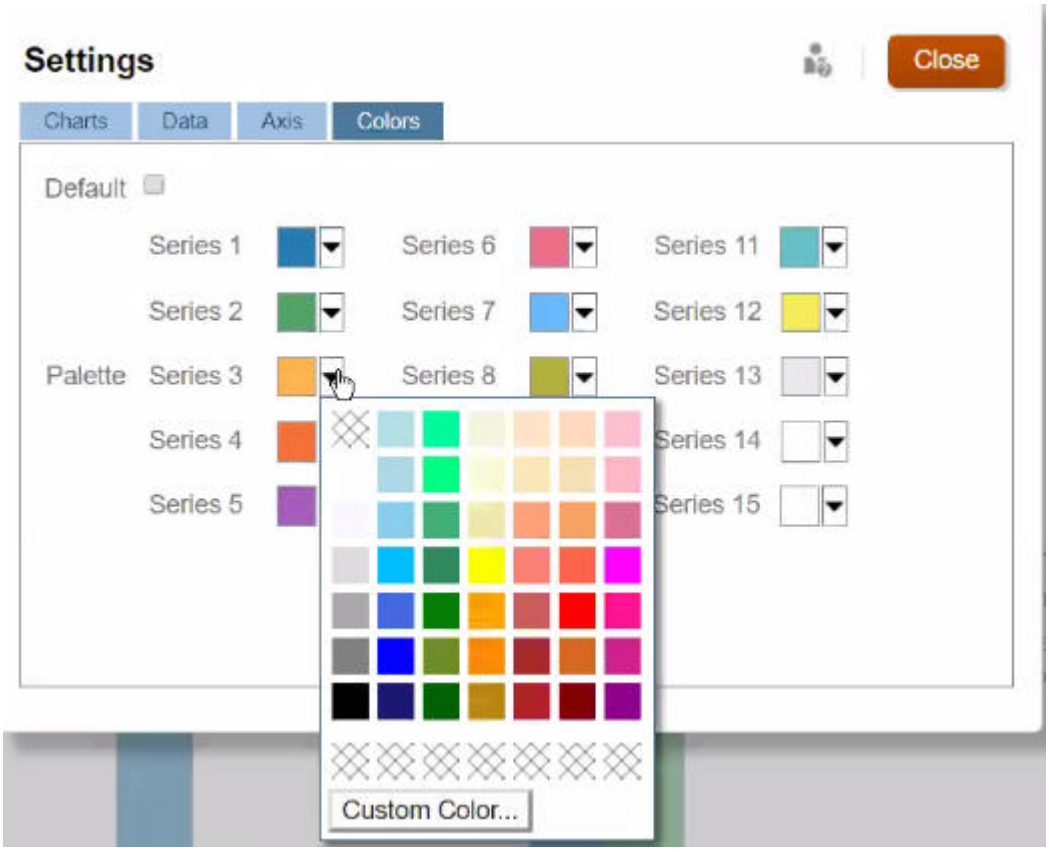
Customizing Dashboard Colors

Maybe your company uses a standard set of colors in charts to denote different types of data. For example, dark blue might represent actual data versus light blue for budget data. When you customize dashboard colors, you select colors in the order of rows on the form. Series 1 represents the first row of data, and so on. You can assign each row in the form a color that represents its data in the chart.

You can customize dashboard colors in Bar, Line, Area, Bubble, Column, Combination, Doughnut, Pie, Radar, and Scatter chart types.

1. With the chart on the dashboard's design palette, click **Settings**  .
2. Click **Colors**.
3. Clear the **Default** check box, and then click the down arrow  for the Series you want to change.

   Check the chart in the background to see the data type that each series represents.



4. Click the colors you want for the selected Series, and then click **Close**.

> **Note:**
>
> To select more shades of colors than the ones initially displayed, click **Custom Color...**.

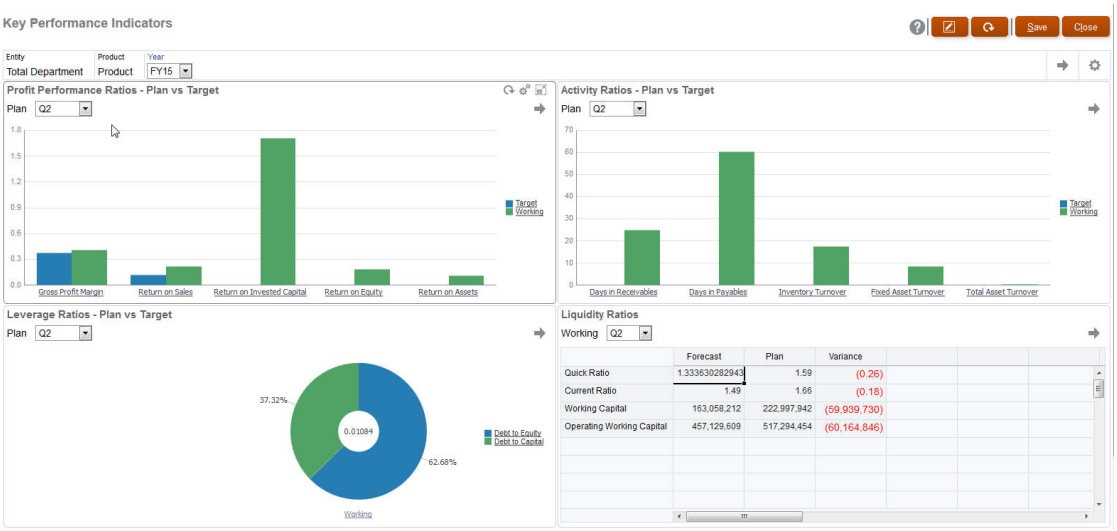
Your selections apply only to the current chart. Follow these steps to change the colors of other charts in the dashboard.

About Global and Local POVs in 1.0 Dashboards

A local POV on a form reflects the dimension members the form designer selected for that form. Dashboards also support *global POV bars*, so that the local POVs that are common are combined in the global POV bar to avoid repeating them in each object.

Dashboard 1.0 POVs Example

Here's a 1.0 dashboard showing a global POV bar (showing Entity, Product, and Year) and a local POV (the Plan drop-down list showing Q2):



With a global POV bar, if you change a page in the global POV bar and then click **GO**, the page changes for all objects that are based on forms. The global POV bar displays at the top of the dashboard above all the objects, while the local POV bar displays within the object. User variables are supported in both global and local POVs in dashboards and you can change dynamic user variables directly from the POV bar.

In dashboard **Settings**, you can set whether to show or hide POVs and whether to enable or disable global POV bars. (If you select **Hide** for **POV Bars**, and **Enable** for **Global POV Bar**, the **Hide** option overrides the **Enable** option.) The global POV bar is enabled by default; if you disable it, the complete POV is displayed for each local POV as applicable.

About global POV bars:

- The POV Bar is made up of local POV dimensions, Page dimensions and user variables.
- Depending on the local POV and page of each form on the dashboard, the global POV bar is automatically calculated.
- They are reflected in the other objects using forms in that dashboard. That is, they apply to forms in a dashboard, to charts that are linked to forms, and to tiles that use forms as a data source. So if the dashboard doesn't include a form as a data source, then neither the local nor global POV bar is available.

Here's an example of how the global POV bar is calculated, based on the local POV dimensions for two forms:

The global POV bar is disabled:

- Form A local POV: Year, Entity, Product
- Form B local POV: Year, Entity, Project

The global POV bar is enabled:

- Global POV bar: Year, Entity
- Form A local POV: Product
- Form B local POV: Project

Because not all dimensions and page selections may be common to all forms on a dashboard, the complete POV for a dashboard object may get split between the local and global POV bar. The global POV bar and the local POV together contain the complete intersection information for each form on a dashboard.

If there is only one object on the dashboard that uses a form as a data source, then the entire POV/page of the form can be moved to the global POV bar.

If there is more than one object on a dashboard that use forms as a data source, then this is how the application determines which dimensions go in the global POV bar or stay in the local POV:

- If the dimension is in the POV or page of all the forms, and the member selection is the same in all the forms, the dimension goes in the global POV bar.
- If the dimension is in the POV on one form and in the page of another form, then the dimension stays in the local POV.
- If the dimension is in the POV, then the same members must be selected in all the forms for the dimension.
- If the dimension is a page dimension, then the selected page members must be the same and display in the same order in all the forms.

The POVs in 1.0 dashboards honor valid intersections by hiding invalid Page members. See Dashboard POVs and Valid Intersections.

Dashboard POVs and Valid Intersections

The POVs in dashboards honor the valid intersections by hiding invalid Page members. Just like in forms, the Page drop-down list is filtered for all selected members in the POV and Page dimensions. Because dashboards support both global and local POVs, the context for filtering the Page drop-down list depends on which POV the members are located. If the Page drop-down list is on a global POV, the filtering context is only the global POV dimensions. If the Page drop-down list is on the local POV, the filtering context is all the global dimensions plus the dimensions on a chart's local POV.

See also About Global and Local POVs in 1.0 Dashboards.

Creating and Managing Dashboards 2.0

Related Topics

- Concepts in Designing Dashboard 2.0 Dashboards
- Considerations for Dashboard 2.0
- Creating Dashboard 2.0 Dashboards
- Working with the Member Selector When Designing Dashboards 2.0
  When designing Dashboard 2.0 dashboards, you can easily search for and select new dimension members using the member selector.
- About Forms 2.0 Grids in Dashboard 2.0
- About the Geomap Chart Type
- About the Pyramid Chart Type
- About the Waterfall Chart Type
- Using Tables in Dashboards
- About the Gauge Chart Type for Dashboard 2.0
- About the Radar Chart Type for Dashboard 2.0
- About the Combination Chart Type for Dashboard 2.0

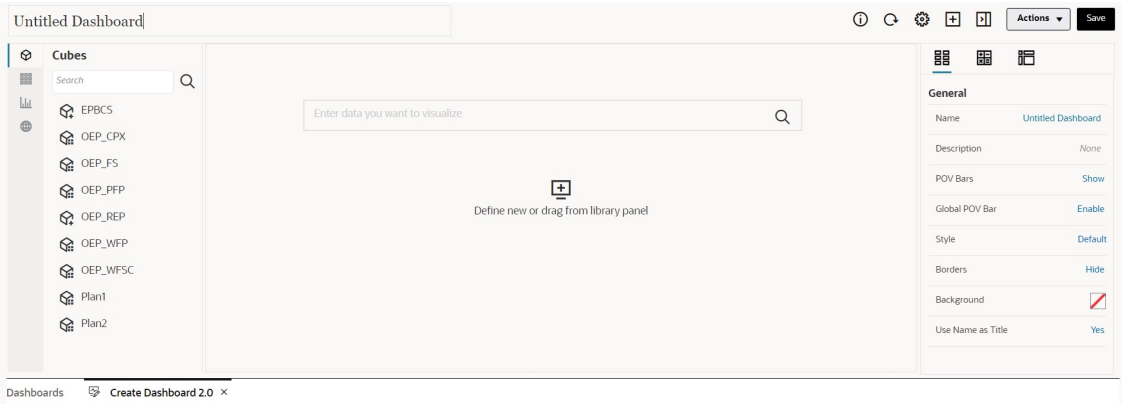
- About the Tile Chart Type for Dashboard 2.0
- About Global and Local POVs in Dashboard 2.0
- About Quick Analysis

Concepts in Designing Dashboard 2.0 Dashboards

This topic contains helpful information as you design Dashboard 2.0 dashboards:

- About the Dashboard 2.0 Designer
- Opening the Dashboard 2.0 Designer
- Data Sources for Charts
- More Helpful Information About Dashboard 2.0
- Videos

About the Dashboard 2.0 Designer



The Dashboard Designer consists of the following dashboard components:

- Object Palette
- Dashboard Toolbar
- Dashboard Workspace
- Properties Panel
- Dashboard Components

Opening the Dashboard 2.0 Designer

As a reminder, Dashboard 2.0 dashboards on the **Dashboards** page are identified by this icon:

Using Dashboard 2.0 is only supported if Redwood Experience is enabled.

To open the Dashboard 2.0 Designer:

1. From the Home page, click **Dashboards**.
2. Select one of the following options:
   - To create a dashboard, click **Create**, and then select **Dashboard 2.0**.

- To edit a dashboard, in the **Actions** column to the right of the Dashboard 2.0 dashboard you'd like to edit, click •••, and then click **Edit**.

  If you don't need to work with data while editing a Dashboard 2.0 dashboard; for example, you'd like to realign the dashboard components and tiles, click •••, and then click **Edit without Data**.

3. If the Dashboard 2.0 dashboard is open in the runtime environment, you can switch from the runtime environment to the Dashboard Designer environment.

   Click **Actions**, and then select **Edit** or **Edit without Data**.

Data Sources for Charts

Dashboard 2.0 supports ad hoc grids, forms, and cubes for choosing the data. You pick the data sources from the object palette.

> **Note:**
>
> You can include a maximum of 12 components in a Dashboard 2.0 dashboard.

- The properties panel provides full data source information for a dashboard component on the **Chart** tab  . Select the dashboard component, then click  . The data source information is displayed under the **Visualization** heading.
- When using a cube as the data source for quick analysis, drag and drop a cube from the object palette to the dashboard workspace, then click in the Search box and either enter the member name or launch the member selector to choose members. The **Layout** tab will then display in the properties panel where you can pivot the dimensions and choose which dimensions appear on the POV, row, or column axis. See About Quick Analysis.

  > **Note:**
  >
  > There is an upper limit of 50 members across all dimensions that can be selected to create a chart using quick analysis. This limit together with the number of dimensions in the cube will determine the maximum size of the grid that is supported for a chart based on quick analysis.

- When the data source for a dashboard component is a form or an ad hoc grid, the data is fetched using the form definition.
  - You can change the source form in the properties panel by clicking the **Form** property in the  tab of the properties panel.
  - The total amount of form data that's allowed to be visualized in a chart is 10,000 cells. By default, this is 100 rows and 100 columns. However, you can change the number of rows and columns to any combination that totals 10,000 cells or less; for example, if you reduce the number of rows to 5, then you can increase the number of columns to 2,000 (5 x 2000=10,000).
  - If a form's precision is set, displayed values after conversion to Dashboard 2.0 are based on the form's precision setting. If the precision value on a form is not set, then the values displayed after conversion to Dashboard 2.0 might change.

> **Note:**
>
> When the data source for a dashboard component is a form, and the form has rows that contain blank header cells or empty formula rows that are not configured, then the corresponding chart legends will not display.

- For Doughnut, Pie, and Gauge chart types, you can add multiple charts to one component. To add multiple charts, click **Single** or **Multiple** in the chart's **Display** property. If **Single** is selected, the chart displays only the first column of the underlying form's data. If **Multiple** is selected, a chart is displayed for each column of the underlying form's data.

> **Note:**
>
> For dashboards with Doughnut, Pie, and Gauge chart types that are converted from Dashboard 1.0 to Dashboard 2.0, the default option for the **Display** property is **Multiple**. For newly created Dashboard 2.0 dashboards, the default option for **Display** is **Single**.

More Helpful Information About Dashboard 2.0

- As you create a dashboard, click **Actions** and then select **Save and Run** to immediately see how the dashboard looks and works to a dashboard user. To return to designer mode to continue designing the dashboard, click **Actions** and then select **Edit** or **Edit without Data**.
- By default, missing or suppressed data is plotted as zeros. For selected chart types (Area, Bubble, Combination, Line, Radar, and Scatter), you can disable this setting by selecting the **Plot Missing Values as Zero** option in the chart's settings. When this option is disabled, the missing or suppressed data is ignored and is no longer plotted as zeros for those chart types.
- To review Dashboard 2.0 considerations, see Considerations for Dashboard 2.0.

Videos

Your Goal	Watch This Video
Learn how to create a dashboard with multiple components in the Dashboard Designer for Dashboard 2.0.	 Creating Dashboards in Dashboards 2.0

Object Palette

On the left side of the Dashboard Designer is the object palette where you select the content for dashboard components. Simply drag and drop objects from the object palette to the workspace or search for content using the Search box.

The object palette can be resized. To resize the object palette, hover over the right side of the palette and drag it. To hide the palette, hover over the palette until you see the handle icon , then click it to hide the palette. Click the handle icon again to reopen the palette.

Content in the object palette is grouped into these categories:

Table 10-3 Object Palette Content Categories

Icon	Description
	Cubes: Perform quick analysis by adding components of a cube to a dashboard without having a prebuilt form. See About Quick Analysis.
	Library: Select ad hoc and standard forms to include in the dashboard by navigating to the forms folders or by searching for them by name. • Click **List** to view forms in a **List View** or **Tree View**. • Click to type search keywords. The access permissions set for forms are honored in dashboards.
	Visualizations: Select a chart to include in the dashboard. To view data in a chart, you must select components of a cube or associate the chart with a form as its data source. When you link a chart to a form, users can immediately see the impact of changing data in the form on the associated charts.
	Other: • **Commentary**: Adds the comment you enter below the graph or chart and to the bottom of the **Chart** tab in the properties panel. **Commentary** includes a WYSIWYG text editor. • **URL**: Adds the URL you enter below the graph or chart and to the bottom of the **Chart** tab in the properties panel. **URL** includes a Reference URL link and the URL must be IFrame supported. Insert only external site URLs starting with the `https://` security protocol. Don't use internal or relative URLs or URLs for unconsenting third party sites such as google.com. Don't imbed into a dashboard direct URLs to other EPM artifacts such as forms and dashboards on either the same Oracle Enterprise Performance Management Cloud instance or to a different EPM Cloud instance. **Note:** If a URL added to a dashboard doesn't work in the runtime environment, right-click the URL and open the link in a new tab.

Dashboard Toolbar

In the upper right corner of the Dashboard Designer is the dashboard toolbar where you can perform these dashboard tasks:

Table 10-4 Dashboard Toolbar

Icon	Description
	The **Information** icon displays the folder in which the dashboard is located.
	The **Refresh Data** icon refreshes all data associated with the entire dashboard.
	The **POV Settings** icon performs these POV actions: • **POV : Hide Dimension Labels**: If your dashboard contains a global POV bar, select to hide the dimension names on the global POV bar. Clearing this option once again displays dimension names on the global POV bar. • **POV : Auto-Apply**: If your dashboard contains a global or local POV, enable this action to automatically apply changes to the POV. If you don't want to automatically apply changes, clear this option. • **POV : Clear All Page Selections**:
	The **Add** icon adds a new empty dashboard component to the design canvas (this is an alternative to drag and drop).
	The **Properties** icon hides and unhides the Properties Panel.
Actions menu	The **Actions** menu enables you to perform these actions: • **Revert to Last Saved**: Discards edits made since the last time the dashboard was saved • **Save and Run**: Saves any edits made and opens the dashboard in the runtime environment
Save	Clicking **Save** saves the dashboard definition.

Dashboard Workspace

The large area in the middle of the page between the object palette and Properties panel is the dashboard workspace. In addition to drag and drop, you can use the Search box to specify or search for the data you want to display for each component (a *component* is an empty object in the dashboard workspace into which you can add a dashboard object, such as a chart or a table). You can add up to 12 components to the dashboard workspace.

If Redwood Experience is enabled and the forms version is Forms 2.0, any forms dropped into the workspace are rendered initially as editable grids. Ad hoc grids dropped into the workspace are rendered initially as tables.

- The first object that you drag and drop occupies the entire workspace, then drag additional objects from the object palette to the left, the right, the top, or the bottom of the existing component.
- To move a component to another part of the dashboard, click the component, and then hover your cursor over the upper edge of the component until you see the drag icon. Drag and drop the component to the left, right, top, or bottom of another component and the other components on the dashboard will move and realign.
- To maximize a component so it fills the workspace, click , and then click **Maximize**. The component will fill the workspace, and the other components on the workspace will be hidden. Clicking and then **Restore** resizes the component to its original size and makes the other components in the workspace visible again.

- To change the height and the width of components in the dashboard, perform one of the following actions:
  - Drag the borders of the components within the dashboard workspace
  - Select a dashboard component, then in the Properties panel on the right, click  to view and change the component's height and width.
- To maximize the overall size of the dashboard workspace, you can resize or hide the object palette and properties panel on either side of the workspace.
- For a dashboard with a URL component, clicking on the body of the URL component does not switch focus to the URL component. To switch focus to the URL component, either click Alt + W or hover over the URL title bar until you see the cursor change to the move icon  , then click to change the focus.

> **Note:**
>
> In the runtime environment, if a user doesn't have access to an object like a form, or if the form is missing, then an adjacent component will take its space. In the designer environment, all empty components are displayed so that the designer can choose to remove them.

Properties Panel

On the right side of the Dashboard Designer is the Properties panel where you can set general properties for the entire dashboard as well as component-specific properties.

The Properties panel can be resized. To resize the panel, hover over the left side of the panel and drag it. To hide the panel, click  at the top of the page. Click  again to reopen the panel.

Table 10-5 Properties Panel

Icon	Description
	The **General** tab includes the following properties for the entire dashboard: • **Name**: Edit the title of the entire dashboard. • **Description** (optional): Describe the dashboard. • **POV Bars**: Show or hide POV bars. • **Global POV Bar**: Enable or disable the global POV bar. • **Style**: Select the dashboard color scheme. Options are **Default**, **Light**, and **Dark**. **Note:** If you select **Dark**, dashboard error messages and legend text will be more difficult to view. • **Borders**: Show or hide component borders. • **Background**: Select a new dashboard background from a menu of colors. **Note:** If your dashboard contains a grid, only the background color will change. The grid color will remain the same. • **Use Name as Title**: Selecting **No** lets you customize the dashboard's title. You can further customize the title by changing the font family, font size, bolding, italicizing, underlining, font color, and font background color. For information about POVs, see About Global and Local POVs in Dashboard 2.0.

Table 10-5 (Cont.) Properties Panel

Icon	Description
	The **Associations** tab lists the rules associated with the dashboard and the underlying form data. **Note:** You can associate rules which are required to be run during a global **Save**. These rules can be associated or removed at any time. You can add rules and edit them: • If no rules are associated, click  to view a list of rules from which to choose. Rules already associated with forms in the dashboard are listed first and are named as follows: Business Rules for <name of form> • When you add rules, you'll need to specify when the rule should run: **Before Load**, **After Load**, **Before Save**, **After Save**, **Use Members on Form**, **Hide Prompt**. • Under **Actions**, you can delete rules or change the order in which they will run: **Move to Top**, **Move Up**, **Move Down**, **Move to Bottom**, and **Delete**. • If the rules have runtime prompts, they will be displayed when the rule is run. See Setting Business Rule Properties.

Table 10-5 (Cont.) Properties Panel

Icon	Description
	The **Chart** tab includes properties for the selected components in the dashboard such as chart title, component height, width, and background color. You can also change the chart type, the data source type, and which underlying form is used for each component. Using different background colors for each component enables you to uniquely differentiate each dashboard component. Removing the chart title frees space that is then adjusted for the local POV bar and for the chart itself. For rows and columns, the total amount of form data that's allowed to be visualized in a chart is 10,000 cells. By default, this is 100 rows and 100 columns. However, you can change the number of rows and columns to any combination where the multiplication totals 10,000 cells or less; for example, if you reduce the number of rows to 5, then you can increase the number of columns to 2,000 (5 x 2000=10,000). **Note:** The 10,000 cell visualization limit does not apply to grids. If the form associated with a chart exceeds the 10,000 cell visualization limit, it might result in discrepancies between the data you see in the form and the way the data is visualized in the dashboard chart. It also might result in some other runtime issues such as drilling through on charts and legends. If you've added **Commentary** or **URL** information types to this component, you can edit the content of those information types at the bottom of this tab.

Table 10-5 (Cont.) Properties Panel

Icon	Description
(this icon will change depending on the component that's selected in the dashboard)	This tab lets you customize visualizations for the selected chart or table. Set display properties for the selected components such as how to scale values, show or hide row dividers or grid lines, choose different font sizes and colors for headers and data, set orientation vertically or horizontally, define legend positions, add spark charts to tables, add a secondary Y-axis, and so on. See the chart type topics for detailed property descriptions. **Note:** For data visualizations in Dashboard 2.0, the custom color palette now supports only 12 colors instead of 15. If you created a dashboard in an earlier update that used 15 colors, you will continue to see 15 colors at runtime. However when you try to change the custom colors, the dashboard designer will only display 12 colors from which you can select and save.
	The **Layout** tab includes the **POV**, **Row**, and **Column** properties for the selected cube for quick analysis. See About Quick Analysis.

Dashboard Components

You can add up to 12 components to the dashboard workspace.

When you hover over a component in the workspace, a toolbar for that component displays in the upper right corner:

Table 10-6 Dashboard Components Toolbar

Icon	Description
	Information: Displays the underlying form details, including the form name, the folder path of the form, and the cube
	Save: Saves any changed data and runs any rules associated with the component
	Refresh: Refreshes the data associated with the dashboard component
	Chart Type: Changes the chart type associated with the component

Table 10-6 (Cont.) Dashboard Components Toolbar

Icon	Description
•••	**Actions** menu: • **Maximize**: Enlarges the dashboard component. Once maximized, click **Restore** to minimize the dashboard component to its original size. • **Open Form**: Opens the runtime form associated with the component in a dynamic tab next to the current tab. • **Edit Form**: If you have the correct permissions, opens the form associated with the component in the forms editor. The forms editor opens in a dynamic tab next to the current tab. • **Delete**: Deletes a form.

Considerations for Dashboard 2.0

Note the following considerations when using Dashboard 2.0:

- 3D visualization is not supported in Dashboard 2.0. This affects the following Dashboard 2.0 chart types:
  - Area
  - Bar
  - Bubble
  - Column
  - Combination
  - Doughnut
  - Gauge
  - Grid
  - Line
  - Radar
  - Scatter

  > **Note:**
  >
  > Dashboard 1.0 still supports 3D visualizations for charts, as applicable.

- In a future update, Dashboard 2.0 will support Smart Push.
- Funnel charts are useful for viewing data for stages of a process, such as the stages of a sales process. The area of a funnel slice is proportional to its value for the corresponding stage. Dashboard 1.0 renders the funnel chart very differently than the way in which Dashboard 2.0 renders the funnel chart. In Dashboard 1.0, the funnel chart supports multiple columns, and the first two rows of the form are used to compare Actual with Target. The chart then plots the differences between the two in the funnel corresponding to the respective column. In Dashboard 2.0 the funnel chart supports only the first column and the area of the funnel corresponding to each cell on the row is colored and is sized proportionally to its numeric value.

- All future dashboard enhancements will be in Dashboard 2.0.

Forms Versions and Dashboard 2.0

Runtime forms are also available in two versions:

- **Forms 1.0**: Uses Slick Grid with Oracle ADF Fuse components
- **Forms 2.0**: Uses Slick Grid with Oracle JET components

Just like in Dashboard 2.0, Forms 2.0 is available only if **Redwood Experience** is enabled. Certain features in Dashboard 2.0 only work with Forms 2.0; for example, Dashboard 2.0 only lets you work directly with data in form grids if the underlying form is a Forms 2.0 form. The **Forms Version** can be updated in your application settings.

You'll need to be aware of some differences in the way the dashboard versions interact with the forms versions.

When the **Forms Version** application setting is set to **Forms 1.0**:

- You can continue to use Dashboard 1.0 with Forms 1.0.
- You cannot use Dashboard 2.0 with Forms 1.0.

  Forms 2.0 is supported with Dashboard 2.0 only. Use Forms 2.0 with Dashboard 2.0 for all interactions.

  If you've selected a Dashboard 2.0 dashboard and you see an `Error during Dashboard 2.0 component initialization` message, the dashboard has Forms 2.0 forms associated with it and you'll need to ensure that Redwood Experience and Forms 2.0 are enabled before you can view the dashboard.
- When Dashboard 2.0 artifacts are created, the **Grid** visualization option will not be available.
- To use forms as grids in Dashboard 2.0, update the **Forms Version** application setting to Forms 2.0.
- Existing Dashboard 2.0 dashboards (Dashboard 2.0 dashboards created before November 2023) will continue to display forms as tables and charts, not grids.

When the **Forms Version** application setting is set to **Forms 2.0**:

- You can continue to use Dashboard 1.0 with Forms 2.0.
- Dashboard 2.0 will display the **Grid** visualization option, which uses Forms 2.0 with read, write, and calculate abilities.
- Forms 2.0 within Dashboard 1.0 will appear as Forms 1.0.

> **Note:**
>
> Saved forms definitions can be used as a data source for dashboard charts regardless of the **Forms Version** application setting: **Forms 1.0** or **Forms 2.0**.

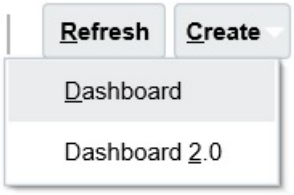
Creating Dashboard 2.0 Dashboards

Need help deciding which dashboard version to choose? See About Dashboard Versions.

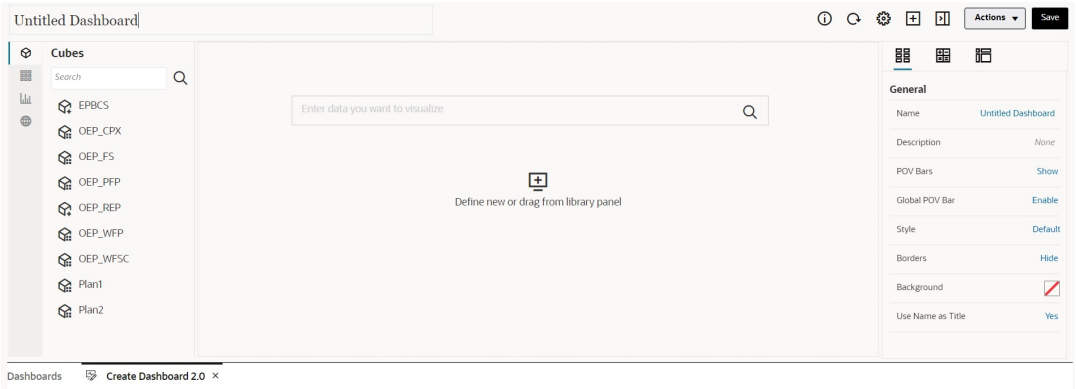
> **Note:**
>
> To create a Dashboard 1.0 dashboard, see Creating Dashboard 1.0 Dashboards.

To create a Dashboard 2.0 dashboard:

1. From the Home page, click **Dashboards**, and then click **Create**.



2. From the options provided, select **Dashboard 2.0**.



3. Enter a name for the dashboard by clicking the default dashboard name and entering a new name in the input box.
4. Drag and drop content from the object palette on the left onto the dashboard workspace. Customize the dashboard using the dashboard settings and properties.

   For a detailed description of the Dashboard Designer, see Concepts in Designing Dashboard 2.0 Dashboards.

   While working on the dashboard, you can save it as a new dashboard, including any unsaved changes. To do so, click **Actions**, then **Save As,** and then save the dashboard with the new name.
5. Click **Save**.

You can easily duplicate and modify a dashboard using **Copy As** on the **Dashboards** listing page. Select the **Actions** icon • • • next to the dashboard you want to copy.

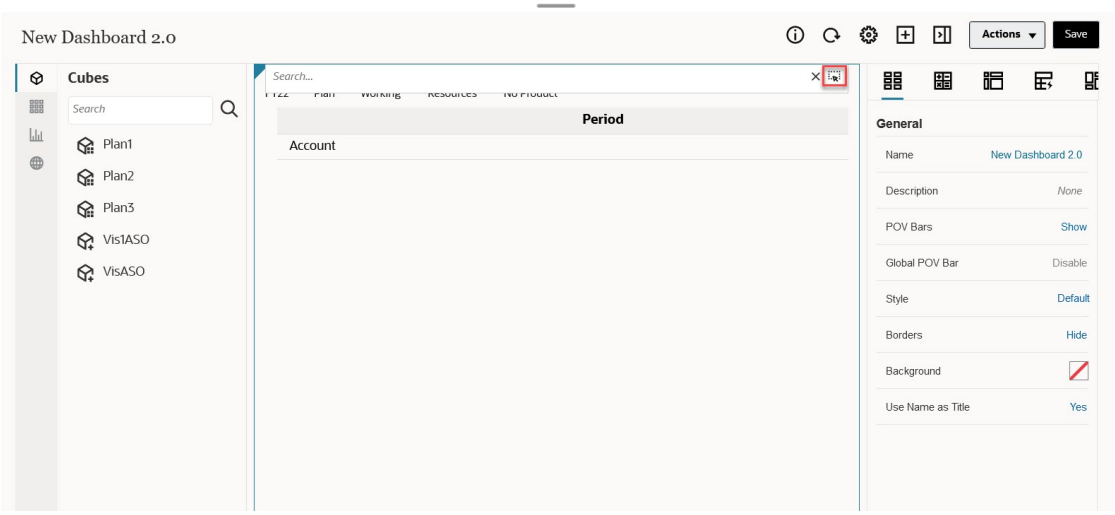
Working with the Member Selector When Designing Dashboards 2.0

When designing Dashboard 2.0 dashboards, you can easily search for and select new dimension members using the member selector.

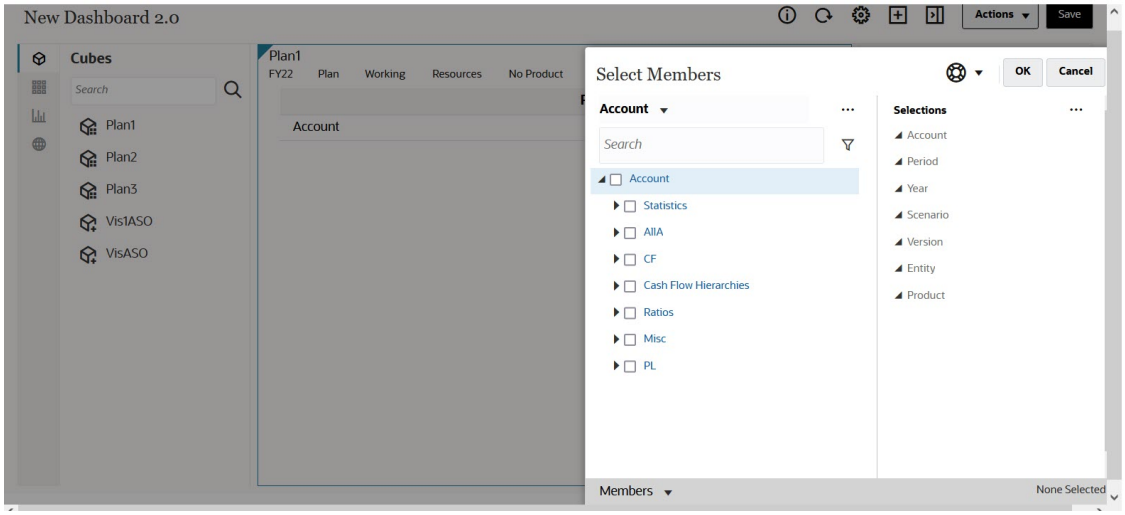
When you're creating a Quick Analysis in your dashboard, after you add a cube, you can select members for each dimension using the member selector.

To open the member selector for Quick Analysis:

Click the member selector icon in the Quick Search bar:



Select members for each dimension using the member selector.

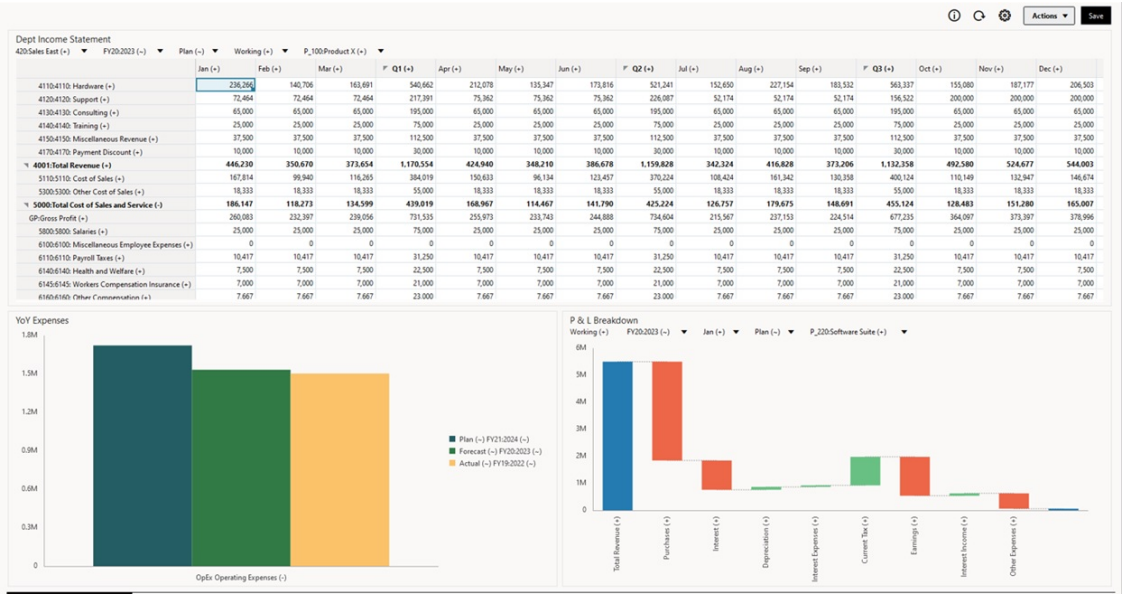


For more information about working with the member selector, see Using the Member Selector.

About Forms 2.0 Grids in Dashboard 2.0

Newly added components in Dashboard 2.0 default to displaying the form grid, and the grids are writable if the associated form is a Forms 2.0 form. Grids in dashboards enable you to work directly with data in forms if the underlying form is a Forms 2.0 form. You can write-back data updates and invoke rules in dashboards with grid components. You can also right-click inside a grid to view a context menu where you can perform additional grid actions such as grid spread, and adding comments and attachments.

Example Dashboard with a Grid and Charts

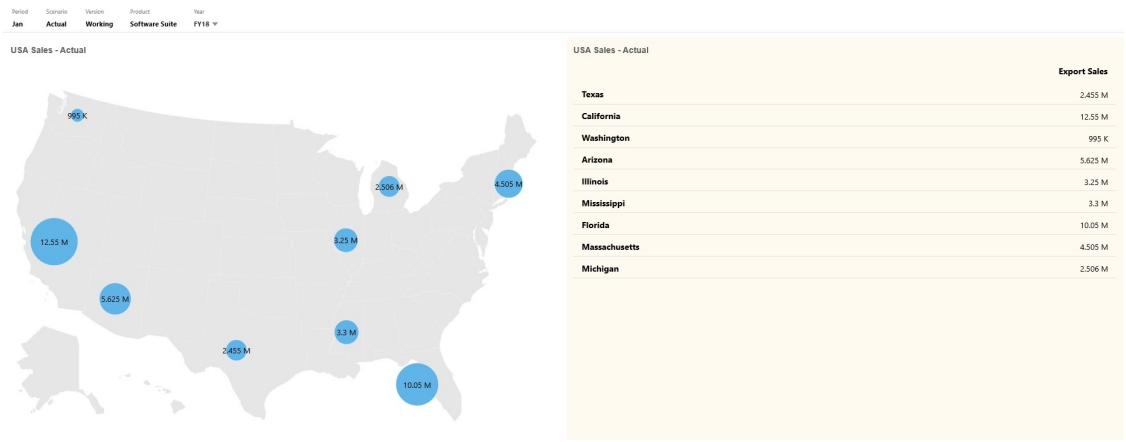


About the Geomap Chart Type

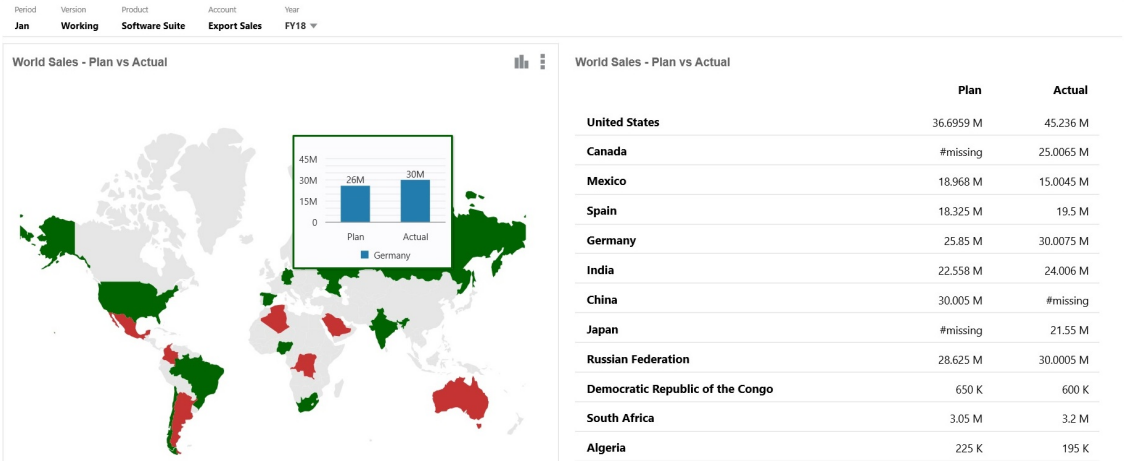
Geomaps are geographical maps with overlays and charts that display some measure about each of the geographical areas that are displayed. Hovering over the bubbles or the color highlight in the Geomap displays a data label.

> **Note:**
>
> Color highlight geomaps are designed to show any variances (red or green, depending upon your **Desired Variance Trend** setting) based on the values in the underlying data. If a geomap region has missing values in the 1st or 2nd column or if the values in those columns are equal and there is no variance, the region will be highlighted blue. Hovering over a highlighted region displays a bar chart with all the values for the row except for any missing values.

Example Geomap - USA Sales Using Data Bubbles



Example Geomap - World Sales Using Color Highlight



Dashboard designers can set these geomap chart properties:

Table 10-7 Geomap Chart Properties

Setting	Description
Map Layer	Select **World** or narrow the scope of the map to a continent such as **Asia**, **North America**, or **Africa**.
Type	The **Data Bubbles** geomap displays data as sized bubbles over the map layer. The **Color Highlight** geomap compares two data values, displaying geographical areas that have an increase in green, and those with a decrease in red. Select **Data Bubbles** or **Color Highlight** and then specify the following additional options: • For **Data Bubbles**, select the desired settings for **Label** (**Show** or **Hide**), **Label Color**, and **Bubble Color**. • For **Color Highlight**, specify the **Desired Variance Trend** (**Increase** or **Decrease**).

Table 10-7 (Cont.) Geomap Chart Properties

Setting	Description
Geography Names	We've provided you with a metadata load file that contains the geographical elements the geomap uses like countries, states, and cities. Click to download a CSV file which you can then upload into a dimension as new members or (with some manual changes) as aliases to existing members in your application. Note that geomaps only work if the geography names in the forms match exactly the geography names that Dashboard 2.0 uses for its map layers. So if your entity/geography dimension says, for example, USA, that won't be recognized as a valid name in the map layer, which uses United States.

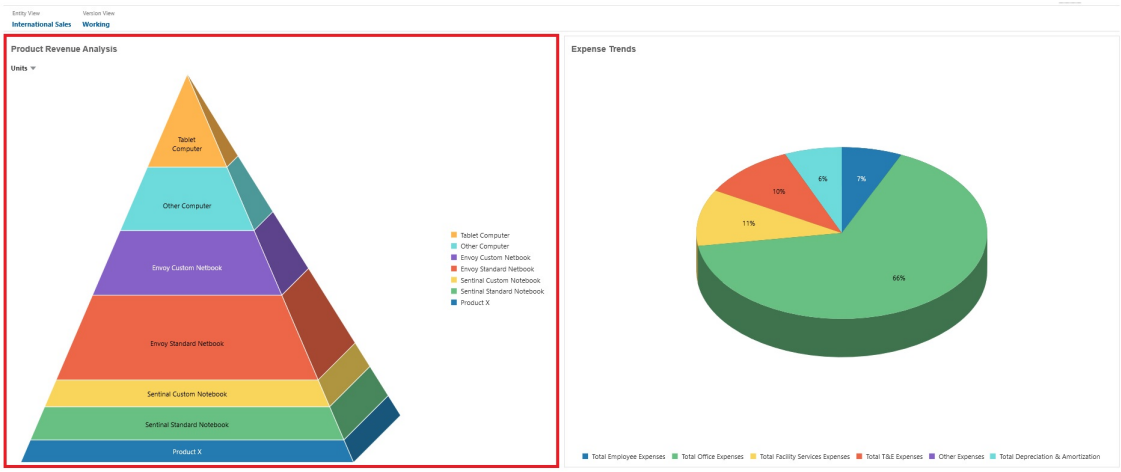
Videos

Your Goal	Watch This Video
Learn how to download geography members for geomap charts, then import them into your dimension. You learn how to set up both data bubble and color highlight geomap charts.	Creating Geomaps in Dashboards 2.0

About the Pyramid Chart Type

Pyramid chart types use a triangle diagram to represent hierarchies visually. The triangle is divided into sections, illustrating vertically top-down and broad-narrow structures and processes.

Example 3D Pyramid Chart



Dashboard designers can set these pyramid chart properties:

Table 10-8 Pyramid Chart Properties

Setting	Description
Legend Position	Choose **Left**, **Right**, **Top**, **Bottom**, or **None**.
Value Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the chart displays the value as 1,689K. Your scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
3D Graph	Choose **Yes** to visualize the data as a 3D chart.
Color	Select **Default** or click a section of the pyramid chart and choose a custom color for that section.

About the Waterfall Chart Type

Waterfall charts display a running total of positive and negative values, which is helpful in showing how you arrived at a net value from an initial value.

Waterfall charts are used to portray how an initial value is affected by a series of intermediate positive or negative values. Usually the initial and the final values (end points) are represented by whole columns, while the intermediate values are shown as floating columns that begin based on the value of the previous column. The columns can be color-coded to distinguish between positive and negative values.

Example Waterfall Chart - Inventory Analysis

This example visualizes changes in a single account over time.

Example Waterfall Chart - P & L Statement

This example has expense accounts that are shown as a negative change to Total Revenue, and income accounts that are shown as a positive change, for a final Net Income value.

P & L Statement

	Total Revenue	Purchases	Interest	Depreciation	Interest Expenses	Current Tax	Earnings	Interest Income	Other Expenses
International Sales	5.5 M	1.845 M	760 K	865 K	925 K	1.98 M	540 K	630 K	60 K

Dashboard designers can set these waterfall chart properties:

Table 10-9 Waterfall Chart Properties

Setting	Description
Value Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the chart displays the value as 1,689K. Your scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
Grid Lines	Select **Hide** or **Show** to display the x-axis lines of the grid.
Connecting Lines	Select **Hide** or **Show** to display the lines connecting the end of each column to the beginning of the next column. This helps visualize the flow of the data in the chart.

Table 10-9 (Cont.) Waterfall Chart Properties

Setting	Description
Plot Deltas	Select **Yes** to automatically calculate and plot the common start and end points from one bar to the next based on the values provided. This is useful for showing the trend for balance type of accounts. Otherwise, select **No** to manually set increase and decrease points. When **No** is selected, you can manually set these increase/decrease options on a bar-by-bar basis: • Bar Direction: Select a bar to manually set the bar direction to increase/decrease. • All bars: Display a list of all bars in the chart to manually set bar directions to increase/decrease. The manual options you can set for each bar are **Auto**, **Reverse**, and **Total**. **Auto** will consider the positive/negative values from the data source and plot accordingly. **Reverse** will change the sign, and **Total** will make the chart restart at the corresponding bar.
Increase **Decrease** **Total**	Select a color for each property.
End Bar	Select whether to **Hide** or **Show** the end bar on the chart. In the preceding example, the end bar is shown. If the end bar is shown, you can specify a label.
Axes	Select a **Custom Range** for the **X-Axis** and the **Y-Axis**, or select **Off**.

Videos

Your Goal	Watch This Video
Learn how to create waterfall charts to track the changes in a single account over time and to track how multiple accounts interact to produce a final total.	 Creating Waterfall Charts in Dashboards 2.0

Using Tables in Dashboards

Tables in dashboards provide a list view of the data showing multiple values from a form or grid. You can embed mini-charts into each row of the table, called Spark Charts, which plot column data. Tables are also searchable.

Example Dashboard Showing Plain Tables

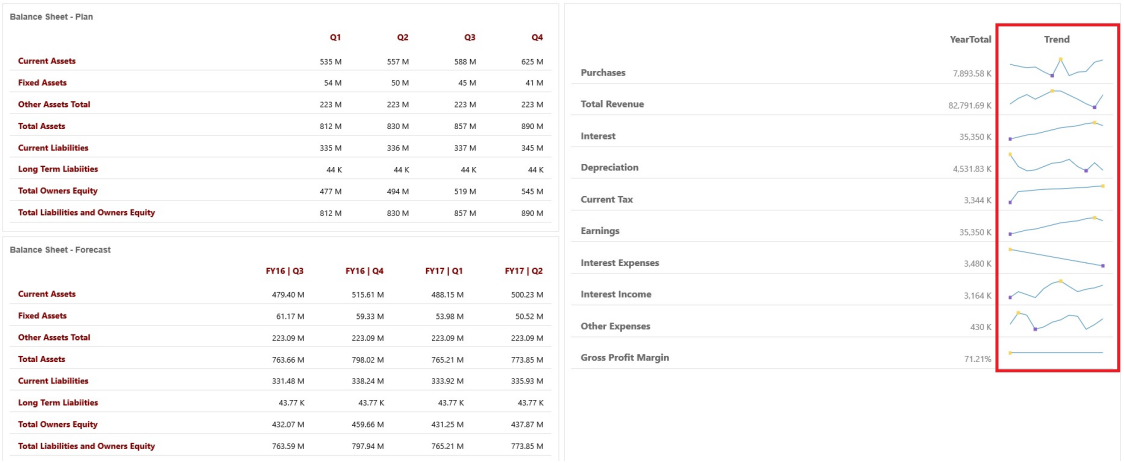
Balance Sheet - Plan

	Q1	Q2	Q3	Q4
Current Assets	535 M	557 M	588 M	625 M
Fixed Assets	54 M	50 M	45 M	41 M
Other Assets Total	223 M	223 M	223 M	223 M
Total Assets	812 M	830 M	857 M	890 M
Current Liabilities	335 M	336 M	337 M	345 M
Long Term Liabilities	44 K	44 K	44 K	44 K
Total Owners Equity	477 M	494 M	519 M	545 M
Total Liabilities and Owners Equity	812 M	830 M	857 M	890 M

Balance Sheet - Forecast

	FY16 | Q3	FY16 | Q4	FY17 | Q1	FY17 | Q2
Current Assets	479.40 M	515.61 M	488.15 M	500.23 M
Fixed Assets	61.17 M	59.33 M	53.98 M	50.52 M
Other Assets Total	223.09 M	223.09 M	223.09 M	223.09 M
Total Assets	763.66 M	798.02 M	765.21 M	773.85 M
Current Liabilities	331.48 M	338.24 M	333.92 M	335.93 M
Long Term Liabilities	43.77 K	43.77 K	43.77 K	43.77 K
Total Owners Equity	432.07 M	459.66 M	431.25 M	437.87 M
Total Liabilities and Owners Equity	763.59 M	797.94 M	765.21 M	773.85 M

Example Dashboard Showing a Table With a Spark Chart



Dashboard designers can set these table properties:

Table 10-10 Table Properties

Setting	Description
Value Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the table displays the value as 1,689K. Your scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
Row Divider	Choose **Show** or **Hide**.
Font	Specify size and color options for headers and data.
Spark Chart	• **Chart Type**: Select **Bar**, **Line**, **Area**, and **Line with Area** with associated options. Or you can select **None**. • **Header**: Column header for the Spark Chart column • **Starting Index**: Column index where to start charting data • **Ending Index**: Column index where to stop charting data

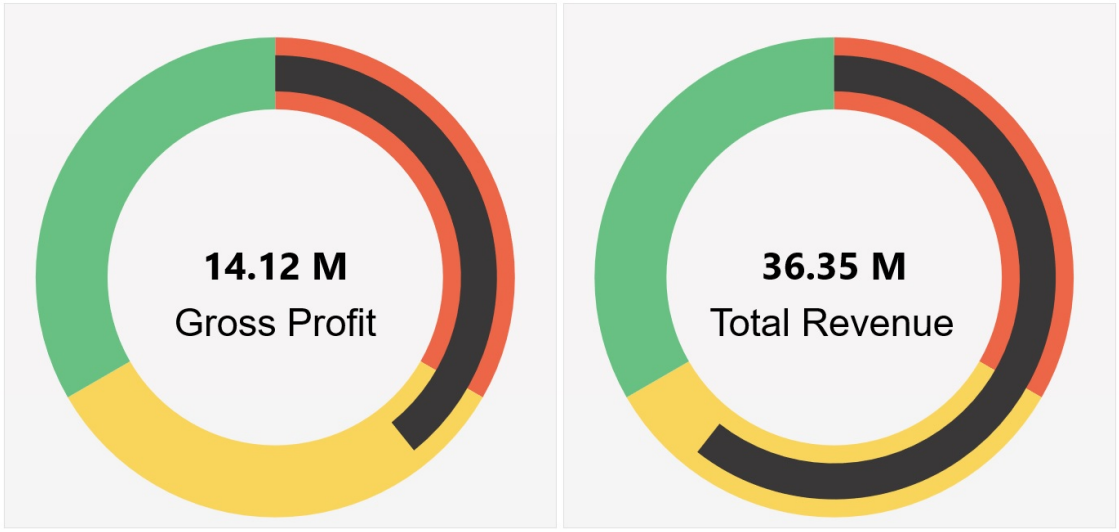
Videos

Your Goal	Watch This Video
See how to create a dashboard with multiple components in the Dashboard Designer for Dashboard 2.0. You create a dashboard with a table, a table with a spark chart, and a tile chart.	 Creating Table, Spark, and Tile Charts in Dashboards 2.0

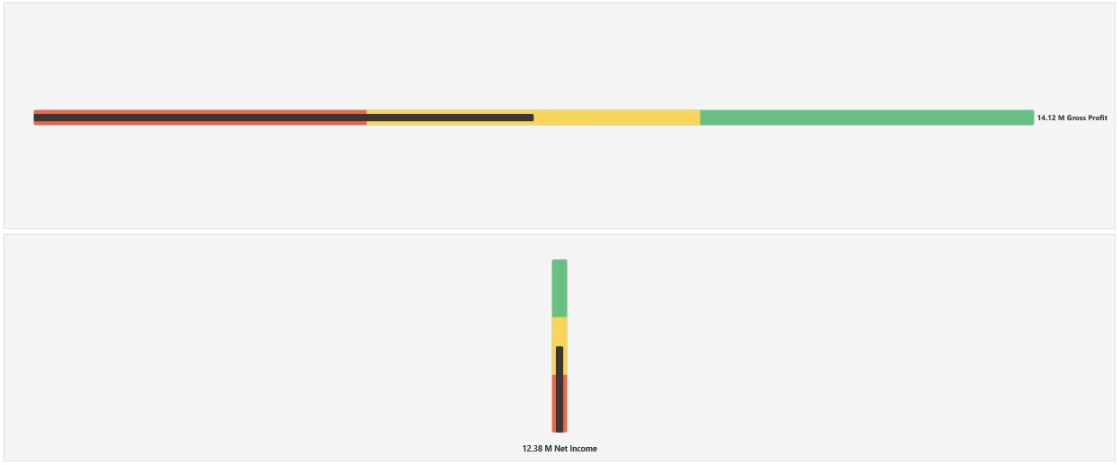
About the Gauge Chart Type for Dashboard 2.0

Gauge chart types show you whether data values fall within an acceptable range or not. You set the minimum and maximum values, the range maximums, and the gauge uses red, yellow, and green traffic lighting to help you quickly assess a current value and identify problems in important data points or measures. For Dashboard 2.0 dashboards, you can select circular, horizontal, and vertical gauge types.

Dashboard 2.0 Example - Circular Gauge Type

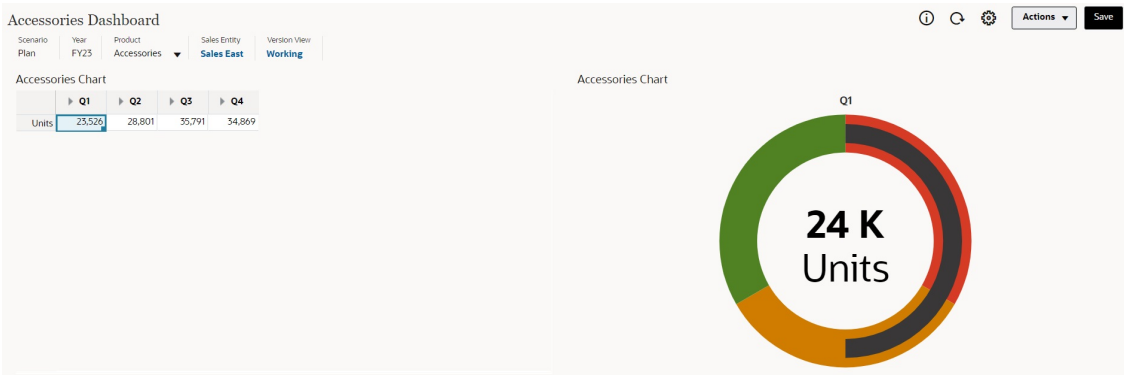


Dashboard 2.0 Example - Horizontal and Vertical Gauge Types



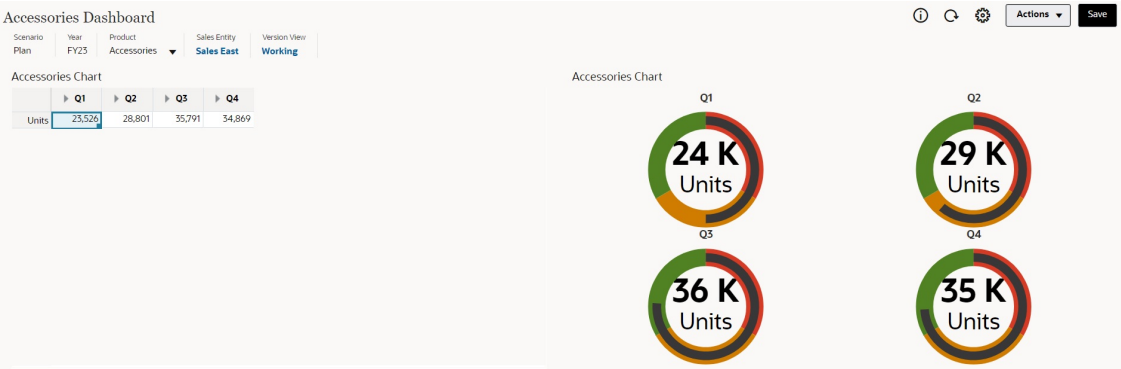
Dashboard 2.0 Example - Single Gauge Display

For **Display**, if **Single** is selected, then the gauge chart displays only the first column of the underlying form's data.



Dashboard 2.0 Example - Multiple Gauge Display

For **Display**, if **Multiple** is selected, then a gauge chart is displayed for each column of the underlying form's data.



Dashboard designers can set these gauge chart type properties:

Table 10-11 Gauge Chart Type Properties

Setting	Description
Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the chart displays the value as 1,689K. Your scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
Display	You can display **Single** or **Multiple** gauge charts in a component. If **Single** is selected, then the gauge chart displays only the first column of the underlying form's data. If **Multiple** is selected, then a gauge chart is displayed for each column of the underlying form's data.
Gauge Type	Select from the following options: • **Circular** • **Horizontal** • **Vertical**
Minimum Value **Maximum Value**	The lowest and highest values on the gauge. The dashboard designer sets the values as a default, and then users can temporarily change them at runtime. If the dashboard designer doesn't specify these values, the application automatically sets the minimum value as zero and the maximum value as greater than the value on the gauge.

Table 10-11 (Cont.) Gauge Chart Type Properties

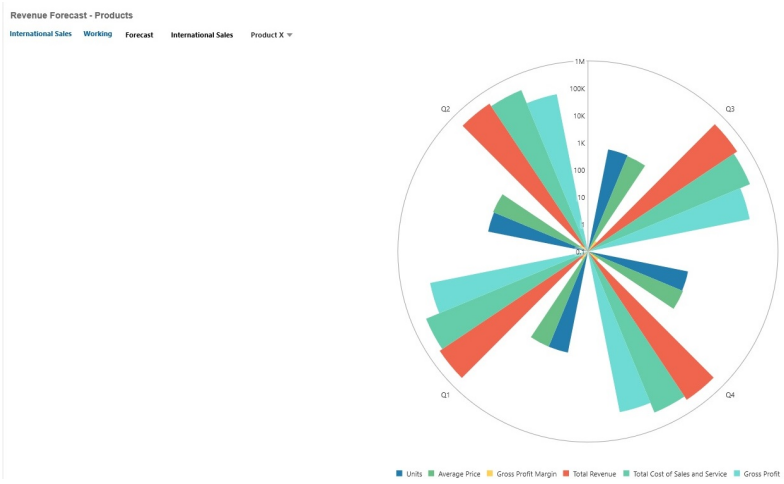
Setting	Description
Desired Goal	Specify whether **High** values or **Low** values are desirable.
Low Threshold **Medium Threshold** **High Threshold**	To visually indicate whether a measure lies in the acceptable range or not, these thresholds enable you to display the gauge in red, yellow and green based on the specified values. Specify a text **Label** and a number **Value** for each threshold.

About the Radar Chart Type for Dashboard 2.0

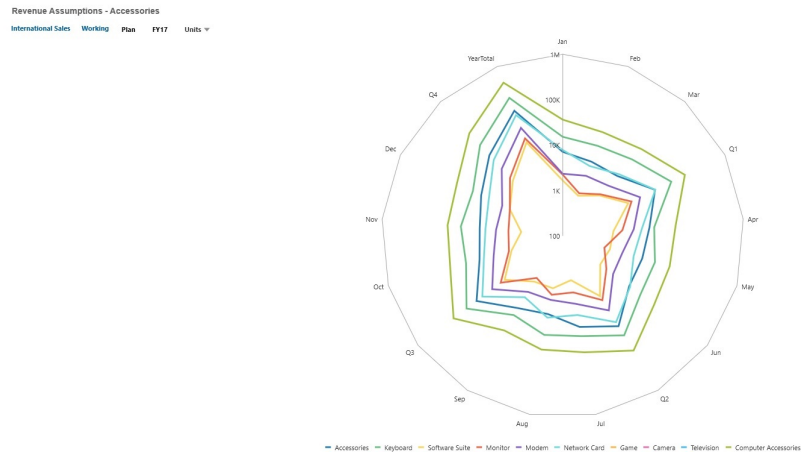
A radar chart is a two-dimensional chart type designed to plot one or more series of values over multiple quantitative variables. Radar charts are best for determining which variable in the data is doing better than the rest. Hence, they are mostly used for performance analysis.

For Dashboard 2.0, you can select bar, line, area, scatter, and bubble radar types.

Dashboard 2.0 Example - Radar Bar Chart Type



Dashboard 2.0 Example - Radar Line Chart Type



Dashboard designers can set these radar chart type properties:

Table 10-12 Radar Chart Type Properties

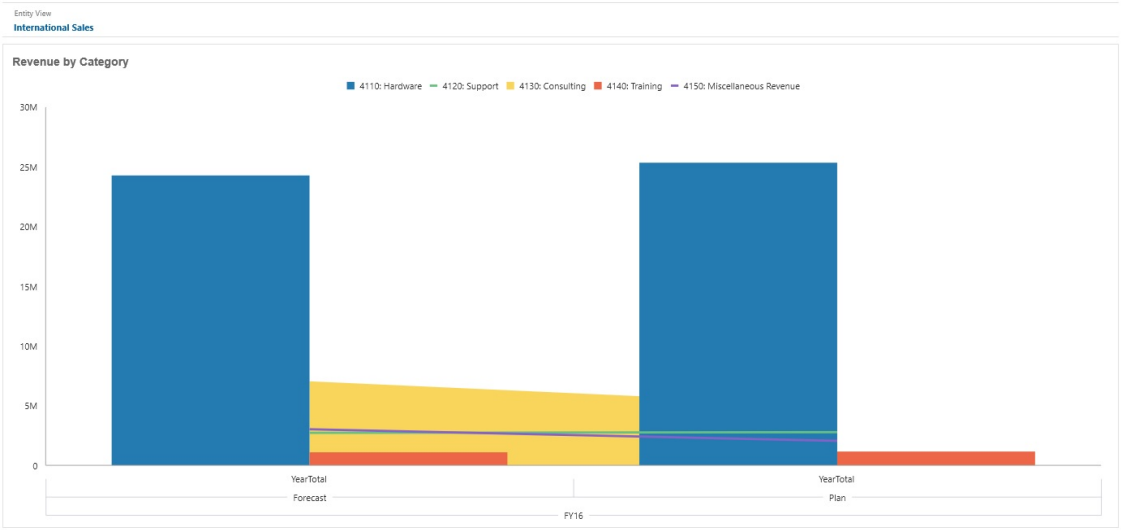
Setting	Description
Value Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the chart displays the value as 1,689K. Your scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
Type	Select from the following options: • **Bar** • **Line** • **Area** • **Scatter** • **Bubble**
Legend Position	Choose **Left**, **Right**, **Top**, **Bottom**, or **None**.
Grid Lines	Select **Hide** or **Show**.
Logarithmic Scale	Especially useful for displaying numerical data that covers a wide range of values (typically, the largest numbers in the data are hundreds or even thousands of times larger than the smallest numbers), a logarithmic scale displays these values in a way that is more compact and readable. Select whether to use a logarithmic scale in your radar chart.
Color	Select **Default** or click data in the chart and choose a custom color for that data.

About the Combination Chart Type for Dashboard 2.0

The combination chart is a visualization that lets you merge into one chart any combination of bar, line, and area charts.

You can also add a secondary Y-axis on the opposite side from the primary Y-axis. Adding a secondary Y-axis lets you display the scale for one measure that doesn't scale appropriately with the other measures on the chart; for example, showing a percentage measure and a currency measure on the same chart.

Dashboard 2.0 Example - Area Bar Line Combination Chart



Dashboard designers can set these combination chart type properties:

Table 10-13 Combination Chart Type Properties

Setting	Description
Value Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the chart displays the value as 1,689K. Your scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
Line Weight	Click the counter to set the line width. Note that the default width of a line is 5 pixels. You can select from 1 to 12 pixels.
Legend Position	Choose **Left**, **Right**, **Top**, **Bottom**, or **None**.
Label Position	Set the position of the data labels on the chart. Choose **Outside Bar Edge**, **Center**, **Inside Bar Edge**, or **None**.
Grid Lines	Select **Hide** or **Show**.
Logarithmic Scale	Especially useful for displaying numerical data that covers a wide range of values (typically, the largest numbers in the data are hundreds or even thousands of times larger than the smallest numbers), a logarithmic scale displays these values in a way that is more compact and readable. Select whether to use a logarithmic scale in your combination chart.

Table 10-13 (Cont.) Combination Chart Type Properties

Setting	Description
Secondary Y	Select which data to plot on a secondary Y-axis which will appear on the opposite side of the chart from the primary Y-axis. When the data values in a chart vary widely from data series to data series, or when you have mixed types of data (for example, currency and percentages), you can plot one or more data series on a secondary vertical (Y) axis. While the secondary Y-axis can be used with any line and bar chart type, it is more commonly used with the combination chart type, which helps to distinguish the data series that are plotted on the secondary Y-axis. For example, use a bar for the primary Y- axis, and line for the secondary Y-axis.
Series	Select the bar/line/area combination for the chart.
Color	Select **Default** or click data in the chart and choose a custom color for that data.
Axes	Select a **Custom Range** for the **X-Axis**, **Y-Axis**, and **Secondary Y-Axis**, or select **Off**.

About the Tile Chart Type for Dashboard 2.0

A tile is a chart type that lets you select specific values from the cube to display. In addition to using a form as a data source, you can directly enter a cell intersection that provides the value for a tile. By default, six tiles are initially displayed in a component, but you can display up to 10 tiles in a component. Until you associate a tile with data, it displays sample data.

With a form as the data source for a tile:

- You can have up to 10 tiles per component (six by default).
- The values in the first column of the form (up to the 10th row) are used to create the tiles.
- By default, the row title in the form is the tile's title, and it gets its values from the first column, then by row.
- You can set the tile's component title, the tile's height percentage, legend labels, and can select which axes from the form to include. For example, if you select a form with three rows, the tile displays three values.

> **Note:**
>
> Sometimes in a form, the first column may be collapsed (hidden) when viewed as a grid. But the collapsed column is still considered when the tile gets its values from the form. We recommend that you update the form definition and use a form as a data source that contains only the cells to be shown on the tile.

With a cell intersection as the data source for a tile, you can have only one tile per object.

The table below lists the options you can set for the tile chart type. Note that you can set whether the displayed value is horizontally aligned in the tile to the left, the center, or the right.

Table 10-14 Tile Chart Type Properties

Setting	Description
Value Scale	Especially useful for large numbers, you can scale how a value is displayed. For example, if the value is 1,689,000 and you select **K - Thousand** as the scaling option, the chart displays the value as 1,689K. Scaling options: • **None**: No scaling is applied. • **Auto**: The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T. • **K - Thousand**: The value is displayed as thousands units. For example, 1,689,000 displays as 1689K. • **M - Million**: The value is displayed as millions units. For example, 123,456,789 displays as 123M. • **B - Billion**: The value is displayed as billions units. For example, 12,345,678,912 displays as 12B. • **T - Trillion**: The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.
Align Text	Text alignment options: • **Left** • **Center** • **Right**
Count	The number of tiles displayed in the component (up to 10)
Display	Legend display options: • **Row and Column Dimensions** • **Row Dimensions** • **Column Dimensions** • **None**
Position	Legend positioning options: • **Left** • **Right** • **Top** • **Bottom**
Legend Font Size	Decrease or increase the legend font size.
Value Font Size	Decrease or increase the value font size.

Note:

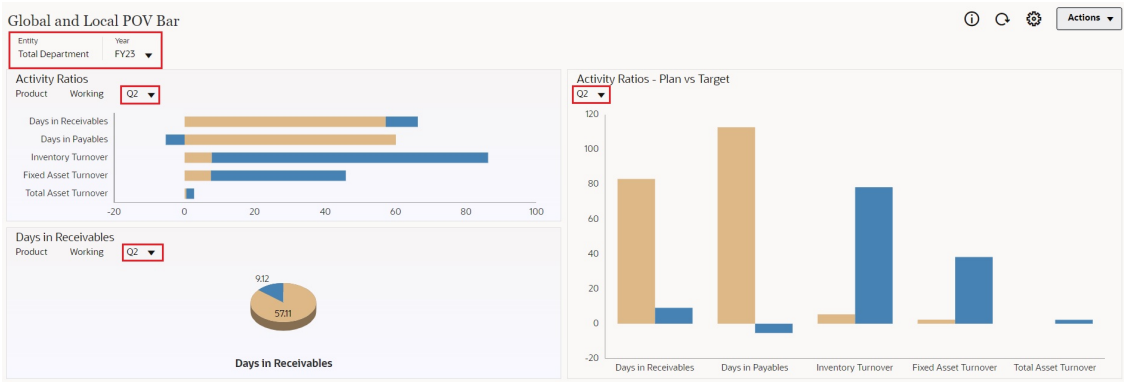
Dashboard 2.0 does not support cell details. Instead we recommend you use Quick Analysis as a data source, which is a more advanced version of cell details. If you convert a dashboard to Dashboard 2.0 that has a component using cell details, delete the component using cell details as a data source, and recreate it using Quick Analysis as a data source.

About Global and Local POVs in Dashboard 2.0

A local POV on a form reflects the dimension members the form designer selected for that form. Dashboards also support *global POV bars*, so that the local POVs that are common are combined in the global POV bar to avoid repeating them in each component.

Dashboard 2.0 POVs Example

This Dashboard 2.0 dashboard shows a global POV bar (showing Entity and Year) and local POVs (the drop-down lists for Q2):



With a global POV bar, if you change a page in the global POV bar and **POV Bar : Auto-Apply** is enabled, the page changes for all components in the dashboard that are based on forms. The global POV bar displays at the top of the dashboard above all the components, while the local POV bar displays within the component. User variables are supported in both global and local POVs in dashboards and you can change dynamic user variables directly from the POV bar.

In the dashboard toolbar  **POV Settings** menu (in both the runtime and edit modes), you can set whether to hide dimension labels in the global POV (**POV Bar : Hide Dimension Labels**) and whether to enable or disable automatically applying changes (**POV Bar : Auto-Apply**).

About global POV bars:

- The POV Bar is made up of local POV dimensions, Page dimensions, and user variables.
- Depending on the local POV and page of each form on the dashboard, the global POV bar is automatically calculated.
- Global POVs reflect the content of the components using forms in that dashboard. That is, they apply to forms in a dashboard, to charts that are linked to forms, and to tiles that use forms as a data source. So if the dashboard doesn't include a form as a data source, then neither the local nor global POV bar is available.

Here's an example of how the global POV bar is calculated, based on the local POV dimensions for two forms:

The global POV bar is disabled:

- Form A local POV: Year, Entity, Product
- Form B local POV: Year, Entity, Project

The global POV bar is enabled:

- Global POV bar: Year, Entity
- Form A local POV: Product
- Form B local POV: Project

Because not all dimensions and page selections may be common to all forms on a dashboard, the complete POV for a dashboard component may get split between the local POV and the global POV bar. The global POV bar and the local POV together contain the complete intersection information for each form on a dashboard.

If there is only one component on the dashboard that uses a form as a data source, then the entire POV/page of the form is moved to the global POV bar.

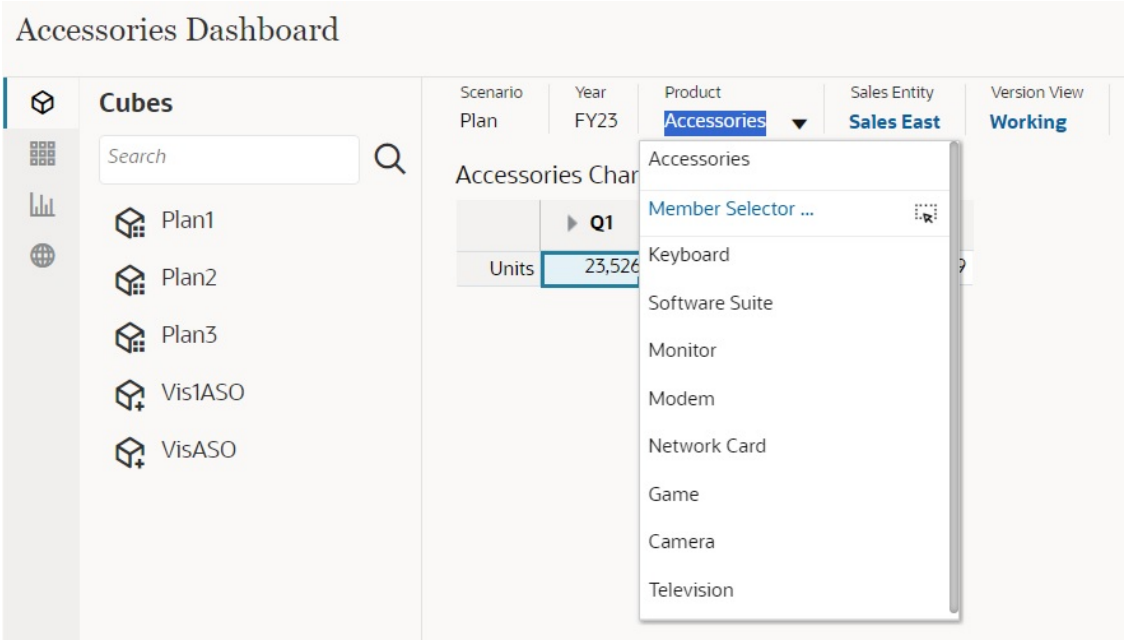
If there is more than one component on a dashboard that uses forms as a data source, then this is how the application determines which dimensions go in the global POV bar or stay in the local POV:

- If the dimension is in the POV or page of all the forms, and the member selection is the same in all the forms, the dimension goes in the global POV bar.
- If the dimension is in the POV on one form and in the page of another form, then the dimension stays in the local POV.
- If the dimension is in the POV, then the same members must be selected in all the forms for the dimension.
- If the dimension is a page dimension, then the selected page members must be the same and display in the same order in all the forms.

Searching for POV Members in Dashboard 2.0 Dashboards

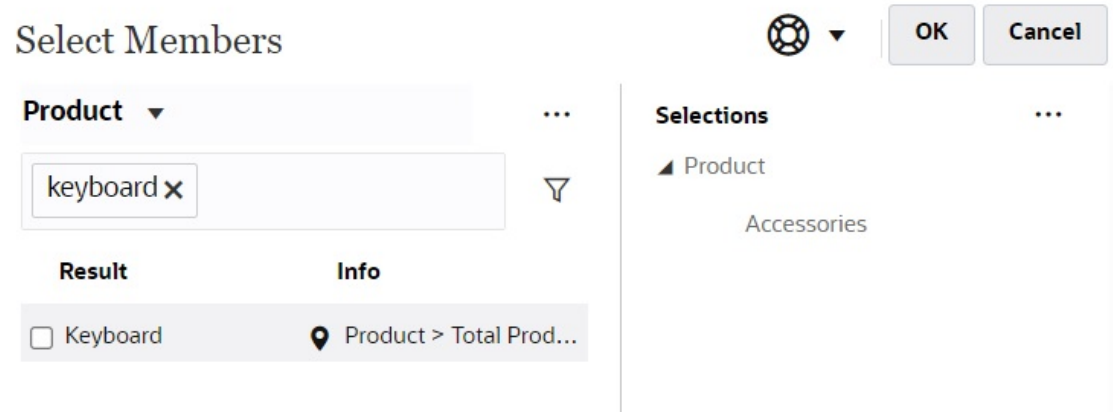
When working in Dashboard 2.0 dashboards, you can easily find new POV members. To search, click a POV link. The search interface displays a list of the most recently used members from which you can choose.

Figure 10-1 Dashboard 2.0 POV Most Recently Used Members List



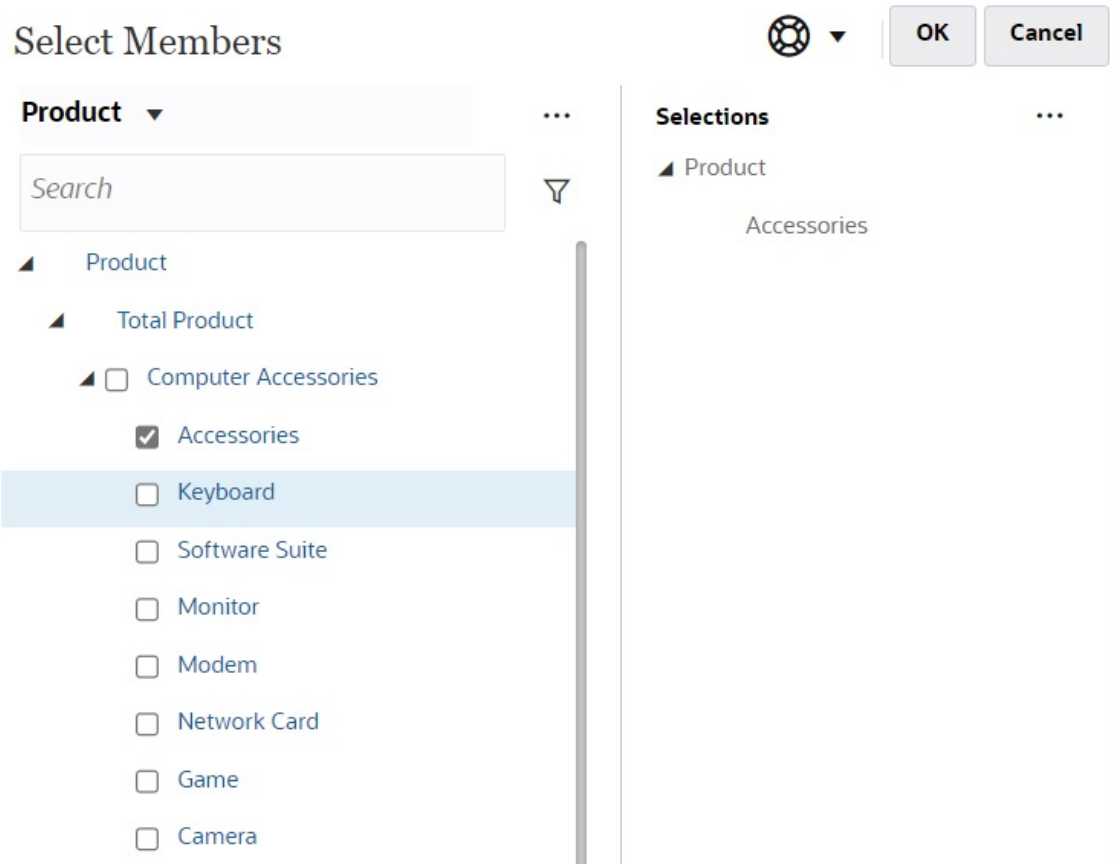
You also can search using the member selector. Click a POV link, then click the member selector icon to launch the member selector. Members in the member selector are displayed as a hierarchy or you can type a member name in the **Search** box. If a member is located using the **Search** box, you'll see a result displayed under the Search box.

Figure 10-2 Dashboard 2.0 Member Selector Search Results



You can view where the member is located within the hierarchy by hovering over the result's **Info** column and clicking .

Figure 10-3 Dashboard 2.0 Member Selector Search Results in Hierarchy



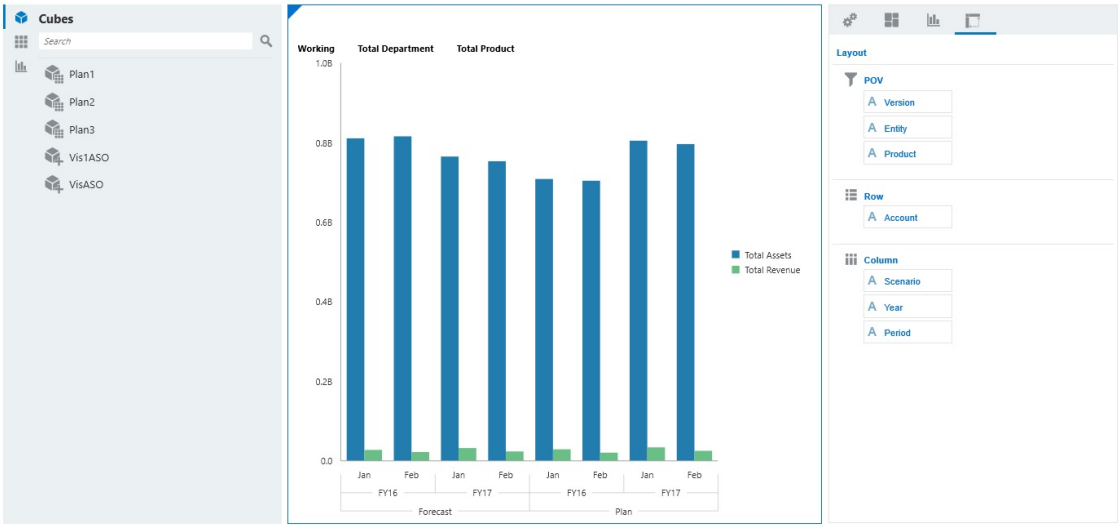
Dashboard 2.0 POV Bar Considerations

- For POV bar Page drop-down lists, the number of members shown in the page drop-down is controlled in **User Preferences** under **Page Options**. (From the Home page, click **Tools**, then **User Preferences**, and then click **Display** to view the **Page Options**.) If the number of items in the page drop-down exceeds this limit, then users cannot view them. Note that this will be fixed in a future release with the ability to use the member selector to browse through the rest of the members, similar to Dashboards 1.0.
- If your dashboard includes multiple forms, but some forms have different properties for the same dimension; for example, the Version dimension is hidden on some forms but it's visible on other forms in the same dashboard, the global POV might not display any Version dimension members. We recommend you ensure that all properties for all dimensions are the same for all forms included within a dashboard.

About Quick Analysis

Quick Analysis lets you add components to dashboards without having a prebuilt form. You create an ad hoc query by typing member names in the Search bar or using the Member Selector, and then you control the layout of the query in the Layout tab of the Properties panel.

Example Dashboard 2.0 Quick Analysis Chart in Design Mode



To create a quick analysis in your dashboard:

1. Select a cube for the query. You can do this in one of two ways:
   - Click in the workspace Search box to display a list of cubes, then select a cube.
   - Drag and drop a cube from the object palette to the dashboard workspace.

> **Note:**
>
> With either method, the system starts with a default query; a single intersection in the cube driven from your Most Recently Used, or MRU. The default query varies based on what you've recently been looking at in the cube.
>
> If the MRU intersection in the cube is invalid, you'll get an error. If you see an error, open another form and change the page to a valid intersection so that the MRU gets updated with a valid intersection.

2. Refine the query by specifying members for the query:
   - In the Search box, type member names to display an auto-complete list from the cube.
   - Launch the member selector to select members, and then click **OK**.

> **Note:**
>
> When using the Search box to find members for quick analysis and more than 25,000 members are present in a cube, the Search box only searches the first 3000 members in the dimension for the cube. If the cube contains less than 25,000 members, then the Search box will search through all members. To search a cube with more than 25,000 members, launch the member selector to make your quick analysis selections.

3. Select a chart type for the query.

   Click to select a chart type; for example, Bar .

4. In the Properties panel, click the Layout tab .
5. Drag and drop the dimensions in the Layout tab of the Properties panel to other sections of the Layout tab to change the POV, Row, and Column layout of the dashboard component.

Videos

Your Goal	Watch This Video
Discover how to add quick analysis components to dashboards without having a prebuilt form. You learn how to create an ad hoc query in the Dashboard Designer and format the results by changing the dimension layout and applying visualizations.	Creating Quick Analyses in Dashboards 2.0

11

Designing Infolets

Create infolets to allow users to view and interact with high-level, essential information from different sources. This helps them quickly assess where to direct their attention.

Related Topics

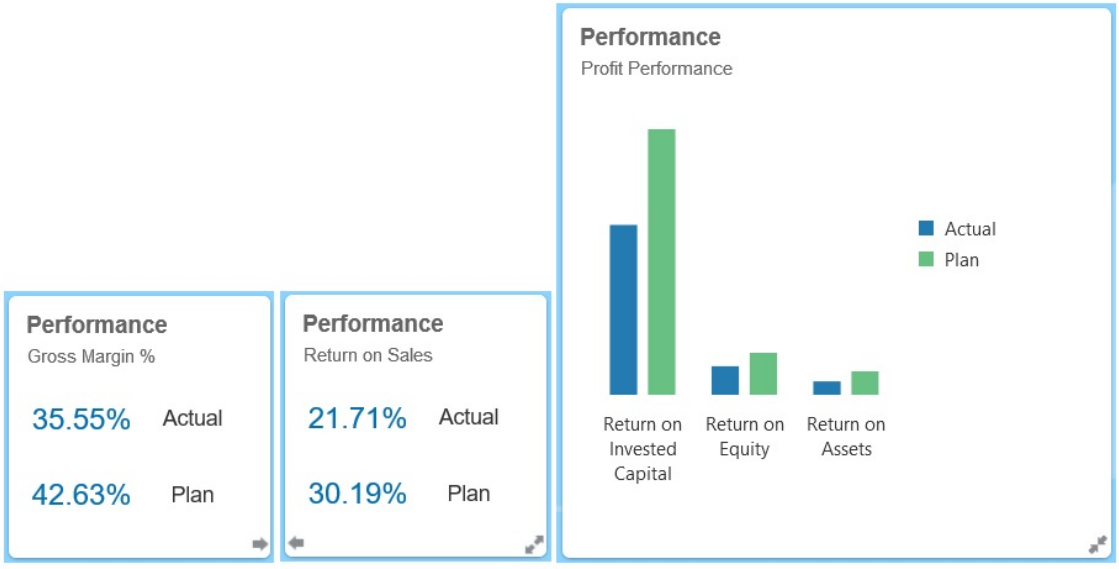
- About Infolets
- Anatomy of an Infolet
- Determining Infolet Content
- Using the Infolets Designer
- Creating Infolets
- Working with Infolets
- Customizing the Interface to Access Infolets

About Infolets

Infolets enable users to view and interact with high-level, essential information generated from different sources so that they can quickly assess where to direct their attention. Service Administrators create, redesign, delete, and assign permissions to infolets.

What is an Infolet?

An infolet is a self-contained, interactive box-shaped container used to display information using text and charts. Infolets are interactive and use progressive disclosure to display high-level, aggregated, essential information for quick consumption at a glance, and then can be acted upon as needed. Infolets can be flipped and resized to display up to three charts or sets of values.



For more information about infolets, see Anatomy of an Infolet.

How Can I Use Infolets?

Use infolets to:

- Promote essential, easily consumable information
- Provide answers to your most critical questions:
  - What is new or what has changed?
  - What is the most important information that supports my work?
- Group key information by user role in a way that helps users quickly assess and prioritize their work
- Progressively display essential details and actions

  Display these additional details in the various infolet views accessed by flipping or expanding an infolet. However, a single infolet view is acceptable.
- Provide a visually rich means of displaying essential or summary information

Do not use infolets to feature highly complex information, such as reporting functions, or to present detailed visuals.

See Determining Infolet Content.

What is an Infolet Page?

An infolets page is a page that contains one or more infolets. It houses a container that manages the space occupied by the infolets and rearranges them depending on the size of the browser and the size of the infolets. Each infolet you create belongs to an infolet page. The **Infolets** card on the Home page provides a list of infolet pages.

> **Note:**
>
> Not all features pictured in the preceding image are supported in this update. Oracle plans to support these features in a later update.

See Working with Infolets.

Anatomy of an Infolet

Infolet Views

An infolet supports up to three views:

1. Front view (required)

2. Back view (optional)

3. Expanded view (optional)

The front view is required, and:

- Provides a quick look or glimpse at high-level information that has a direct effect on your work; for example, the front view can display status, counts, totals, or the most recent updates
- Promotes a glancing action that helps you identify important information that you may want to explore a bit more
- Uses all infolet sizes except 3x2 (see information about infolet sizes below)
- Returns the expanded view to its original size in the front view or back view
- Includes an Actions menu icon available only on hover and either a (optional) flip to back icon or an expand icon in the lower-right corner

> **Note:**
>
> If only one view is presented, it must be a front view.

The back view is optional, and:

- Presents analytical information (for example, a graph)
- Promotes a scanning action that helps you explore or become more familiar with the information represented on the front view
- Is sized the same as the front view
- Includes an Actions menu icon available only on hover, a flip to front icon in the lower-left corner, and an (optional) expand icon in the lower-right corner

The expanded view is optional, and:

- Presents even more detailed information about the single data point or interdependent data set presented in the front and back views; for example, the expanded view can display more details about an object or a list of recent items than what's displayed on either the front or back view

- Provides enough information to help you decide if you're ready to take action and move to a focused context on a work area page
- Transitions smoothly from other views. You see a smooth, seamless expansion, as one infolet pushes others into new positions.
- Must be sized bigger than the front or back views
- Includes an Actions menu icon available only on hover, and a collapse icon in the lower-right corner

Infolet views honor the access permissions assigned to the underlying forms and dimensions. Therefore, the same infolet may display varying views from user to user if they have different access permissions.

Infolet Sizes

Infolets can be sized as follows:

> **Note:**
>
> 1x1 refers to a box that spans one column and row width (170 pixels).

- 1x1
- 2x1
- 3x1
- 2x2
- 3x2 (expanded view only)

The size of the front and the back views are always the same. Changing the size of the front view will automatically reset the size of the back view. Because the size of the expanded view must always be greater than the size of the front/back views, if the size of the front/back view of an infolet is enlarged, the expanded view automatically resets to a larger size than the front/back view.

> **Note:**
>
> Front and back views cannot use the 3x2 size. This size is applicable for the expanded view only.

The size, title, and subtitle of a view is set by the designer in the properties panel. See Using the Infolets Designer.

Navigating Between Infolet Views

An infolet can be created with one of the following view combinations:

1. Front view only
2. Front and back views
3. Front and expanded views
4. Front, back, and expanded views

Switching from one view to another is controlled by clicking a flip icon, an expand icon, or a collapse icon available in the bottom-right or bottom-left corner of the infolet. Hovering your cursor over the bottom corners reveals a flip, expand, or collapse icon.

Determining Infolet Content

Consider the following general tips when determining infolet content:

- Look to existing dashboards and work area pages.

  Dashboards and work area pages are excellent starting points because they display collections of information summaries.

- Apply the 10/90/90 principle.

  Seek out the most essential, easily consumable information that addresses frequently asked questions garnered from the top 10 percent of use cases. These use cases can come from across your enterprise—from business intelligence, social, transactional, external, and so on.

  Then focus this information to reveal what 90 percent of your users would benefit from viewing 90 percent of the time. You can apply this 10/90/90 percent principle to existing dashboard content, to existing work area page content, or generally, to percolate eligible information for an infolet.

- Restate your top use cases in the form of frequently asked business questions.

  Present the corresponding infolet content in such a way as to answer these business questions; for example, how many orders are in jeopardy, listed by status?

- Look for one point or a tightly related, interdependent set of points, instead of multiple points of information.

  The process of determining content for an infolet is similar to the process that is used to yield dashboard content—but to a deeper level of analysis. Look for information within a data point or data set that is suitable to be displayed in no more than three views of information hierarchy and that answers a critical business question.

- Start with the single most important point.

  An infolet displays aggregated information about a single data aspect or point of information in relation to an event that the user needs to know about or a task that the user needs to address.

  If a dashboard contains multiple aspects about one or more objects (for example, numeric totals and currency totals), start with the single most important point and add that as a simple overview (for example, as a total using a stylized numeric value) to the front view of an infolet. Then determine the content for the back view, if needed. Finally, determine the content for the expanded view, if needed.

  An infolet should have no more than three views. If there is only a single data point or only one tightly related, interdependent data set to display on an infolet, use only the front view.

Related Links

Designing Forms for Infolets

Designing Charts for Infolets

Designing Forms for Infolets

Use only forms with small data sets in infolets. Permissions that are set for forms are honored in infolets.

You can create forms that are specifically used in infolets:

- Forms used in infolets should have fewer cells than a traditional data entry form; for example, forms used in infolets should only have up to 12 cells.
- Forms used in infolets should have no more than 12 rows and columns. If a form has more than 12 rows and columns, the infolet will only display the first 12 rows and columns.
- Infolets currently don't support Page dimensions or POVs, therefore forms used in infolets should not contain Page dimensions.
- If a form that is used in infolets contains a grid with members that expand, the infolet will display all the members in the form, including the expanded members.

Designing Charts for Infolets

Use the title and subtitle in charts to show static context.

There are six types of charts that you can use in infolets:

- Bar: Displays a graphical summary of multiple data values for comparison purposes. Bar charts can be plotted vertically or horizontally. Up to eight bars are recommended for bar chart infolets.
- Column: Displays stacked bars that represent different data sets on top of each other. The height of the resulting bar shows the combined result of the data sets.
- Doughnut: A circular graph which is divided into segments to compare data sets to one another. The blank center displays the sum of all data sets. Up to six segment values are recommended for doughnut chart infolets.
- Line: Use to visualize a trend in data over intervals of time.
- Pie: A circular graph which is divided into slices to compare data sets to one another. Up to six slices are recommended for pie chart infolets.
- Tile: Lets you select specific values from a data set to display. No more than three values are recommended for tile chart infolets.

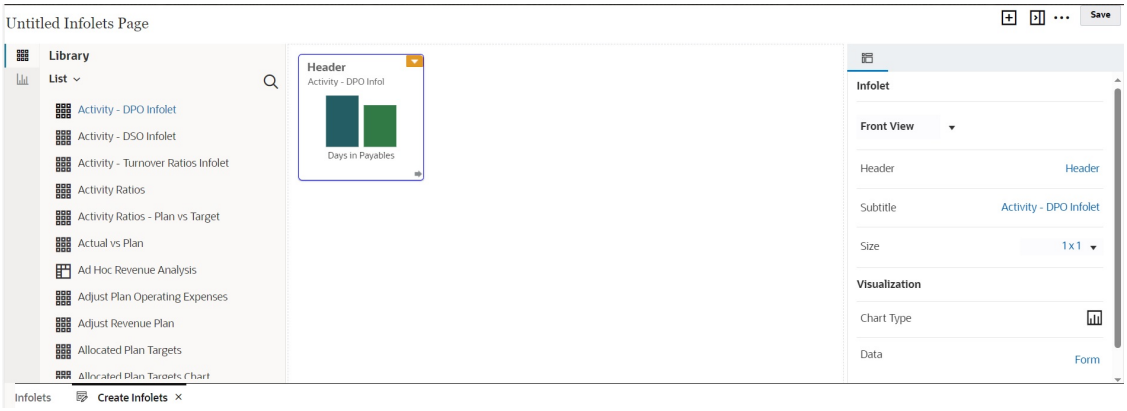
> **Note:**
>
> Tile charts can only use the 1x1 size. You cannot resize an infolet using the tile chart until the chart type is changed. If you drag and drop a tile chart to an infolet that is greater than 1x1, you will be prompted to change either the size of the infolet or the chart type.

Using the Infolets Designer

Service Administrators use the Infolets Designer to create infolets and infolet pages. The Infolets Designer enables easy toggling between runtime and designer views. To access the Infolets Designer, launch the **Infolets** icon on the Home page and click **Create**, or click the

Actions icon next to an infolets page in the listing, and then click **Edit**. Clicking the infolet name in the listing, launches the runtime version of the infolet page. You can toggle from runtime view to designer view by clicking .

Infolets Designer



Infolets Toolbar

On the upper right is the Infolets Toolbar.

: Adds a new infolet to the Infolets Designer

: Hides and unhides the Properties panel

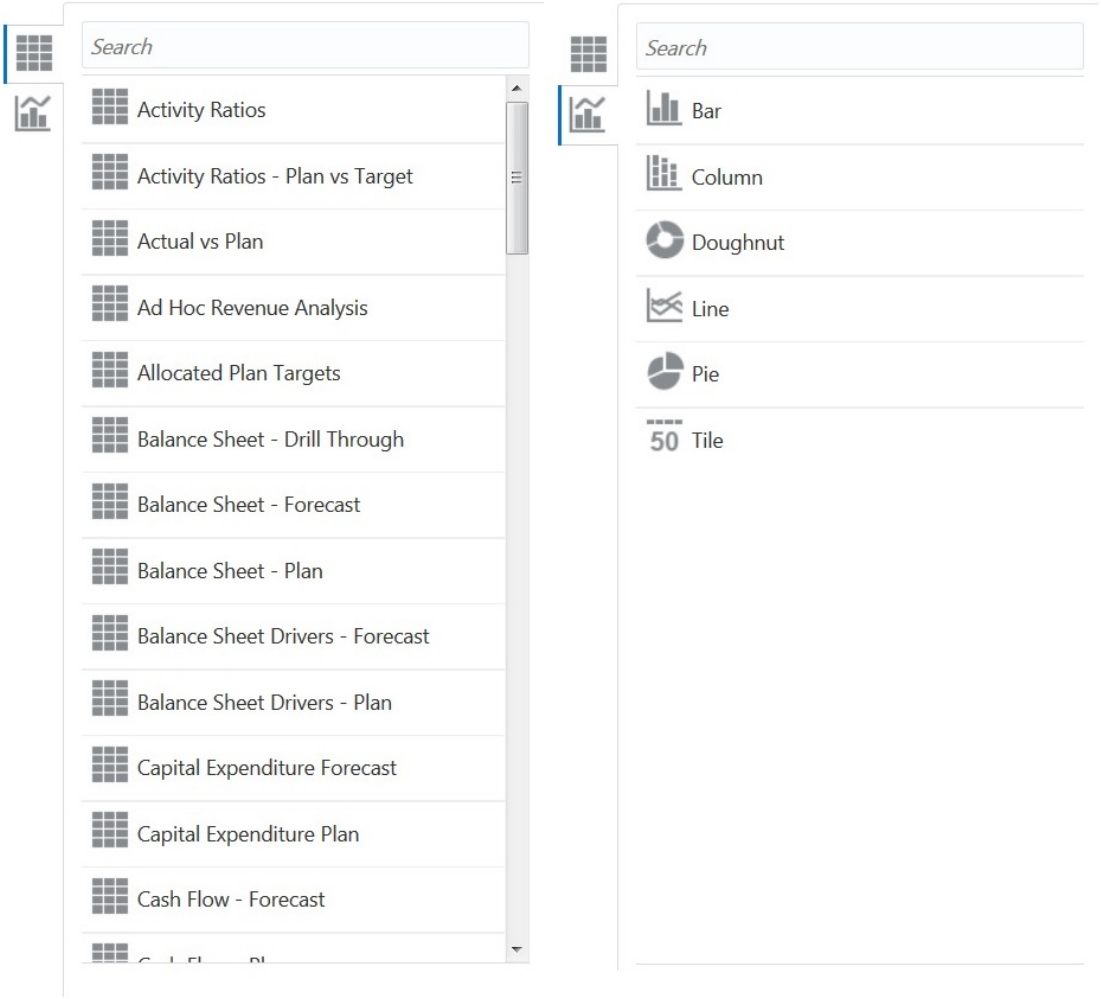
: Click to perform these actions:

- **Reset**: Resets the Infolets Designer to a previously saved state
- **Refresh**: Refreshes the data from Essbase and updates the infolet definition from the database
- **Runtime**: Hides all Infolets Designer elements and displays the infolet as it would appear to users during runtime

: From runtime mode, displays the Infolets Designer

Infolets Designer Palette

On the left is the Designer Palette. The Designer Palette has two tabs: Forms and Chart Types. Highlight objects in the Designer Palette and then drag and drop them onto the drop zone.



Designer Palette objects:

- Forms: Select forms to include in the infolet by scrolling through the forms or by searching for them by name. For details about forms, see Designing Forms for Infolets.

> **Note:**
>
> The access permissions set for forms are honored in infolets.

- Chart Types: Select the chart types to include in the infolet. Infolets display sample data in the charts until you associate the chart with a form as its data source. When you link a chart to a form, users can see the impact of changing data in the form on the associated charts. To associate a chart with a form, highlight a chart and drag and drop it onto the drop zone, in the Properties panel, click **Sample**, and then click **Forms** to select the data source. For details about charts, see Designing Charts for Infolets.

Infolet Menu

The infolet menu contains the infolet delete and clear actions. To view the infolet menu, hover over the upper right corner of the infolet, and then click the down arrow to display the menu options:

- **Delete**: Removes the infolet from the page
- **Clear**: Clears infolet details

Properties Panel

The properties panel on the right side of the Infolets Designer enables you view and work with these infolet properties:

> **Note:**
>
> The header you specify is the same for all views of an infolet, but you can specify a different subtitle for each view; for example, the front, back, and expanded views of an infolet can each have different subtitles, but they must have the same header.

- **Header**
- **Subtitle**
- **Size**: Displays the infolet in the selected size
- **Chart Type**: Displays the infolet data as the selected chart type
- **Data**: Displays the associated data source (Sample or Form)
- **Form**: Displays the selected infolet form

Actions such as delete and clear are on the infolet menu.

By default, the front view of an infolet is displayed in the properties panel. You can see the other views by selecting them from the drop-down. If you flip or expand an infolet to view the back or expanded views, the properties for those views display in the properties panel. In addition, the properties for the corresponding chart type are also displayed in the properties panel.

Creating Infolets

To create infolets:

1. From the Home page, click **Infolets**, and then click **Create**.
2. Click **Untitled Infolets Page**, and enter a title for the new infolets page you are creating.
3. From the designer palette on the left, choose either the **Library** tab or the **Visualizations** tab, highlight an object, and then drag and drop it onto the infolets drop zone.

   Alternatively, you can click

   at the top of the page to add a new infolet to the workspace.
4. If the **Properties** panel isn't already displaying, click

   at the top of the page to reveal it. Customize the highlighted infolet using selections made in the **Properties** panel.
5. Click **Save**.

You can easily duplicate and modify an infolet using the **Copy As** action on the Infolet list page. From the list page, click the Actions icon next to the infolet you want to copy, and then click **Copy As**.

Working with Infolets

Once you've created an infolets page, it is displayed in the infolets listing on the **Infolets** card.

The listing page for infolets supports folders. Folders enable you to assign permissions to all infolets within a folder rather than assigning permissions to each individual infolet. The infolets listing page uses the same folder hierarchy as dashboards and data entry forms and all artifacts and folders reside under a root folder called **Library**.

> **Note:**
>
> Only administrators can add files (for example, dashboards, infolets, forms, reports, and so on) to the **Library** root folder.

To view and work with infolets:

1. From the Home page, click **Infolets**.
2. To work with infolets, perform an action:
   - On the listing page for infolets, you can toggle between viewing infolets by a flat view or a tree view: 

     Then you can search for infolets using **Search**  . The flat view displays only the artifacts that meet the search criteria, not the folders that contain them. The tree (or hierarchical) view displays artifacts in the context of the folders that contain them.

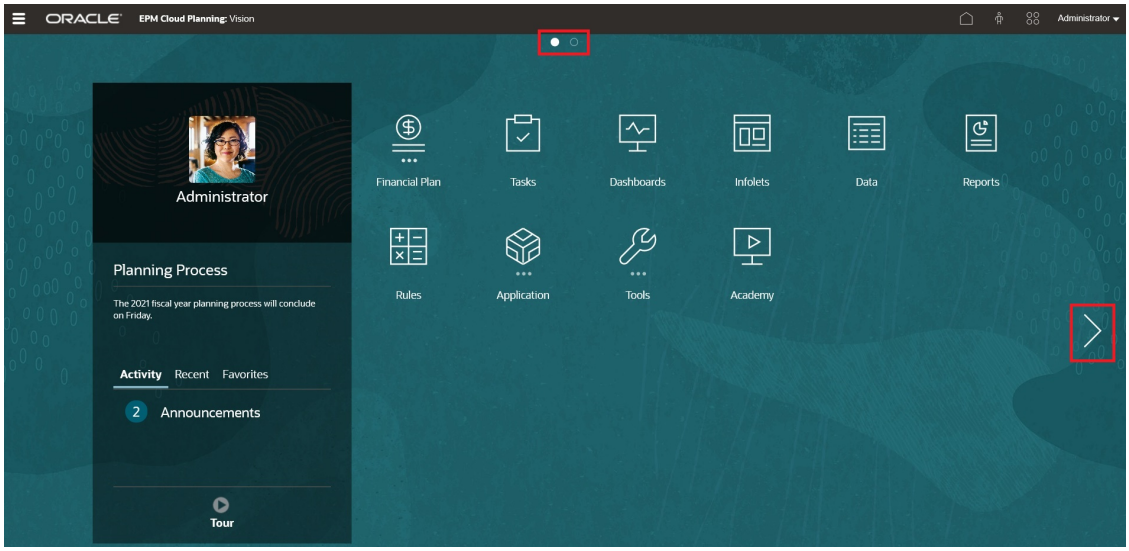
     To search on another keyword, clear the search criteria by clicking **X** in the **Search** box.
   - To refresh the infolets listing, click **Refresh**.
   - To create infolets, click **Create**. See Creating Infolets.
   - To perform the following actions on infolet pages, click the **Actions** icon next to the infolet page, and then select:
     - **Create Folder**: Creates a folder in the listing
     - **Edit**: Opens the infolet page in the Infolets Designer
     - **Rename**: Renames the infolet page
     - **Copy As**: Duplicates an infolet page
     - **Delete**: Deletes the infolet page from the listing
     - **Move To**: Moves an infolet page to another folder
     - **Default** or **Unmark**: **Default** marks an infolet page as default and makes it accessible directly from the Home page by clicking the second infolet dot that is displayed beneath the global header on the Home page. Only one infolet page can be marked as default, and a "(Default)" prefix displays before the name of that infolet in the listing. **Unmark** removes the default designation from the infolet page.

> **Note:**
>
> You can no longer mark a dashboard page as default. Only an infolet page can be marked as default.

– **Assign Permission**: Enables you to assign Read, Write, and None access permissions to infolet pages and folders for individual users or groups

Customizing the Interface to Access Infolets

You can customize the business process interface to add links to infolet pages from the Home page using the Navigation Flow Designer. When you customize your interface to access infolet pages, dots will appear on the Home page beneath the global header and arrows will display to help you navigate easily between the Home Page and infolet dashboard pages. Each dot that appears on the Home page represents an infolet page and hovering over each dot displays the name of the infolet page. Clicking an infolet dot launches the infolet page associated with that dot. You can define up to seven infolet dots on the Home page. If you've created connections to other EPM Cloud environments, you can also add links to infolet pages in other EPM Cloud environments.



Users will only see dots displayed on the Home page for infolet pages to which they have access. The types of infolet dots displayed are as follows:

- Home dot: This dot always appears first and it links to the Home page. There can only be one home dot. If you are not viewing the Home page, clicking the home dot will bring you back to the Home page.
- User dot: Links to an infolet page marked by an end user as the default infolet page. There can only be one user dot and it always appears after the Home dot on the user's Home page. User dots cannot be added using the Navigation Flow Designer. For more information about marking an infolet as the default, see Working with Infolets.
- Customizable dot: Links to infolet pages created by Service Administrators. Customizable dots can be integrated into navigation flows and their visibility and the order in which they

appear is determined by the navigation flow designer. There can be up to seven customizable dots and they always appear after the home and user dots.

To add infolet dots to your business process interface using the Navigation Flow Designer:

1. From the Home page, click **Tools**, and then click **Navigation Flows**.
2. Select an inactive navigation flow in the listing, and then click the **Infolets** tab on the Navigation Flow Designer.
3. Click  .
4. In **Manage Infolet**, name the infolet dot, set visibility, and then click  to select an infolet in the Artifact Library.

   > **Note:**
   >
   > You can select an infolet from another EPM Cloud environment if you've created connections to other environments. First select the environment under **My Connections**, and then navigate to the infolet in that environment.

5. Click **Save and Close**.

> **Note:**
>
> Infolets can also be associated with a tab or a card in a navigation flow. While adding or updating a tab or a card, select an infolet in the Artifact Library.

To view design time changes to the navigation flow, activate the navigation flow, and then from the Home page, click the down arrow next to the user name (upper right corner of the screen) and click **Reload Navigation Flow**.

To learn more about designing navigation flows, see Designing Custom Navigation Flows.

To learn more about connecting EPM Cloud environments and for information about the **Copy URL** feature, see Connecting Environments in EPM Cloud.

12

Bring Your Own ML: About Machine Learning Model Import

With Bring Your Own ML, EPM administrators can import a fully trained Machine Learning (ML) model and deploy it to a FreeForm application. Planners can then leverage robust, ML-based forecasting that uses advanced predictive modeling techniques to generate more accurate forecasts.

Data scientists gather and prepare historical data related to a business problem, train the algorithm, and generate a PMML file (Predictive Model Markup Language, a standard language used to represent predictive models) using a third party tool. These predictive analytic models and machine learning models use statistical techniques or ML algorithms to learn patterns hidden in large volumes of historical data. Predictive analytic models use the knowledge acquired during training to predict the existence of known patterns in new data.

EPM administrators can then import and configure the fully trained ML model, which generates two Groovy rules. Adminstrators attach the rule to a form or dashboard, or schedule a job to generate prediction results on a regular basis. This puts the benefits of machine learning and the power of data science into the hands of business users, enhancing the planning and budgeting process and leading to better business decisions.

For example, you can predict product volume for an entity, using key drivers such as average sales price, planned spend on promotions and advertising, historical volumes, and estimated industry volumes.

You can import ML Models and use them to predict numeric values in other finance use cases, for example:

- Trade promotion impacts on sales uplifts
- Marketing mix modeling to drive better ROMI
- Internal and external driver impacts on revenue forecasts
- Predictive cash forecasting for better cash position

Overview of Steps

Prerequisite: Data scientists build and train the ML model in a data science tool (any third party tool or Oracle Data Science Cloud) and save it as a PMML file.

Next, EPM administrators put the model to work to get business value from the trained model:

1. Administrators import the ML model in PMML format to a FreeForm application and define how the input variables and target variable maps to dimension members or cell values in the FreeForm application. This step generates automatic Groovy rules that connect the ML model to the FreeForm application. Two Groovy rules are generated for each ML model definition: one rule to associate with a form or dashboard, which allows users to make predictions on demand, and another to generate large scale predictions in a scheduled job for bulk processing. See Importing an ML Model.
2. Administrators deploy the ML model in a FreeForm application by associating the Groovy rule to relevant action menus, forms, or dashboards. See Deploying an ML Model to Planners. Administrators can also create a job to run the Groovy rule in a batch process.

3. Planners leverage ML-powered business rules in forms to generate predicted values, which are saved on the form. Planners can perform what-if analysis using the generated predictions, or modify predicted values on the form. Planners add value with their expertise and judgement, and then finalize the forecast.
4. This is an iterative process. As planners make predictions based on the ML model, administrators can measure the performance of the model, and can work with data scientists to update or replace the ML model when needed. Then, administrators re-import and deploy the retrained ML model.

   When you re-import the retrained ML model, the Groovy rules are regenerated.

Videos

Your Goal	Watch This
This overview introduces you to Bring Your Own ML (Machine Learning), where EPM Administrators can import a fully trained ML model and deploy it to a FreeForm application. Planners can then leverage robust, ML-based forecasting that uses advanced predictive modeling techniques to generate more accurate forecasts.	Overview: Bring Your Own Machine Learning (ML)
Learn how to configure ML model import for Bring Your Own Machine Learning. You import a fully trained ML model into FreeForm. You follow the steps in a wizard to map, analyze, and test the model. After saving the model, two Groovy rules are created. To complete the integration process, see the related video for deploying an ML model to FreeForm.	Configuring Machine Learning (ML) Model Import
Learn how to deploy an ML Model to FreeForm. After configuring the ML Model Import, you integrate the PMML file into your application by creating an Action Menu with the Groovy rule generated from the configured ML model. Then you associate the Action Menu with a form. When planners run the rule from the form, the rule returns the set of predicted values.	Deploying a Machine Learning (ML) Model to Planning

Considerations for Bring Your Own ML

The following are some of the considerations for Bring Your Own ML.

Considerations:

- Bring Your Own ML requires an EPM Enterprise Cloud service subscription; it is not available in EPM Standard Cloud. It is also available with Enterprise Planning and Budgeting Cloud and FreeForm.

  In addition, you can use Bring Your Own ML with Oracle Sales Planning Cloud and Oracle Strategic Workforce Planning Cloud.
- Bring Your Own ML works with Custom, FreeForm, Modules, Sales Planning, Strategic Workforce Planning, and Cash Forecasting application types.
- Bring Your Own ML requires Groovy rules.

- Bring Your Own ML is available only with Redwood Experience enabled.

Importing an ML Model

Import a fully trained ML model into a FreeForm application to prepare it for use by business users.

Prerequisite: Before you can import the ML model, the data science team must build, train, and save the ML model as a PMML file.

To import an ML model to a FreeForm application:

1. From the Home page, click **IPM** and then click **ML Models**.
2. Click **Import**, and then drag and drop the PMML file, or browse to it and select it.
   On the **Import Model** page, you see information about the PMML file, such as the target column (the variable to be predicted using the ML model) and the training date.
3. Enter a model name and description, and then click **Next**.
4. On the **Generate Rule** page, enter information that will generate a Groovy rule to associate with forms or dashboards:
   In **Model Mapping**, select the cube where the ML model will be used and define the scope of data in which to use the ML model by selecting a member or set of members from each dimension.
5. Map **Input** and **Output** to the appropriate dimensions and members in the cube and then click **Next**.
   The **Input** and **Output** sections contain the list of input features (features/columns that are used to make predictions) and target feature (column that is expected to be predicted). FreeForm analyzes the PMML file to generate the list of inputs and outputs.

   Input features are independent variables, similar to drivers, that act as input to your system. When you make predictions, the model uses input features to predict your output. In this step, you map the input from the ML model to the output in the FreeForm cube. **Input** describes how to extract the data from the ML model. **Output** defines the target measure you want to predict and where to paste the predicted values in the FreeForm application.

   For example, product, price, and industry volume, the input features, might be used to predict volume, the output.

   - In the **Input** area, for each input feature, select an **Input Type** and if you select **Cell Value** or **Member**, select the members or dimensions in the Planning application to map to. Input types:
     - **Prompt**: If you don't have a member or dimension in FreeForm that maps to this input value from the ML model, when the predicion is made, prompt the user to enter an estimate for the value.
     - **Cell Value**: Map an input feature to one or more dimension members in the FreeForm cube. For example, the input feature called Price maps to an account member called Price in the FreeForm application.
     - **Member**: Map an input feature to a dimension in the FreeForm cube. For example, the Input feature called Product maps to the Product dimension in the FreeForm application.
   - In the **Output** area, select an **Input Type** and if you select **Cell Value** or **Member**, select the members or dimensions in the FreeForm application to map to to store prediction results.

6. In **Analyze Model**, review the ML model and then click **Next**.
   This step represents MLX (Machine Learning Explainability), and extracts additional information about the ML model. For example, review Regression Coefficients to see how the relative impact of key input features is used to predict the output. The height of the bar represents the incremental effect of one unit increase in an input feature on the target variable.
7. In **Test Model**, test the ML model by generating a prediction for a set of sample values. For each **Input**, enter a sample input value and then click **Predict**.
8. Review the predicted **Output** value, and then click **Save and Close**.
9. Click **Yes** to confirm the creation of Groovy Rules.

Two Groovy rules are generated for each ML model definition:

- `ML_MLModelName_Form`: Use this rule to associate with a form or dashboard, which allows users to make predictions on demand.
- `ML_MLModelName`: Use this rule to generate large scale predictions in a scheduled job for bulk processing.

You can review the generated rules in Calculation Manager. The Groovy rules define the name and location of the PMML file, along with input and output based on the mapping you defined. For more information on using these generated Groovy rules, see Deploying an ML Model to Planners.

Tutorials

Tutorials provide instructions with sequenced videos and documentation to help you learn a topic.

Your Goal	Learn How
Learn how to import a fully trained ML model and deploy it to a FreeForm application. Planners can then leverage robust, ML-based forecasting that uses advanced predictive modeling techniques to generate more accurate forecasts.	 Importing ML Models

Deploying an ML Model to Planners

Deploy an imported ML model to make it available to planners on forms and dashboards. Planners can use machine learning to generate a prediction.

After importing a PMML model, EPM administrators integrate the PMML file into the FreeForm application by associating the generated Groovy rule with FreeForm forms.

You can first review the Groovy rules in Calculation Manager to validate the mappings and make changes if needed. When the Groovy rule is finalized, you can associate it with relevant forms or dashboards.

Make the Groovy rule available to planners by creating an Action Menu that you associate with a form.

To associate an ML Groovy rule with a form:

1. Create an Action Menu item:
   a. From the **Navigator**, under **Create and Manage**, click **Action Menus**.

b. Click , enter the menu's name, and then click **OK**.

c. Select the new menu item, click , and then click **Add Child**.

d. Enter a **Menu Item** name, and give it a label, for example, **Predict Volume**, of type **Business Rule**, and select the appropriate cube and Groovy rule with the `Form` suffix, for example: `ML_MLModelName_Form`. Then click **Save**.

For more information, see Creating and Updating Action Menus.

2. Associate the action menu with a form:

a. From the **Navigator**, under **Create and Manage**, click **Forms**.

b. Navigate to the form, click , and under **Other Options**, click the Action Menu item you created. Associate the menu with the form by moving it from **Available Menus** to **Selected Menus**. Then click **Save**.

For more information, see Administering Forms.

When a planner selects the Action Menu item from the form, it triggers the Groovy rule. The rule picks the data for all input drivers, sends it to the PMML file for processing, returns the set of prediction values, and pastes them to the output, as defined in the **Import Model** wizard. The rule runs in the context of the form, predicting values only for cells on the form. Security is honored so that planners see predictions only for intersections to which they have access. Planners can run through various what-if scenarios to adjust their forecasts and plans, or adjust the predicted values.

Note that planners must have access to the model, form, rule, and members in order to run the rule.

Running an ML Groovy Rule as a Batch Job

You can run the ML Groovy rule as a job. The batch rule runs for all cells for the full scope of the model, not just for cells on a particular form. You can set it up to run as a recurring job, for example, to load predictions so they are available to planners on a regular basis.

To run or schedule a batch job, in **Jobs**, select a **Job Type** of **Rules**, and then select the Groovy batch version of the business rule for the PMML model—`ML_MLModelName`.

For information about scheduling jobs, see Scheduling Jobs.

Tutorials

Tutorials provide instructions with sequenced videos and documentation to help you learn a topic.

Your Goal	Learn How
Learn how to take the Groovy rule created from an imported ML Model, and add it to a form. Then you make predictions with the ML model in FreeForm.	Making Predictions with ML Models

13

Defining Valid Intersections and Cell-Level Security

Define rules that restrict who can enter or view data in your application.

You can restrict who can enter data in your application by creating rules that mark certain member intersections as valid (or invalid) for data entry. These rules are called valid intersections. You can also restrict who can view data in your application by creating rules that remove read or write access to cells that a user would normally have access to due to their regular security. These rules are called cell-level security.

- To define valid intersections, see Defining Valid Intersections.
- To define cell-level security, see Defining Cell-Level Security

Defining Valid Intersections

Define valid intersection rules which filter certain cell intersections to users when they enter data or select runtime prompts.

Related Topics

- Understanding Valid Intersections
- Creating Valid Intersections
- Managing Valid Intersections
- Suppressing Invalid Data in Forms
- Clearing Invalid Data
- Working with Valid Intersections

Understanding Valid Intersections

Valid intersections are cell intersections that are filtered based on rules you define, called valid intersection rules, which filter certain cell intersections to users when they enter data or select runtime prompts. For example, you can specify that certain programs are valid only for some periods or departments. After valid intersections are defined, cells containing invalid data are read-only. This restriction speeds the planning process and optimizes the information available to users.

Conversely, you might have a use case where data entry is allowed in most cell combinations and you only need to prevent data entry to a select few cell combinations. In this case you can ease the definition process by defining invalid intersections. You define invalid intersections in the same way that you define valid intersections except you can specify the invalid intersection definition type while creating the intersection group.

To better understand how valid intersections affect behavior in forms and in runtime prompts, see Working with Valid Intersections.

Videos

Your Goal	Watch This Video
Learn how to manage valid intersections.	 Managing Valid Intersections in Cloud EPM

Related Topics

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

Valid Intersection Groups

Valid intersection groups define:

- Dimensions to be included
- One of those dimensions as the anchor dimension
- Whether nonanchor dimensions are required or not
- Whether the anchor dimension members not specified or referenced will be valid or invalid

Valid Intersection Rules

Valid intersection rules:

- Must use the same dimensions that were defined within their valid intersection group
- Valid intersection rules within the same valid intersection group that produce an apparent conflict or overlap, are marked valid if either valid intersection rule condition is met
- Valid intersection rules in different valid intersection groups that produce an apparent redundancy or overlap, are marked valid if they satisfy the requirements of all valid intersection groups

  Thus, if any valid intersection group marks an intersection invalid, regardless of other valid intersection groups making it valid, the system will mark the intersection invalid. Invalid groups override valid group results.

> **Note:**
>
> If you want to remove valid intersections regardless of what other valid intersection groups allow, then this rule must be in a different valid intersection group.

See Valid Intersection Examples.

Anchor and Nonanchor Dimensions

Anchor and nonanchor dimensions:

- Anchor dimensions are always required dimensions in the cube that is used in the valid intersection evaluation.
- Nonanchor dimensions are either required or not:
  - If a nonanchor dimension is required, any cube that doesn't use that dimension will ignore any valid intersection group where that dimension is tagged as required as it evaluates the valid intersections.
  - If a nonanchor dimension isn't required, any cube that doesn't use that dimension will still evaluate any valid intersection group that includes that dimension as not required and evaluate the intersections of any other dimensions in the valid intersection group in use in the cube.
- Unselected anchor dimension members are valid by default, but you can mark them invalid by clearing the **Unselected Members are Valid** option. This option marks all intersections with anchor dimension members not selected in this rule as invalid.

See Valid Intersection Examples.

Valid Intersection Examples

This section provides valid intersection group and valid intersection rule examples to illustrate a few simple, complex, and edge-case scenarios.

Example: Anchor and Nonanchor Dimensions

The choice of the anchor dimension is critical. Consider the following example, which produces a different result based on the anchor dimension definition:

- Valid intersection group 1 defines Entity as the anchor dimension and Product as a nonanchor dimension.
- Valid intersection group 2 reverses this definition with Product as the anchor dimension and Entity as the nonanchor dimension.

Table 13-1 Example - Anchor Dimension is Entity

Valid Intersection Group	Anchor Dimension - Entity	Nonanchor Dimension - Product
1	DESC(500 - Manufacturing) - Unselected members are valid	DESC(P_TP1 - Computer Equipment)

Group 1 means entities that are descendants of Manufacturing are valid only with descendant products of Computer Equipment. No other products are valid with descendants of Manufacturing. All other entities besides descendants of Manufacturing are valid with all products, including descendants of Computer Equipment.

Table 13-2 Example - Anchor Dimension is Product

Valid Intersection Group	Anchor Dimension - Product	Nonanchor Dimension - Entity
2	DESC(P_TP1 - Computer Equipment) - Unselected members are valid	DESC(500 - Manufacturing)

Group 2 means products that are descendants of Computer Equipment are only valid with descendant entities of Manufacturing. No other entities are valid with descendants of Computer Equipment. All other products besides descendants of Computer Equipment are valid with all entities, including descendants of Manufacturing.

> **Caution:**
>
> The choice of anchor dimension is significant. You'll get dramatically different results if you choose the wrong anchor dimension.

Example: Required Dimension

In the following example, if a nonanchor dimension isn't required, then the application evaluates all remaining dimension intersections in the valid intersection group for a cube that doesn't contain the nonrequired dimension. This behavior could result in the evaluation of a valid intersection group with only one effective dimension.

Table 13-3 Example - Required vs. Non-Required Nonanchor Dimensions

Valid Intersection Group	Anchor Dimension - Entity	Nonanchor Dimension - Product
1	DESC(500 - Manufacturing) - Unselected members are valid	DESC(P_TP1 - Computer Equipment) - Not required

In Group 1, the product dimension isn't required, and unselected entities are valid. Therefore, if the cube of the form or business rule, at runtime, doesn't include the product dimension, the application evaluates the entity dimension selections to mark all entities as valid for a cube that doesn't contain the product dimension.

Table 13-4 Example - Required vs. Non-Required Nonanchor Dimensions

Valid Intersection Group	Anchor Dimension - Entity	Nonanchor Dimension - Product
2	DESC(500 - Manufacturing) - Unselected members are invalid	DESC(P_TP1 - Computer Equipment) - Not required

In Group 2, the product dimension isn't required, and unselected entities are invalid. Therefore, if a cube doesn't include the product dimension, then the application evaluates the entity dimension selections to mark all entities except descendants of Manufacturing as invalid. Thereafter, any cube that doesn't use the product dimension will only allow data entry in the descendants of Manufacturing entities.

> **Caution:**
>
> Carefully consider whether a nonanchor dimension is required or not, especially if the result leaves a valid intersection group with only one effective dimension. Additionally, selecting the **Unselected Members are Valid** option for anchor dimension members also plays a significant role in the system behavior for valid intersections. See Valid Intersection Examples.

Example: Unselected Members are Valid

In the following example, two intersection groups are valid. In one group, the anchor dimension unselected members are invalid (this option is cleared). In the other group, the anchor dimension unselected members are valid (this option is selected).

Table 13-5 Example - Unselected Members are Valid

Valid Intersection Group	Anchor Dimension - Account	Nonanchor Dimension - Entity
1	IDESC(BS - Balance Sheet) - Unselected members are invalid	000 - No Department
2	IDESC(GP - Gross Profit) - Unselected members are valid	IDESC(403 - Sales)

Because Group 1 defines all unselected members as invalid, the application marks noninclusive descendants of Balance Sheet invalid. Gross Profit isn't an inclusive descendant of Balance Sheet. So even though Group 2 explicitly states inclusive descendants of Gross Profit are valid with inclusive descendants Sales entities, the invalid definition from Group 1 overrides any further valid intersections of the same anchor dimension member set.

Example: Redundant or Overlapping Valid Intersection Rules Within the Same Valid Intersection Group

When valid intersection rules are within the same valid intersection group and produce any redundancy or overlap, the system marks an intersection valid if either of the valid intersection rule conditions are met.

Table 13-6 Example - Redundant or Overlapping Valid Intersection Rules Within the Same Valid Intersection Group

Valid Intersection Rule	Anchor Dimension - Account	Nonanchor Dimension - Entity
1	IDESC(GP - Gross Profit) - Unselected members are valid	IDESC(403 - Sales)
2	IDESC(NI - Net Income) - Unselected members are valid	IDESC(TD - Total Department)

Because Gross Profit is a descendant of Net Income and Sales is a descendant of Total Department, inclusive descendants of Gross Profit are valid with any inclusive Descendant of Total Department. Rule 1 is a subset of Rule 2, so Rule 1 is effectively a "No operation" rule and is unnecessary. There is no restriction on inclusive descendants of Gross Profit accounts only being valid for inclusive descendants of Sales Entities.

Example: Redundant or Overlapping Valid Intersection Rules in Different Valid Intersection Groups

When valid intersection rules are in different valid intersection groups and produce any redundancy or overlap, the system marks an intersection valid only if it satisfies the requirements of all valid intersection groups.

In the following example, there are redundant or overlapping rules in different groups:

Table 13-7 Example - Redundant or Overlapping Valid Intersection Rules in Different Valid Intersection Groups

Valid Intersection Group	Anchor Dimension - Account	Nonanchor Dimension - Entity
1	IDESC(GP - Gross Profit) - Unselected members are valid	IDESC(403 - Sales) - Required
2	IDESC(NI - Net Income) - Unselected members are valid	IDESC(TD - Total Department) - Not required

Because Group 1 is further restrictive for inclusive descendants of Gross Profit accounts being valid with inclusive descendants of Sales entities, the application enforces this group for these intersections. Other, non-Gross Profit accounts can still use all inclusive descendants of Total Department entities, but inclusive descendants of Gross Profit accounts must use inclusive descendants of Sales entities.

Redundancy or Overlap in Valid Intersection Rules

Valid intersection rules within the same valid intersection group, which produce any apparent conflict or overlap, are marked valid if either valid intersection rule condition is met.

If different valid intersection groups share the same attributes, including the anchor dimension, required and not required nonanchor dimensions, and **Unselected Members are Valid** attribute, they will be treated as rules of the same valid intersection group.

Shared Members and Valid Intersection Rules

Shared members are supported in valid intersection rules. If a base member is selected for a valid intersection rule, any shared members are also included in the rule. Conversely, if a shared member is selected for a valid intersection rule, the base member is also included in the rule.

Substitution Variables and Valid Intersection Rules

You can use substitution variables in valid intersection rules. User variables are not supported. Substitution variables can be set on the Essbase server, application, or database level. The same substitution variable can exist on multiple levels; the application uses the first one it finds as it searches in this order:

1. Database (cube)
2. Application
3. Server

Evaluation Order

Evaluation order for valid intersection groups orders invalid results sets as quickly as possible, increasing the speed and efficiency of the overall valid intersection evaluation.

For example, the application evaluates the first valid intersection group in the list, then the second group, and so on. If the application finds an invalid intersection in the second group in the list, it will stop evaluating the rest of the list because, once an intersection is defined as invalid, it will override other valid intersection rule results.

To change the order in which the groups are evaluated, see Changing the Valid Intersection Group Evaluation Order.

Creating Valid Intersections

You can define rules to filter certain cell intersections to users when they enter data, select members, or select runtime prompts.

To create a valid intersection:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** tab is at the bottom of the page.

3. Create the valid intersection group:
   a. Click **Create**.
   b. Enter a name and description for the new intersection.
   c. The **Enabled** checkbox is selected by default. To disable the valid intersection group, clear the **Enabled** checkbox. You can also enable or disable a valid intersection group directly on the **Valid Intersections** page. See Disabling and Enabling Valid Intersection Groups.
   d. For **Definition Type**, select one of the following:
      - **Valid Intersection**
      - **Invalid Intersection**
   e. To select the anchor dimension for the intersection group, click  next to **Select Anchor Dimension**.
   f. **Optional:** By default, the anchor dimension members that are not specified in the valid intersection rule are marked valid. To clear this option, click  next to the anchor dimension, and then click **Unselected members are valid**.

      For an explanation of this option and for an example of how it's used, see Anchor and Nonanchor Dimensions.
   g. To select additional dimensions (called nonanchor dimensions), click **Add Dimension**.

h. **Optional:** By default, nonanchor dimensions are not required. To make a nonanchor dimension required, click  next to the nonanchor dimension, and click **Required**.

4. Define the valid intersection rule:
   a. Click **Add Rule**.
   b. To select the range of members to include, exclude, or remove in the valid intersection, click  next to the dimensions in the new rule:
      - Click **Edit** to open the **Select Members** page and select the members, substitution variables, and attributes to include in the valid intersection rule. See Using the Member Selector.
      - Click **Exclude** or **Exclude All** to define the dimension members you want to exclude from the rule:
        – **Exclude**: Selecting this option excludes members by ID. Only the specified members (base or shared) will be excluded.
        – **Exclude All**: Selecting this option excludes members by name. If a base member is specified, then the base and all of its shared members will be excluded. If a shared member is specified, then this member, its base member, and all other shared members of this member will be excluded.
      - Click **Clear** to clear the selection.

      To delete a rule, click  .
5. Click **Save and Close**.

The new valid intersection group is added to the end of the valid intersections list. To reorder the rules in the list, see Changing the Valid Intersection Group Evaluation Order.

Managing Valid Intersections

Related Topics

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

Viewing Valid Intersections

To view valid intersections:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** tab is at the bottom of the page.

3. To search the entire listing for certain intersections, enter search criteria in the **Search** box and then click  to display only the intersections that meet the search criteria.
4. To filter the listing so it displays only the intersections that meet the filter criteria, specify the following filter options:
   -  : Click to display the **Filter** page where you can select from a robust list of filter options. For a complete list of filter options and their descriptions, see Filtering Valid Intersections.
   -  : Click to clear all filters.
   - **Enabled**: Filters the list so you view only the intersections that are enabled (**Yes**), not enabled (**No**), or select **All**.
   - **Definition Type**: Filters the list so you view only the intersections that are the **Valid Intersection** or **Invalid Intersection** types, or select **All**.
5. You can also perform these tasks on the **Valid Intersections** page:
   - **Actions** menu: To import and export valid intersections, see Importing and Exporting Intersections .
   - **Create**: See Creating Valid Intersections.
   - **Refresh**: Click to refresh the intersections listing.
6. For each listing on the intersections page you can perform the following tasks:
   - **Enabled** column: Indicates whether an intersection is enabled. Click the check mark next to the intersection to disable or enable it. A green check mark indicates that the definition is enabled. See Disabling and Enabling Valid Intersection Groups.
   - **Action** column: Click ••• to edit, duplicate, or delete an intersection, or to move it up or down in the listing order. See the following topics:
     - Changing the Valid Intersection Group Evaluation Order
     - Editing Details for a Valid Intersection Group
     - Duplicating Valid Intersection Groups
     - Deleting a Valid Intersection Group

Filtering Valid Intersections

You can filter the list of intersections by certain criteria such as whether the intersection is enabled or not, whether the definition is valid or invalid, when it was modified, and by whom. When you filter, only the intersections that meet the filter criteria are displayed on the **Valid Intersections** page.

To filter intersections:

1. Click **Application**, and then click **Valid Intersections**.
2. Click  to specify filter criteria:

- **Enabled**: View only the intersections that are enabled (**Yes**), not enabled (**No**), or select **All**.
- **Definition Type**: View only the intersections that are the **Valid Intersection** or **Invalid Intersection** types, or select **All**.
- **Modified**: View only the intersections that were modified before or after a certain date and time, or the intersections modified within a range of dates or times. Select **After**, **Before**, or **Between** and then click  to specify the date and time criteria.
- **Modified By**: View only the intersections modified by selected users.

3. Click **More** to further refine the filter criteria:
   - **Dimensions**
   - **Anchor Dimensions**
   - **Unselected members are valid**: Choose **Yes**, **No**, or **All**.
   - **Additional Dimensions Required**: Choose **Yes**, **No**, or **All**.
4. Click **Apply**.

Importing and Exporting Intersections

About Importing and Exporting Valid Intersections

You can export the filtered list of valid intersections from the listing page to a location on your local computer, or you can export them to the server. If no filter is defined on the listing page, then all intersections are exported.

Use the Import action to import intersections into the application from a location on your local computer or import them from the server. When you perform an import, the system tests the import file for anything that might break a definition; for example, if a cube is missing, or if an anchor dimension doesn't exist, or if a subrule is not found, errors are logged in the error file. Import jobs will only complete successfully if the import file has no errors.

> **Note:**
>
> Subrules provide information about the content of the rules such as the member selection or exclusion for anchor and non anchor dimensions, and the restriction type applied for each subrule.

Depending on the export or import location you choose, the intersections are exported or imported in either an Excel file format (XLSX) or a zip file format.

> **Note:**
>
> Locked valid intersection rules are not exported when you export valid intersections. Locked valid intersection rules (and rules that start with restricted prefixes such as FCCS_, OCX_, OEP_, OFS_, OPF_, OWP_, TRCS_) are not imported when you import valid intersections.

Valid Intersections Import File

The Excel import file must have two sheets with the following names for the first and second sheets:

1. `Rules`
2. `Sub Rules`

The `Rules` sheet has the following column headings:

- **Name**
- **Position**
- **Description**
- **Enabled**
- **Definition Type**
- **Anchor Dim Name**
- **Anchor Dimension Apply to Selected Members**
- **Dim1**
- **Dim1 Required**
- **Dim2**
- **Dim2 Required**
- **Dim***X*
- **Dim***X* **Required**

The `Sub Rules` sheet has the following column headings:

- **Name** - This column must contain the name of the Rule from the first sheet
- **Anchor Members**
- **Anchor Exclusion**
- **Anchor Exclusion All**
- **Dim1 Members**
- **Dim1 Exclusion**
- **Dim1 Exclusion All**
- **Dim2 Members**
- **Dim2 Exclusion**
- **Dim2 Exclusion All**
- **Dim***X* **Members**
- **Dim***X* **Exclusion**
- **Dim***X* **Exclusion All**

Exporting and Importing Valid Intersections

To export and import intersections:

1. Click **Application**, and then click **Valid Intersections**.

2. Apply filters to the listing, as needed. See Filtering Valid Intersections.
3. To export, click **Actions**, then **Export**, and then select the target location for the export file:
   - **Local**: Saves the export file to a location on your local computer. If you choose this option, click **Export**, and then specify where to save the export file.
   - **Outbox**: Runs a job that saves the export file in a zip format to the server which you can download now or use to schedule an export job later. If you choose this option, click **Save and Run Job**.

     To download the export file from the Outbox:

     a. Click **Application**, and then click **Jobs**.

     b. Under **Recent Activity** click the export job.

     c. On the **Job Details** page, click  to select a download location for the export file. You can also download the export file from the Inbox/Outbox Explorer for your business process.
4. If editing the export file in Excel, note that the Excel file has two sheets: `Rules` and `Sub Rules`.
5. To import, click **Actions**, then **Import**, and then select the location of the import source file:
   - **Local**: Loads the import file from a location on your computer. For **Source File**, click **Browse** to select the import file on your computer, and then click **Import**.
   - **Inbox**: Runs a job that loads the import file from the server. The import file must be in a zip file format. Enter the name of the file in **Source File**, click **Save and Run Job**, and then specify the **Name** and **Error File** on the **Save as Job** dialog. The error file provides information about the intersections that were not imported. You can download the error file from the Inbox/Outbox Explorer for your business process.

Changing the Valid Intersection Group Evaluation Order

Evaluation order for intersection groups orders invalid results sets as quickly as possible, increasing the speed and efficiency of the overall intersection evaluation.

To learn more about evaluation order, see Evaluation Order.

To change the position of a valid intersection group in a list:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** tab is at the bottom of the page.

3. To the right of the intersection group in the listing, click •••.
4. Select **Move Up** or **Move Down**.

> **Tip:**
>
> You can also drag intersection groups to move them up and down in the list.

Disabling and Enabling Valid Intersection Groups

Intersection groups, by default, are enabled at the time of creation. If you don't want an intersection group to be evaluated or used, you can disable it on the **Valid Intersections** page. When an intersection group is disabled, the intersection rule for that group no longer applies when viewing application forms, business rules, or runtime prompts. You can also reenable a disabled intersection group.

To disable and enable an intersection group:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** tab is at the bottom of the page.

3. In the Enabled column of the intersection list, click the check mark next to the intersection group that you're disabling or enabling.

> **Note:**
>
> The check mark is green if the group is enabled.

4. Ensure that any remaining groups that are enabled are still listed in the correct evaluation order in the intersections list. If they are not, then move them up or down in the order.

   See Changing the Valid Intersection Group Evaluation Order.

Editing Details for a Valid Intersection Group

To edit intersection group details, you work with dimension members in the member selector. You can also define exclusions in intersection rules.

To edit intersection group details:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** tab is at the bottom of the page.

3. Click the name of the intersection group you want to edit.

> **Note:**
>
> A pagination bar displays at the bottom of the rules page. If you have a large number of rules, you can use the pagination bar to easily navigate between the rules pages and to jump to a specific page. The page size options (number of rules per page), which you can select on the pagination bar, are **25**, **50**, **100**, and **All**. (Note that the **All** option isn't available if you have more than 300 rules.)

- To edit dimension details, next to the dimension, click  to select the members to include, exclude, or remove in the intersection rule:
  - Click **Edit** to open the **Select Members** page and select members, substitution variables, and attributes to include in the intersection rule. You can also type in the members or functions.

    See Using the Member Selector.
  - Click **Exclude** or **Exclude All** to define the dimension members you want to exclude from the rule:
    - **Exclude**: Selecting this option excludes members by ID. Only the specified members (base or shared) will be excluded.
    - **Exclude All**: Selecting this option excludes members by name. If a base member is specified, then the base and all of its shared members will be excluded. If a shared member is specified, then this member, its base member, and all other shared members of this member will be excluded.
  - Click **Clear** to clear the selection.
- To delete a dimension from an intersection group, next to the dimension, click  , and then click  .
- To remove a rule from an intersection group, click  .
- To add a dimension or a rule to an intersection group, click **Add Rule** or **Add Dimension**.

4. Click **Save and Close**.

Duplicating Valid Intersection Groups

To speed intersection group creation, you can duplicate an existing intersection group and then edit it.

To duplicate an intersection group:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.
3. Click ••• to the right of the intersection group you want to duplicate, and then select **Duplicate**.

   The duplicate group is added to the end of the intersections list with the word "Copy" appended to the name.
4. Open the intersection group and edit it.

5. Reorder the intersection groups, if needed. See Changing the Valid Intersection Group Evaluation Order.

Deleting a Valid Intersection Group

After a group is deleted, the intersection groups are reordered. If there are three intersection groups, and the second one in the order is deleted, the third intersection group becomes number two.

To delete an intersection group:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click the **Setup** tab.

> **Note:**
>
> If Redwood Experience is enabled, the **Setup** tab is at the bottom of the page.

3. Click ••• to the right of the intersection group that you want to remove, and then select **Delete**.
4. Reorder the remaining intersections, if needed. See Changing the Valid Intersection Group Evaluation Order.

To delete an intersection rule from an intersection group, see Editing Details for a Valid Intersection Group.

Suppressing Invalid Data in Forms

Suppressing invalid data hides rows or columns in application forms that contain invalid data. If this option isn't selected, then the application displays rows or columns that contain cells with data that are invalid. Cells with invalid data are read-only.

To suppress invalid data in forms:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Forms**.
2. Select the form, click  , and then click **Layout**.
3. Under **Grid Properties**, select **Suppress invalid data - Rows** and/or **Suppress invalid data - Columns**.
4. Click **Save**.

Clearing Invalid Data

Related Topics

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

About Invalid Data

If data already exists at intersections, then adding or modifying valid intersections invalidates the existing data. Creating a valid intersection rule or modifying an existing valid intersection rule doesn't clear data in the invalid intersections. You must generate a valid intersection rule report, which will show where data exists at invalid intersections, and then determine whether to clear the data.

> **Note:**
>
> Data may remain at an invalid intersection for historical purposes or for use in forward looking scenarios. Therefore, it's not a requirement to clear data at invalid intersections.

Working With Invalid Intersection Reports

You can view reports that show data at invalid intersections on the **Reports** tab of the **Valid Intersections** page. **Invalid Intersection Reports** lists existing reports, their status, and the last time they were run.

To work with invalid intersection reports:

1. Click **Application**, then **Valid Intersections**, and then click the **Reports** tab.

   > **Note:**
   >
   > If Redwood Experience is enabled, the **Reports** tab is at the bottom of the page.

2. Perform a task:
   - To refresh the listing, click **Refresh**.
   - To create a report, click **Create**, name the report, select the cube, choose the scope of the report, and then select when to run the report. You can run the report now or run the report later by saving it as a job. To save a report without running it, click **Save and Close**. To remove a report after it's run, click **Remove Reports**.

     > **Note:**
     >
     > Invalid intersection reports are not supported for aggregate storage cubes. Aggregate storage cubes are not listed in the **Cube** drop-down list.

   - To edit a report, click  , and then **Edit**.
   - To copy a report, click  , and then **Duplicate**.
   - To run a report, click  , and then **Run**.

- To delete a report, click  , and then **Delete**.

Clearing Data at Invalid Intersections

Users with appropriate permissions can clear the invalid data if the data isn't needed. To clear invalid data, run the report, and then click **Clear Invalid Intersections**.

Working with Valid Intersections

Related Topics

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

Working with Valid Intersections in Application Forms

Using valid intersections prevents data entry for invalid intersections as defined in the applicable valid intersection group. The affected cells in the form display as read-only following standard, read-only color coding. If you hover the cursor over an invalid intersection, a tool tip displays indicating the cell is read-only because it's defined as an invalid intersection.

The valid intersection group applies first to the form point of view and page axis. If the point of view intersections are all invalid, then a warning message is displayed, and the form doesn't render a data grid until a valid intersection is selected.

If the point of view has valid intersections, then the rows and columns are filtered to restrict data entry at invalid intersections. If the **Suppress Invalid Data** option for the form is enabled, then the form suppresses invalid rows, columns, or both, as appropriate.

Any rows or columns, which consist of a mix of valid and invalid intersections, display those intersections as valid or invalid, as appropriate. Invalid intersections are displayed with standard, read-only shading and preclude data entry.

Application users can reset the point of view to the default, unfiltered list without closing and reopening the form by clearing the selections. Application users can also clear a selection, thus opening up more selections for other dimensions. They will not be able to render a form with a dimension cleared, because valid members must be selected for each dimension.

In the member selector, invalid members are suppressed due to valid intersection rules. Application users can display invalid members in the member selector using the **Show Invalid Members** option. Invalid members are displayed but are unavailable for selection.

> **Note:**
>
> Valid intersection groups don't grant access to dimension members. Valid intersection groups further restrict the valid intersections of dimension members already granted to an application user.

Table 13-8 Form Behavior if Valid Intersections are Applied

Action	Behavior
Open a form	The form renders with member selections as defined in the form definition, adhering to the user's access rights for dimensions, and applies valid intersection groups with the most recently used as current selections.
Select members from a point of view dimension	The application: • Enables users to select a member on the point of view • In the member selector for a point of view dimension, enables users to select from a filtered list of remaining valid intersections, which is based on the members that were selected for the other point of view dimensions • Ignores the order in which point of view dimension members are selected because selecting a member from any dimension included in a valid intersection group dynamically filters the remaining dimension members lists for those dimensions included in the valid intersection group, as appropriate, when that dimension is selected • Provides the option to hide invalid members from dimension lists or display them as unselectable in the point of view • Provides the ability to reset the point of view to the fully unfiltered list without closing and reopening the form by clearing the selections Note that ad hoc forms, both in Web and Oracle Smart View for Office, will not filter page or point of view members according to valid intersection groups.
Select **Go** to render a form based on point of view selections. You can also click the right arrow in the form point of view.	The form renders as defined based on the valid point of view intersections.
Enter and save data	The form data is entered and saved.

Working with Valid Intersections in Calculation Manager Rule Runtime Prompts

Valid intersection groups apply to runtime prompts when launched from within the context of the application. Runtime prompts will prevent users from selecting invalid intersections as defined in the valid intersection groups.

Filtering according to valid intersection groups isn't supported in Oracle Smart View for Office forms. The rule, however, will not launch if you choose an invalid intersection in the runtime prompts both in the Web and in Smart View.

Table 13-9 Runtime Prompt Behavior if Valid Intersections are Applied

Action	Behavior
Launch a Calculation Manager rule runtime prompt	The application: • Prevents the user from selecting invalid intersections within the runtime prompt based on the valid intersection group • Prevents the Calculation Manager rule from executing if there are invalid intersections in the runtime prompts
Enter valid intersections	The valid intersection is allowed to be entered.

Defining Cell-Level Security

Define security rules which restrict users and groups from viewing data in certain cell intersections in forms.

Related Topics

- Understanding Cell-Level Security
- Creating Cell-Level Security Definitions
- Viewing Cell-Level Security Definitions
- Filtering Cell-Level Security Definitions
- Importing and Exporting Cell-Level Security Definitions
- Testing Cell-Level Security
- Reordering the Cell-Level Security Definitions List

Understanding Cell-Level Security

About Cell-Level Security

Service Administrators applying cell-level security can deny access to cells that a user would normally have access to due to their regular security. Cell-level security is therefore defined as an exception to the existing member security. For example, a Department Manager requires access to all accounts in their own department, but only a certain account in all other departments. With the usual metadata security the Manager would have access to all accounts across all departments, but using cell-level security enables the Service Administrator to control the intersection of all accounts with the Manager's department and only the specific account in all other departments.

Cell-level security uses rules, similar to valid intersection rules, to deny read or write access to users viewing certain cell intersections anywhere a cell is shown (for example, forms, runtime prompts, Smart View, reports, dashboards, infolets, and so on). When cell-level security rules are applied, users with read access can see the data value in a cell but the cell is not editable. If users are denied read access to a cell, the value displayed in the cell is #noaccess.

If you are a Service Administrator, you can define and assign cell-level security rules to any user or group. Cell-level security doesn't affect you.

Anchor and Nonanchor Dimensions

Cell-level security definitions use anchor and nonanchor dimensions:

- Anchor dimensions are always required dimensions in the cube that is used in the cell-level security definition.
- Nonanchor dimensions are either required or not:
  - If a nonanchor dimension is required, any cube that doesn't use that dimension will ignore any cell-level security definitions where that dimension is tagged as required.
  - If a nonanchor dimension isn't required, any cube that doesn't use that dimension will still evaluate any cell-level security definition that includes that dimension as not required and evaluate the definitions of any other dimensions in the definition in use in the cube.

  - By default, nonanchor dimensions aren't required. To make a nonanchor dimension required, click  next to the nonanchor dimension, and click **Required**.
- By default, the anchor dimension members that are not specified in the rule are included in the security definition, but you can clear this option by clicking  next to the anchor dimension, and then clicking **Apply to Selected Members Only**.

Creating Cell-Level Security Definitions

To create a cell-level security definition:

1. Click **Application**, and then click **Cell-Level Security**.
2. Create the definition:
   a. Click **Create**.
   b. Enter a name and description for the definition.
   c. The **Enabled** checkbox is selected by default. To disable the definition, clear the **Enabled** checkbox. You can also enable or disable a definition directly on the **Cell-Level Security Definitions** page.
   d. To define cube-specific security, click **Cubes** and select from the list of cubes or select **All**.
   e. An anchor dimension is required. To select the anchor dimension, click **Select Anchor Dimension**. For information about anchor and nonanchor dimensions, see Understanding Cell-Level Security
   f. **Optional:** By default, the anchor dimension members that are not specified in the rule are included in the security definition. To clear this option, click  next to the anchor dimension, and then click **Apply to Selected Members Only**.
   g. To select additional dimensions (called nonanchor dimensions), click **Add Dimension**.
   h. **Optional:** By default, nonanchor dimensions are not required. To make a nonanchor dimension required, click  next to the nonanchor dimension, and click **Required**.
3. Define the cell-level security rule:
   a. Click **Add Rule**.
   b. In the **Users, Groups** column, click  to find the users and groups to include in the cell-level security rule.
   c. For **Restriction**, choose **Deny Read** (default) or **Deny Write**. **Deny Read** is the default option because it is the most restrictive. If users are denied read access to a cell, the value displayed in the cell is #noaccess. Users with **Deny Write** access can see the data value in a cell but the cell is not editable.
   d. Click  next to the dimensions in the new rule:
      - Click **Edit** to open the **Select Members** page and select the members, substitution variables, and attributes to include in the cell-level security rule.
      - Click **Exclude** or **Exclude All** to define the dimension members you want to exclude from the rule:
        - **Exclude**: Selecting this option excludes members by ID. Only the specified members (base or shared) will be excluded.

  - **Exclude All**: Selecting this option excludes members by name. If a base member is specified, then the base and all of its shared members will be excluded. If a shared member is specified, then this member, its base member, and all other shared members of this member will be excluded.

- Click **Clear** to clear the selection.

To delete a rule, click  .

4. Click **Save**.

The new cell-level security definition is added to the end of the list. Definitions are evaluated in the order they appear in the list. To reorder the definitions list, see Reordering the Cell-Level Security Definitions List.

After a definition is created, you can test it in a form to see how the form will look from a user's perspective. See Testing Cell-Level Security.

Viewing Cell-Level Security Definitions

To view cell-level security definitions:

1. Click **Application**, and then click **Cell-Level Security**.
2. To search the entire listing for certain definitions, enter search criteria in the **Search** box and then click  to display only the definitions that meet the search criteria.
3. To filter the listing so it displays only the definitions that meet the filter criteria, specify the following filter options:
   -  : Click to display the **Filter** page where you can select from a robust list of filter options. For a complete list of filter options and their descriptions, see Filtering Cell-Level Security Definitions.
   -  : Click to clear all filters for **Effective Assignment**, **Enabled**, and **Restriction**.
   - **Effective Assignment**: Filters the list so you view only the definitions that effect certain users or groups.
   - **Enabled**: Filters the list so you view only the definitions that are enabled (**Yes**), not enabled (**No**), or select **All**.
   - **Restriction**: Filters the list so you view only the definitions that are assigned the **Deny Read** or **Deny Write** restriction, or select **All**.
4. You can also perform these tasks on the **Cell-Level Security Definitions** page:
   - **Actions** menu: To import and export definitions, see Importing and Exporting Cell-Level Security Definitions.
   - **Test**: See Testing Cell-Level Security.
   - **Create**: See Creating Cell-Level Security Definitions.
   - **Refresh**: Click to refresh the definitions listing.
5. For each listing on the definitions page you can perform the following tasks:
   - **Enabled** column: Indicates whether a definition is enabled. Click the check mark next to the definition to disable or enable it. A green check mark indicates that the definition is enabled.

- **Action** column: Click ••• to edit, duplicate, or delete a definition, or to move it up or down in the listing order.

Filtering Cell-Level Security Definitions

You can filter the list of cell-level security definitions by certain criteria such as by cube, by restriction, or by date. When you filter, only the definitions that meet the filter criteria are displayed on the **Cell-Level Security Definitions** page.

To filter cell-level security definitions:

1. Click **Application**, and then click **Cell-Level Security**.
2. Click  to specify filter criteria:
   - **Cubes**: View only the definitions in the selected cube or cubes, or select **All**.
   - **Enabled**: View only the definitions that are enabled (**Yes**), not enabled (**No**), or select **All**.
   - **Restriction**: View only the definitions that are assigned the **Deny Read** or **Deny Write** restriction, or select **All**.
   - **Modified**: View only the definitions that were modified before or after a certain date and time, or the definitions modified within a range of dates or times. Select **After**, **Before**, or **Between** and then click  to specify the date and time criteria.
   - **Modified By**: View only the definitions modified by selected users.
   - **Effective Assignment**: View only the definitions that effect the selected users or groups.
3. Click **More** to further refine the filter criteria:
   - **Dimensions**
   - **Anchor Dimensions**
   - **Anchor Dimension: Apply to Selected Members Only**: Choose **Yes**, **No**, or **All**.
   - **Additional Dimensions Required**: Choose **Yes**, **No**, or **All**.
4. Click **Apply**.

Importing and Exporting Cell-Level Security Definitions

About Importing and Exporting Cell-Level Security Definitions

You can export the filtered list of cell-level security definitions from the listing page to a location on your local computer, or you can export them to the server. If no filter is defined on the listing page, then all cell-level security definitions are exported.

Use the Import action to import cell-level security definitions into the application from a location on your local computer or import them from the server. When you perform an import, the system tests the import file for anything that might break a definition; for example, if a cube is missing, or if an anchor dimension doesn't exist, or if a subrule is not found, errors are logged in the error file. Import jobs will only complete successfully if the import file has no errors.

> **Note:**
>
> Subrules provide information about the content of the rules such as the user assignment for each member of the rule (subrule), the member selection or exclusion for anchor and non anchor dimensions, and the restriction type applied for each subrule.

Depending on the export or import location you choose, the application definitions are exported or imported in either an Excel file format (XLSX) or a zip file format.

Cell-Level Security Import File

The Excel import file must have two sheets with the following names for the first and second sheets:

1. `Rules`
2. `Sub Rules`

The `Rules` sheet has the following column headings:

- **Name**
- **Position**
- **Description**
- **Enabled**
- **Valid Cubes** - This column can contain either `All` or a list of comma-separate names of cubes, such as `Plan1, Plan2`
- **Anchor Dim Name**
- **Anchor Dimension Apply to Unselected Members**
- **Dim1**
- **Dim1 Required**
- **Dim2**
- **Dim2 Required**
- **Dim***X*
- **Dim***X* **Required**

The `Sub Rules` sheet must have the following column headings:

- **Name** - This column must contain the name of the Rule from the first sheet
- **Users**
- **User Groups**
- **Restriction** This column can contain `Deny Read` or `Deny Write`
- **Anchor Members**
- **Anchor Exclusion**
- **Anchor Exclusion All**
- **Dim1 Members**

- **Dim1 Exclusion**
- **Dim1 Exclusion All**
- **Dim2 Members**
- **Dim2 Exclusion**
- **Dim*X* Members**
- **Dim*X* Exclusion**
- **Dim*X* Exclusion All**

Exporting and Importing Cell-Level Security Definitions

To export and import cell-level security definitions:

1. Click **Application**, and then click **Cell-Level Security**.
2. Apply filters to the listing, as needed. See Filtering Cell-Level Security Definitions.
3. To export, click **Actions**, then **Export**, and then select the target location for the export file:
   - **Local**: Saves the export file to a location on your local computer. If you choose this option, click **Export**, and then specify where to save the XLSX export file.
   - **Outbox**: Runs a job that saves the export file in a zip format to the server which you can download now or use to schedule an export job later. If you choose this option, click **Save and Run Job**.

     To download the export file from the Outbox:

     a. Click **Application**, and then click **Jobs**.

     b. Under **Recent Activity** click the **Export CLS** job.

     c. On the **Job Details** page, click  to select a download location for the cell-level security export file. You can also download the export file from the Inbox/Outbox Explorer for your business process.
4. If editing the export file in Excel, note that the Excel file has two sheets: `Rules` and `Sub Rules`.
5. To import, click **Actions**, then **Import**, and then select the location of the import source file:
   - **Local**: Loads the import file from a location on your computer. For **Source File**, click **Browse** to select the import file on your computer, and then click **Import**.
   - **Inbox**: Runs a job that loads the import file from the server. The import file must be in a zip file format. Enter the name of the file in **Source File**, click **Save and Run Job**, and then specify the **Name** and **Error File** on the **Save as Job** dialog. The error file provides information about the definitions that were not imported. You can download the error file from the Inbox/Outbox Explorer for your business process.

Testing Cell-Level Security

After defining cell-level security and enabling it, you can test it to see what an effected user would see when they view a form. Testing ensures that users are seeing only the cell values they are allowed to see and nobody is seeing more than they should. To test a definition, you must enable it.

To test cell-level security:

1. Click **Application**, and then click **Cell-Level Security**.

2. Enable the definition you want to test.
3. Click **Test**, and then select the form that you want to test.
4. In the **Select or Enter a User Name** text box, specify a user name or click  to select a user, and then click OK.

> **Note:**
>
> You can test only one user at a time.

The form displays the user access to each cell as Read, Write, or None.

Reordering the Cell-Level Security Definitions List

Cell-level security definitions are evaluated in the order they appear on the definitions list; for example, the application evaluates the first cell-level security definition in the list, then the second definition, and so on.

To change the position of a cell-level security definition in the list:

1. Click **Application**, and then click **Cell-Level Security**.
2. To the right of the cell-level security definition, click •••.
3. Select **Move Up** or **Move Down**.

> **Tip:**
>
> You can also drag cell-level security definitions to move them up and down in the list.

14

Defining Data Maps and Creating File-Based Integrations

Define data maps to move data, comments, attachments, and supporting detail from source cubes and Smart Lists to target reporting cubes to consolidate data.

Data Integration is the mechanism by which integration processes are performed in Oracle Enterprise Performance Management Cloud. You can define file-based and direct integration sources, create mapping rules to translate source data in the required target format, and execute and manage the periodic data loading process. To learn how to create a file-based integration see Creating File-Based Integrations in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

To learn how to define data maps, see Defining Data Maps in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

15

Managing Jobs

Schedule jobs to perform common administrative tasks and lighten your workload. You can start jobs right away or schedule jobs to run later at intervals.

Related Topics

- How Jobs Save You Time
- Viewing the Status of Jobs
  View pending jobs and recent activity on the Jobs console.
- Scheduling Jobs
- Editing and Canceling Jobs
- Duplicating Jobs
- Canceling Rules Jobs and Ruleset Jobs
- Downloading Export Files From Your Outbox

How Jobs Save You Time

Jobs are actions such as exporting data or refreshing the database, which you can start right away or schedule to run later at intervals. To lighten your workload, define jobs to perform common administrative tasks such as:

- Importing and exporting metadata and data
- Refreshing the database
- Mapping cubes

Viewing the Status of Jobs

View pending jobs and recent activity on the Jobs console.

The Jobs console lists jobs that are in a pending state under **Pending Jobs**. Jobs that are processing, have run and are completed, or have errors are listed under **Recent Activity**.

Jobs are retained in the Jobs console for 90 days.

To view the status of jobs:

1. Click **Application**, and then click **Jobs**.
2. Perform a task:
   - To filter jobs and activity by criteria such as date or job type, click  .
   - To search for a job by name, enter text in the **Search**, and then click  .
   - To view a job's details, click the job.

Note:

- The application is automatically refreshed during an application upgrade. If any refresh errors occur during an application upgrade, you can view the errors on the Jobs page. These errors are captured in the job called **Content Update**.
- The **Recent Activity** list on the **Jobs** page defaults to display only the last day's data. Similarly, the job filter's **Start Time** and **End Time** dates default to yesterday and today to have only a one day difference. Click  to change the default display.
- Jobs older than 90 days are purged.

Scheduling Jobs

To schedule jobs:

1. Click **Application**, then **Jobs**, and then click **Schedule Jobs**.
2. Specify the following:
   - The type of job you're creating. For a list of jobs and descriptions, see Job Types.
   - When to run the job. You can schedule a job to run now or to run later. If scheduling the job to run later, see Scheduling Jobs to Run Later.
3. Click **Next**.
4. Select a job from the job list. Depending on the job type, there may be additional options and considerations. See Job Types.

Note:

- You can run up to five import or export jobs at one time.
- For export jobs, you can specify a unique output file name for each job. From the job listing page, click  next to the selected job and specify the output file name with a zip extension. The unique file name you specify will override the default export file name when the job runs.
- To prevent automatic backup from failing due to certain jobs running during daily maintenance, EPM Cloud disallows certain jobs from starting while the daily maintenance process is running. If the system prevents a job from starting, the reason will be stated in the Job Details. If you've enabled email notifications for the Job Console, you'll receive an email notification when a job does not start. If you have a job that is scheduled to start during the daily maintenance process, it is recommended that you reschedule your job to start outside of the daily maintenance window. See Setting the Daily Maintenance Process Start Time.

5. Click **Next**.
6. Review your choices. If satisfactory, click **Finish**.

To edit or cancel a job after scheduling it, see Editing and Canceling Jobs.

Job Types

Table 15-1 Job Types

Job Type	Description
Rules	Runs a rule that you select. Note the following: • You can filter the rules list by cube and by rule type. • You must click the check mark next to the rule you want to run before you can proceed. • For rules jobs with runtime prompts, clicking the check mark next to a rule will display the runtime prompt parameters. Set the runtime prompt values with which to run the rule in the job scheduler, and then click **OK**. • Hidden runtime prompts will pick up the default values that were set at design time in Calculation Manager. See About Rules.
Import Data*	Performs a data import that was saved as a job. See Importing Data.
Import Metadata*	Performs a metadata import that was saved as a job. See Importing Metadata.
Import Cell-Level Security Definition	Imports a cell-level security definition that was saved as a job. See Importing and Exporting Cell-Level Security Definitions.
Import Valid Intersections	Imports valid intersections that were saved as a job. See Importing and Exporting Intersections.
Export Data*	Performs a data export that was saved as a job. You can specify a unique output file name for each **Export Data** job. From the job list, click  next to the selected job and specify the **Output File Name** using a zip extension. You can verify the export file name on the **Review** page. The unique file name will override the default export file name when the job runs. See Exporting Data. **Note:** The driver/column dimension in the Export Data job definition needs to be dense.
Export Metadata*	Performs a metadata export that was saved as a job. You can specify a unique output file name for each **Export Metadata** job. From the job list, click  next to the selected job and specify the **Output File Name** using a zip extension. You can verify the export file name on the **Review** page. The unique file name will override the default export file name when the job runs. See Exporting Metadata.

Table 15-1 (Cont.) Job Types

Job Type	Description
Export Cell-Level Security Definitions	Exports a cell-level security definition that was saved as a job. See Importing and Exporting Cell-Level Security Definitions.
Export Valid Intersections	Exports valid intersections that were saved as a job. See Importing and Exporting Intersections.
Refresh Database*	Refreshes the application database. See Refreshing Application Databases.
Data Map	Performs a data mapping operation. See Defining Data Maps in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.
Invalid Intersection Reports	Runs a report that shows where data exists at invalid intersections. See Working With Invalid Intersection Reports.
Clear Cube*	Performs a cube clearing operation. See Creating Clear Cube Jobs.
Restructure Cube*	Performs a full restructure of a block storage cube to eliminate or reduce fragmentation. This will also remove empty blocks. Running this job won't push any changes from the business process to Essbase. See Improving Cube Performance.
Compact Outline*	Compacts the outline file of an aggregate storage cube. Compaction helps keep the outline file at an optimal size. Compacting the outline doesn't clear the data. Running this job won't push any changes from the business process to Essbase. See Improving Cube Performance.
Merge Data Slices*	Merges incremental data slices of an aggregate storage cube. Fewer slices improve a cube's performance. You can merge all incremental data slices into the main database slice or merge all incremental data slices into a single data slice without changing the main database slice. You can optionally remove cells that have a value of zero. See Improving Cube Performance.
Optimize Aggregation*	Generates optimized views based on collected query tracking information in an aggregate storage cube. For additional option descriptions, see Improving Cube Performance.
Administration Mode	Changes the login level for a business process. If you select **Administrators**, all non-administrative users will be logged off from the application after job execution. To restore access to an application for all users, select **All users**. See What Application and System Settings Can I Specify?
Execute Bursting Definition	Runs a report bursting job If you're using the next-generation Reports reporting solution, the bursting feature enables you to run a single report or book for more than one member of a single dimension for one data source, and publish a PDF output for each member. See "Working with Bursting" in *Designing with Reports for Oracle Enterprise Performance Management Cloud*.

Table 15-1 (Cont.) Job Types

Job Type	Description
Integration Pipeline*	Runs a Pipeline definition. This job supports running a Pipeline based on the parameters and variables that were defined for the Pipeline in the Data Integration user interface; for example, **Start Period** and **Import Mode**. For descriptions of the job parameters and variables, see the following topics: • Using an Integration Job Type in *Administering Data Integration for Oracle Enterprise Performance Management Cloud* • Running a Pipeline in *REST API for Enterprise Performance Management Cloud* For information about creating a Pipeline definition, see Using the Pipeline in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

*This job is prevented from starting during daily maintenance, whether it's a scheduled job or a job that is started on an ad hoc (unscheduled) basis. Oracle recommends that you start this job outside of the daily maintenance window. See Setting the Daily Maintenance Process Start Time.

Scheduling Jobs to Run Later

Table 15-2 Scheduling Jobs Options

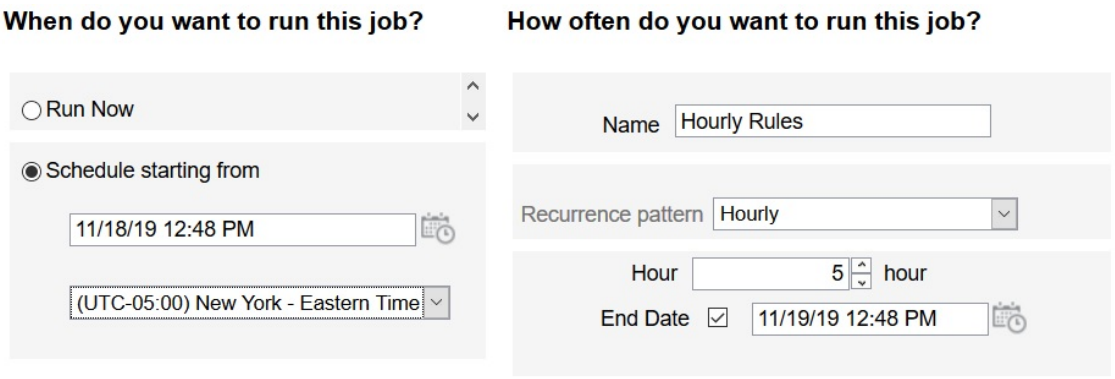
Option	Description
Schedule starting from	Select the starting date and time, including the time zone.
Name	Specify a name for the scheduled job; for example, **MyDailyCubeRefresh**. The name you specify is displayed along with the job name (which you'll choose on the next screen); for example, **MyDailyCubeRefresh : Refresh Database**.

Table 15-2 (Cont.) Scheduling Jobs Options

Option	Description
Recurrence Pattern	Specify the frequency with which to run the job: • **Hourly**: Hourly jobs run according to a timetable which is based on the values you set in the **Schedule starting from** and **Hour** fields. The schedule for **Hourly** jobs restarts each day during the midnight hour and the recurring job starts at the first hour, second hour, third hour (and so on up to 12 hours) after the midnight hour, depending on the value you select in the **Hour** field. So for example, if you specify a value of **5** in the **Hour** field, the possible times the job could start are during the 12:00 AM hour, the 5:00 AM hour, the 10:00 AM hour, the 3:00 PM hour, and the 8:00 PM hour each day. If the scheduled start time for a job is 12:48 PM and you set the **Hourly** recurrence for **5 Hours**, the job will start at 3:48 PM, which is the first scheduled time available after 12:48 PM in which to start a 5-hour recurring job. Then the job will run again at 8:48 PM, 12:48 AM, 5:48 AM, and 10:48 AM. For the default recurrence of **1 Hour**, the job will start running at the start time you specify and continue to run each hour until the ending date and time; for example, if the scheduled start time for a job is 12:48, the job will run at 12:48, 1:48, 2:48, 3:48 and so on. For a schedule of times based on the values selected in the **Hour** field, see Scheduling Hourly Jobs. • **By Minute**: Set the **Frequency** for 15 or 30 minutes. The job will start running in 15 or 30 minutes and continue to run at the selected frequency until the ending date and time; for example, if you set the job to run every 15 minutes and the starting time is 3:15, the job will start to run at 3:30, then 3:45, and so on. Jobs cannot be scheduled to run in increments smaller than 15 minutes. • **Run Once**: The job will run once at the starting date and time. • **Yearly**: The job will run at the starting date and time and continue to run each year thereafter until the end date. • **Monthly**: The job will run at the starting date and time and continue to run each month thereafter until the end date. • **Weekly**: The job will run at the starting date and time and continue to run each week until the end date. • **Daily**: The job will run at the starting date and time and continue to run each day until the end date.
End Date	If the job is recurring, select an ending date and time.

Scheduling Hourly Jobs

Hourly jobs run according to a timetable which is based on the values you set in the **Schedule starting from** and **Hour** fields. The schedule for **Hourly** jobs restarts each day during the midnight hour and the recurring job starts at the first hour, second hour, third hour (and so on up to 12 hours) after the midnight hour, depending on the value you select in the **Hour** field.



For example, if you specify a value of **5** in the **Hour** field, the hours during which the job will run each day are 12 AM, 5 AM, 10 AM, 3 PM, and 8 PM. The start time indicates the hour and minute when the system should start honoring the schedule, and the end time indicates when the system should stop. So if the start time is 12:48 PM on 11/18, then that means 12:48 AM, 5:48 AM, and 10:48 AM have already occurred on the current day (11/18) and the next available timeslot is 3:48 PM, which is when the scheduled job will run for the first time. After that it will run at 8:48 PM on 11/18. However on 11/19 it starts over again and will run at 12:48 AM, 5:48 AM, 10:48 AM, and so on until the system reaches the specified end time (in this case, 12:48 PM on 11/19).

> **Note:**
>
> When a job spans two days, the job runs during the midnight hour the next day and re-sets the recurrence pattern.

To schedule hourly jobs, see Scheduling Jobs.

Table 15-3 Timetable for Hourly Jobs

Hourly Recurrence	Timetable
1 (default)	AM: 12:00, 1:00, 2:00, 3:00, 4:00, 5:00, 6:00, 7:00, 8:00, 9:00, 10:00, 11:00 PM: 12:00, 1:00, 2:00, 3:00, 4:00, 5:00, 6:00, 7:00, 8:00, 9:00, 10:00, 11:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 1 hour, the job will run for the first time at 12:48 PM, and then again at 1:48 PM, 2:48 PM, 3:48 PM, and so on.
2	AM: 12:00, 2:00, 4:00, 6:00, 8:00, 10:00 PM: 12:00, 2:00, 4:00, 6:00, 8:00, 10:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 2 hours, the job will run for the first time at 12:48 PM, and then again at 2:48 PM, 4:48 PM, 6:48 PM, and so on.
3	AM: 12:00, 3:00, 6:00, 9:00 PM: 12:00, 3:00, 6:00, 9:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 3 hours, the job will run for the first time at 12:48 PM, and then again at 3:48 PM, 6:48 PM, 9:48 PM, 12:48 AM, 3:48 AM, and so on.

Table 15-3 (Cont.) Timetable for Hourly Jobs

Hourly Recurrence	Timetable
4	AM: 12:00, 4:00, 8:00 PM: 12:00, 4:00, 8:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 4 hours, the job will run for the first time at 12:48 PM, and then again at 4:48 PM, 8:48 PM, 12:48 AM, 4:48 AM, and so on.
5	AM: 12:00, 5:00, 10:00 PM: 3:00, 8:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 5 hours, the job will run for the first time at 3:48 PM, and then again at 8:48 PM, 12:48 AM, 5:48 AM, 10:48 AM, and so on.
6	AM: 12:00, 6:00 PM: 12:00, 6:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 6 hours, the job will run for the first time at 12:48 PM, and then again at 6:48 PM, 12:48 AM, 6:48 AM, and so on.
7	AM: 12:00, 7:00 PM: 2:00, 9:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 7 hours, the job will run for the first time at 2:48 PM, and then again at 9:48 PM, 12:48 AM, 7:48 AM, and so on.
8	AM: 12:00, 8:00 PM: 4:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 8 hours, the job will run for the first time at 4:48 PM, and then again at 12:48 AM, 8:48 AM, and so on.
9	AM: 12:00, 9:00 PM: 6:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 9 hours, the job will run for the first time at 6:48 PM, and then again at 12:48 AM, 9:48 AM, 6:48 PM, and so on.
10	AM: 12:00, 10:00 PM: 8:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 10 hours, the job will run for the first time at 8:48 PM, and then again at 12:48 AM, 10:48 AM, 8:48 PM, and so on.
11	AM: 12:00, 11:00 PM: 10:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 11 hours, the job will run for the first time at 10:48 PM, and then again at 12:48 AM, 11:48 AM, 10:48 PM, and so on.
12	AM: 12:00 PM: 12:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 12 hours, the job will run for the first time at 12:48 PM, and then again at 12:48 AM, and so on.

Editing and Canceling Jobs

You can edit the schedule for pending jobs, and delete pending and complete jobs. You can't modify or delete jobs that are processing.

To edit or delete jobs:

1. Click **Application**, and then click **Jobs**.
2. To change when a job runs, click •••, then **Edit** and modify the schedule.

> **Note:**
>
> When changing the frequency of a job, the job will not run at the new frequency until the next time it runs at the previous frequency; for example, if the job is set up to recur daily and you change the frequency to hourly, the job won't start running at the new hourly frequency until after the scheduled daily job runs the next day. If you want the job to begin running at the new frequency sooner, Oracle recommends that you delete the scheduled job and create a new one.

3. To delete jobs, select them, click •••, and then **Delete**.

Duplicating Jobs

Use the **Save As** option to create a duplicate of an existing job, and then update it to avoid starting over making selections each time you create a new job.

Save As is supported for the following job types:

- Export Data
- Import Data
- Export Metadata
- Import Metadata
- Refresh Database
- Clear Cube
- Import Exchange Rates

> **Note:**
>
> For descriptions of the job types, see Job Types.

To duplicate a job:

1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then select one of the following actions:
   - **Export Data**
   - **Import Data**

- **Export Metadata**
- **Import Metadata**
- **Refresh Database**
- **Clear Cube**
- **Import Exchange Rates**

3. On the listing page, click ••• in the **Actions** column next to the job that you want to duplicate, and then select **Save As**.
4. Enter a name for the new job, and then click **OK**.

After the duplicate job is created, you can then open the job and update it. The error file for the duplicate job will automatically use the new job name as a prefix.

Canceling Rules Jobs and Ruleset Jobs

You can cancel ruleset or rule jobs that are processing and display in **Recent Activity**. To cancel these jobs, click **Application**, then click **Jobs**.

To cancel a rules job, click  beside the job, then  , and then **Cancel**.

To cancel a ruleset, click  , then  on **Job Details**, and then **Cancel**.

Downloading Export Files From Your Outbox

After running a metadata or data export job, you can download the export file from the Outbox.

To download files:

1. Click **Application**, and then click **Jobs**.
2. Under **Recent Activity**, click the job.
3. At the top of **Job Details**, click the export file option to select a download location.

16

Auditing Tasks and Data

Understand how to audit tasks performed by users and how to view audit details.

Related Topics

- Auditing Overview
- Enabling Audit Tracking
- Viewing Audit Details

Auditing Overview

Use the Audit feature to view tasks performed by users. You can filter audited tasks by audit type (for example, Data or Clear Cell Details), date range (for example, Yesterday or Last 60 Days), and user name.

You must be a Service Administrator to enable audit tracking, and to view and export audit information.

These are the types of user activities the system can log in the task audit:

Table 16-1 Tasks That Can be Audited

Audit Types	Tracked Changes
Dimension Administration	• Dimension hierarchy: Adding a member or dimension; moving, deleting, and changing properties; renaming a member and dimension • Performance settings: Resetting a dimension's dense or sparse setting, changing the order of dimensions
Alias Table Administration	Changes to alias tables: Creating, copying, renaming, deleting, and clearing
Data	• Cell values • Supporting detail • Account annotations • Cell-level documents

> **Note:**
>
> The audit log only captures modifications if the data change occurs within a data form. If a modification occurs as part of, for example, a business rule calculation, a data map, or a direct data load to Oracle Essbase, it will not be captured in the audit log.

Table 16-1 (Cont.) Tasks That Can be Audited

Audit Types	Tracked Changes
Launch Business Rules	Updates from calculation scripts and business rules (including runtime prompts)
Form Definition	Forms: Creating, moving, deleting forms (The audit record doesn't record how the design changed.)
Form Folder Administration	Folders: Created, moved, and deleted
Security	Access permissions to dimension members, forms, form folders, business rules, and task lists
Users Administration	Users added, changed, and deleted
Groups Administration	Groups added, changed, and deleted; users added and removed
Tasklist	Task lists: created, updated, saved, moved, and deleted
Copy Data	Users' selections for Static Dimensions, Source Dimension, and Destination Dimension, including supporting detail, cell text, cell attachments, and data without any details
Clear Cell Details	Users' selections for clearing supporting details, comments, and attachments
Variables	Variables (substitution variables and user variables): Added, changed, and deleted

The Audit page displays the following information:

- Audit (task)
- Source
- Action
- User
- Name
- Date
- Details
- Property
- Old Value
- New Value

All columns can be sorted in ascending or descending order.

Enabling Audit Tracking

Audit tracking isn't enabled by default. Service Administrators must enable auditing so data changes can be tracked.

To enable auditing:

1. On the Home page, click **Tools**, and then click **Audit**.
2. From the **Audit** page, click **Enable Auditing**.
3. From the **Enable Auditing** page, select an audit task or tasks, and then click **Save and Close**.

Viewing Audit Details

You can view and export up to 180 days of audit details in the business process interface.

To view audit details:

1. On the Home page, click **Tools**, and then click **Audit**.

   If auditing is enabled, by default the **Audit** page displays the audit records for **Data** tasks for the **Last 7 Days**. To enable auditing, see Enabling Audit Tracking.

2. To filter, click  , and select filter criteria:

   - **Audit Types**: Select one or more, or **All**. For a list of audit types and descriptions, see Auditing Overview.
   - **Date Range**: Select **Today**, **Yesterday**, **Last 7 Days**, **Last 30 Days**, **Last 60 Days**, or **Last 180 Days**.

     > **Note:**
     >
     > – You can't select a date range prior to 180 days from today. The date range you select must occur between today and 180 days prior to today (including today).
     >
     > – The business process retains up to 365 days of audit details from the current system date. To export more than 180 days of audit details, you can use the EPM Automate `exportAppAudit` command or the REST API Export Audit job. See *Working with EPM Automate for Oracle Enterprise Performance Management Cloud* or *REST API for Oracle Enterprise Performance Management Cloud*.

   - **User Name**: Enter a user name or click  to search for the user.

3. When you're done selecting filter criteria, click **Apply**.

   The grid displays the first 200 records from the audit table that match the filter criteria. You can view a legend at the bottom of the table if the filter criteria has more than 200 records in it.

4. To export the audit information to a Microsoft Excel spreadsheet, click **Export**, and follow the download instructions.

   When you select the **Export** option, the system exports all of the records matching the filter criteria to an `xlsx` file (format for Microsoft Excel versions 2007 and later).

> **Note:**
>
> If data auditing is enabled, users can see what data has changed by selecting **Actions**, and then **Change History**.

17

Managing Data Validation

Learn how to build data validation rules that are checked when conditions are met in forms. This chapter includes helpful scenarios that give examples of how data validation can help implement business policies.

Related Topics

- Creating and Updating Data Validation Rules
- Formatting Cells
- Viewing Data Validation Rules
- Order of Evaluation and Execution for Data Validation Rules
- Conditions Supported by the Rule Builder
- Data Validation Conditional Operators
- Data Validation Rule Scenarios

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.

Sample scenarios that can be addressed using data validation rules are described in Data Validation Rule Scenarios.

Defining data validation rules involves these main tasks:

- Identifying the data cells or location that you want to display with validation messages or in different colors when conditions are met.
- Identifying the cells that need to participate during rule evaluation, and defining the rule accordingly.
- Creating the data validation rule at the location identified, as described in this topic.

To create and update validation rules:

1. On the Home page, click **Navigator**  , and then under **Create and Manage**, click **Forms**.
2. Create or edit a form, and then on the **Form and Ad Hoc Grid Management** page, click **Layout**.
3. In **Layout**, right-click the grid, row heading, column heading, or cell for which you want to add or update the validation rule.

> **Note:**
>
> When you hover the cursor over cells in **Layout**, 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.

4. Select **Add/Edit Validation Rules** to create or update rules.
5. To add a rule, click  and enter a name and description for the rule.

   If necessary, move the rule by selecting an option from **Location**. To create a rule similar to an existing rule, click  and then update the rule. To view rules, click **View Rule**. See Viewing Data Validation Rules.
6. Update the rule.
   a. For **Condition**, select an option to begin the condition statement: **If**, **Else If**, **Else**, **Then**, **Check Range**, or **Range**.

      The first part of a rule must include an If condition. Rules must also contain some form of Then condition. See Conditions Supported by the Rule Builder.
   b. For **Source Type**, select an option for evaluation by the rule.

      The **Source Type** list displays the appropriate options for the Condition. For example, If conditions can include **Current Cell Value**, **Cell Value**, **Column Value**, **Row Value**, **Member Name**, **Member**, **Cross-Dim Member**, **Account Type**, **Version Type**, **Var Reporting Type**, **UDA**, or **Attribute**. For detailed information about each type of condition, see Conditions Supported by the Rule Builder.
   c. If applicable for the selected **Source Type**, enter a value in **Source Value** by selecting an option or entering a free form value.
   d. Select an operator for the evaluation: **=**, **!=**, **<**, **<=**, **>**, **>=**, **Equals**, **Not Equals**, **Contains**, **Starts With**, or **Ends With, In, or Not In**.

      For examples, see Data Validation Conditional Operators.
   e. Select an option for the appropriate **Target Type** for the rule.
   f. Update conditions by clicking an icon in the **Actions** column to the right of the condition builder:

      -  : Add a condition next to the current row.
      -  : Delete a condition at the current row.
   g. Select conditions or condition blocks to update.

      To update condition blocks, click an icon in the **Condition** area, at the top of the condition builder:

      -  : Add a condition block within the validation rule starting with If. You can expand or collapse the condition. See Conditions Supported by the Rule Builder.
      -  : Delete the selected condition block.
      -  : Delete selected conditions, and copy them to paste in a new location.

- : Copy selected conditions.
- : Paste selected conditions to a new location.
- : Group the selection within a condition, and add a grouping parenthesis. In addition to groupings that you set, the If statement in the grouping block is grouped when a condition is grouped, and the **Custom Grouping** option is enabled.
- : Ungroup the selected conditions. The grouping parentheses are removed from the selected condition. One grouping is removed from the condition each time Ungroup is selected.

  You can also set your own grouping for conditions by selecting **Custom Grouping**, then setting up the grouping in the rule definition area.

  Selected conditions are displayed as shaded. To clear selected conditions, click once more to the left of the **Condition** column.

7. Click in the rightmost column to add processing instructions.

> **Note:**
>
> If the icon isn't displayed, ensure that the rule is valid and that it permits cell processing instructions. For example, cell processing instructions are included for Else, Range, and Then conditions. The icon isn't displayed until all required columns are selected for a rule.

See Formatting Cells.

8. When you're ready to enable the rule to make it available in the form, select **Enable Validation Rule**.

> **Tip:**
>
> While you're building a rule, you can save the rule without enabling it. After any errors are resolved and the rule is ready to be validated and used, you can enable and save the rule to make it available in the form. You can temporarily disable a rule by clearing **Enable Validation Rule**.

9. When you finish updating the rule, validate the rule:
   a. Click **Validate**.

      The validation status displays at the top of the dialog box. You must correct errors before you save changes. If you close the dialog box without validating rules and fixing errors noted during validation, updates are not saved.

   b. After fixing any errors noted during validation, ensure that **Enable Validation Rule** is selected above the rule definition area to enable the rule for the application.

   c. After the rule is validated, click **OK**.

10. **Optional**: In the **Form and Ad Hoc Grid Management** page, view and update rules:

- In the **Validation Rules** pane on the right side of the **Form and Ad Hoc Grid Management** page, add, edit, or delete rules by clicking  ,  , or  .
- If multiple rules are defined at the same location, you can change the order in which rules are processed when rules have the same precedence. To move a rule up, down, or to the top or bottom of the list, select the rule and click the arrows. See Order of Evaluation and Execution for Data Validation Rules.
- Select form validation rules options:

Table 17-1 Form Validation Rules Options

Option	Description
Validate only for pages with existing blocks	When enabled, the system figures out which page combinations have potential blocks and runs the validations only for those page combinations. There are a few exceptions to this. If a page combination has any Dynamic Calc, Dynamic Calc and Store, Label only, or Store with one child member, then that page is always loaded.
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 Service Administrator, which means the user's security will be applied to the form members.

11. In the **Form and Ad Hoc Grid Management** page, preview and validate the form, resolve any validation errors, and then save changes.

    For forms that have data validation rules enabled, rules are validated when the form is loaded or saved. Data validation rules are saved when the form is saved. See Creating Forms.

    When users open the form, they can view and resolve validation messages using the **Data Validation Messages** pane.

    See Resolving Data Validation Errors in *Working with FreeForm*.

Formatting Cells

After a rule is set up, use the Process Cell dialog box to set how cells display in forms.

To format cells:

1. In the Data Validation Rule Builder dialog box, click  in the right-most column.

> **Note:**
>
> If the icon isn't displayed, ensure that the rule is valid and that it permits cell processing instructions. For example, cell processing instructions are included for Else, Range, and Then conditions. The icon isn't displayed until all required columns are selected for a rule. See Creating and Updating Data Validation Rules.

2. In the Process Cell dialog box, set how the cell should appear in forms if the conditions defined by this rule are fulfilled.

   When the rule is enabled, it doesn't validate unless you specify at least one of these options: a cell background color or a validation message.

- To add or update the cell background color, click  . To remove the cell background color, click  .
- To display a validation message for the cell, enter the message in the **Validation Message** field. Users see this text when they select **Show Data Validation Messages** in the context menu that appears when you hover over the cell in the form. It also appears as a link in the **Data Validation Messages** pane if data cells are flagged by the validation rules and the **Display message in the Data Validation Messages pane** check box is selected. For information on viewing and resolving data validation errors, see *Working with FreeForm*.

3. Click **OK**.

   Updates for the rule are displayed in the Process column for the rule. If you specified a cell color, that color is displayed. You can preview a validation message by hovering the cursor over the Process column.

Viewing Data Validation Rules

After data validation rules are set up with processing instructions, you can use the View Rule dialog box to view all rules that apply to the selected grid, row, column, or cell.

To view data validation rules:

1. In the **Data Validation Rule Builder** dialog box, click **View Rule** to view all the rules at this level (grid, row, column, cell) for this location in the form.
2. Select the rule name, and then double-click the rule or click **OK** to view details.

Order of Evaluation and Execution for Data Validation Rules

For data validation rules in forms, the precedence for rule evaluation depends on condition priority, location of the rule, and position of the rule in the rule list (if multiple rules exist in the same location). First, cell-level rules are processed. Next, rules at the column level are processed, and then row-level rules are processed. Finally, rules at the grid level are processed. The rules are evaluated based on their position in the rule list within each level.

Location and position determine the order in which the rule will be processed. However, the priority of the processing instructions determines which rule is applied to the data cell. So, if a cell-level rule contains processing instructions with priority 4 and a grid-level rule contains processing instructions with priority 5, the grid-level rule is applied to the data cell. If all rules have processing instructions with the same priority, the first processed rule wins. The priority is based on whether the cell processing instructions specify a validation message, a color, the **Do Not Promote** promotional path option, or a combination of these settings.

Table 17-2 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

Table 17-2 (Cont.) Priority for Rules in Forms

Default Condition Priority	Validation Message	Color	Do Not Promote
5 (highest)	X	X	X

Conditions Supported by the Rule Builder

These conditions are supported by the data validation rule builder: If, Else, Else If, Then, Check Range, and Range.

For details and examples of the values supported by these conditions, see these sections:

- **If, Else, Else If**: If Condition Values
- **Then**: Then Condition Values
- **Check Range, Range**: Range Condition Values

If Condition Values

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

Current Cell Value

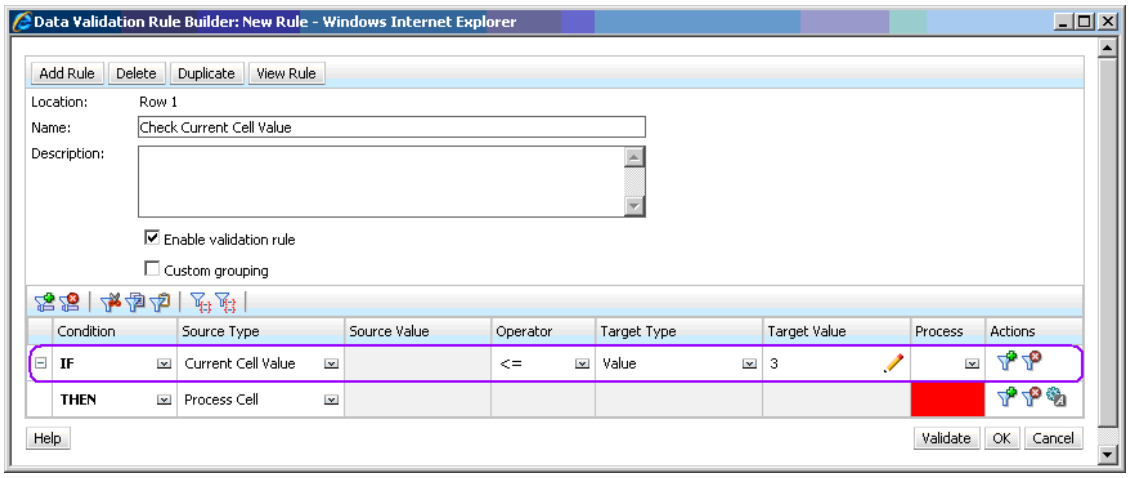
Action:

The action is performed when the value in the current data cell on which the rule is invoked satisfies this condition.

Operators:

The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, or Ends With. These operators act on the selected target value, which can be a free form value, cell value, column value, row value, or cross-dimension member.

Condition Definition:



Condition Evaluation:

With the condition shown in the previous figure, the cells in Row 1 with member Row_Member1 will turn red when the condition is evaluated.

		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

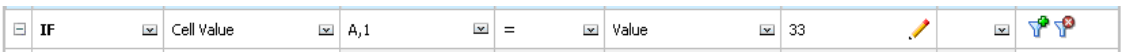
Action:

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:



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:

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
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

Row Value

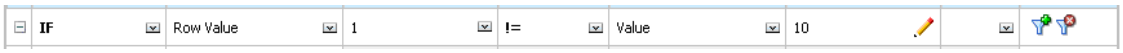
Action:

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

Operators:

The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, Ends With. These operators act on the target value selected, which can be a free form value, cell value, column value, row value, or cross-dimension member.

Condition Definition:



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
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

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.

> **Note:**
>
> If a form validation rule uses the cross-dim operator, the cross-dimension cell must be represented on the form. The column or row containing the cross-dimension cell might be hidden on the form if the user does not want to see it there.

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:

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
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

Member Name

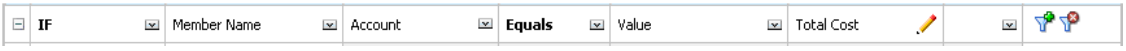
Action:

The action is performed if the current data cell on which the rule is invoked has the specified dimension member in its intersection. The member name for the selected dimension should be in the cube for which the form is created.

Operators:

The operators can be Equals, Not Equals, Contains, Starts With, or Ends With. The target value, which is the member name, is selected or entered free form.

Condition Definition:



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  , and then clicking **Variables** in the Member Selection dialog box. You can use the member selection function selector to select functions for the attribute, such as NotEqual and GreaterOrEqual.

When using attribute values in data validation rules, keep in mind the way attributes are evaluated. If a rule references one or more attributes from one or more 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 Selecting Attribute Values as Members.

Condition Definition for Member Source Type

Condition Definition for Attribute Source Type

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:

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:

Variance Reporting Type

Action:

The action is performed if the current cell on which the rule is invoked has an account with the specified variance reporting type in its intersection. Refers to the available variance reporting types, Expense and Non-Expense.

Operator:

The available operator is Is.

Condition Definition:

UDA

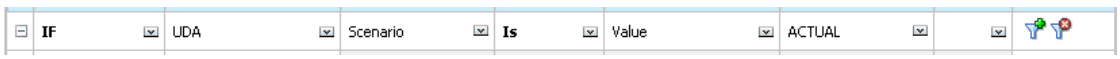
Action:

The action is performed if the current cell on which the rule is invoked has this UDA associated with the specified dimension's member in its intersection. The UDA reference is selected based on the selected dimension. The condition is based on the UDA for this dimension being equal to the selected value. You must select the UDA value from the drop-down list.

Operator:

The available operator is Is.

Condition Definition:



Attribute

Action:

The action is performed if the current cell on which the rule is invoked has this attribute associated with the specified dimension's member in its intersection. The attribute reference is selected based on the selected dimension. The condition is based on the attribute for this dimension being the selected target value.

Operator:

The available operator is Is.

Condition Definition:

Then Condition Values

Action:

The Then conditions supported by the data validation rule builder support Process Cell conditions only. To enter Process Cell Conditions, see Formatting Cells.

Condition Definition:

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

Range Condition Values

The Check Range and Range conditions are used together. These conditions can be used in the data validation rule builder in a Then clause or standalone.

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

Check Range

Action:

Defines the value that needs to be in a specific range.

Value:

This value can be the Current Cell Value or the value in a particular Row, Column or Cell.

Range

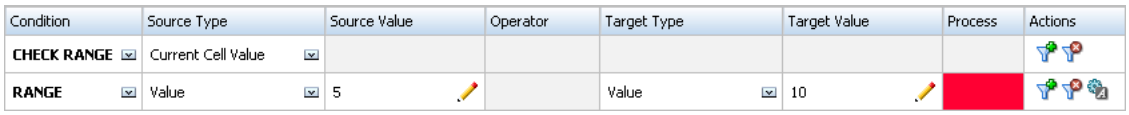
Action:

Defines a valid range for the value defined in the Check Range condition. This range includes all values that are >= 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:



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

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 (percentage and date) are treated as double.

Table 17-3 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

Table 17-3 (Cont.) Examples of Results for Starts With, Ends With, and Contains

Operator	Compare Value	Compare To Value
	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"

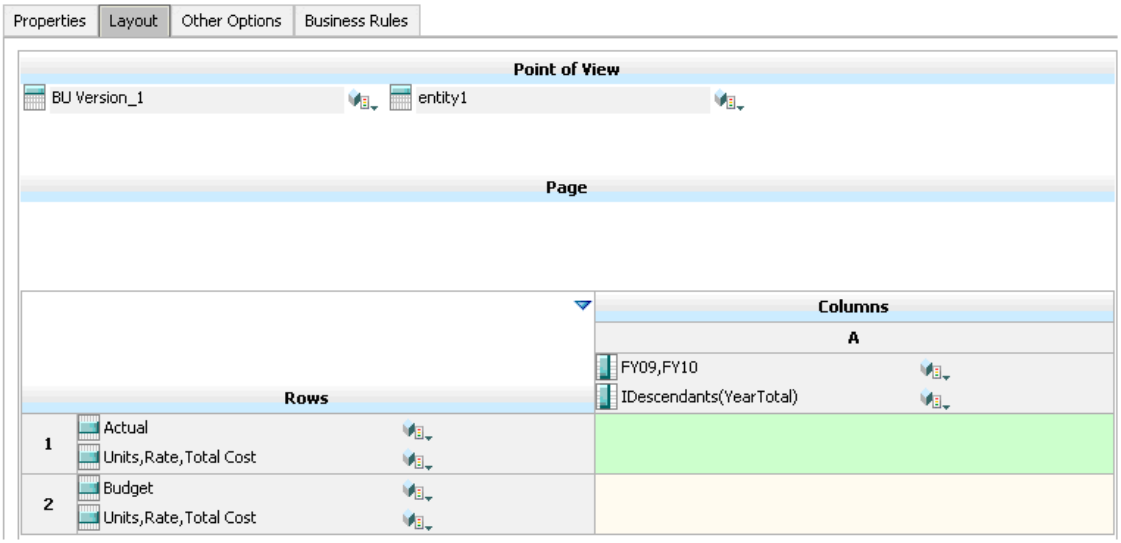
Data Validation Rule Scenarios

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

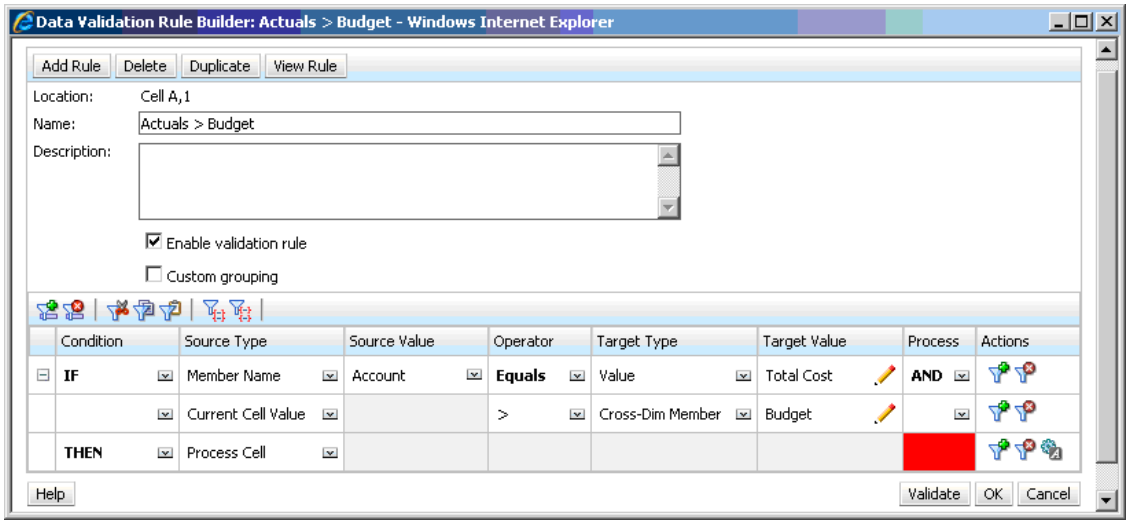
Scenario 1

John is hired by a company called Acme, Inc. as a consultant to design forms and implement data validation rules that enforce some of the company policies. He is asked to implement a validation rule that flags Actual amounts in red if the Total Cost in actuals exceeds the budgeted amount. This test must be repeated for each year and time period in the application. John designs the form and adds a data validation rule at cell level using a cross-dimension member, as shown in the following figures.

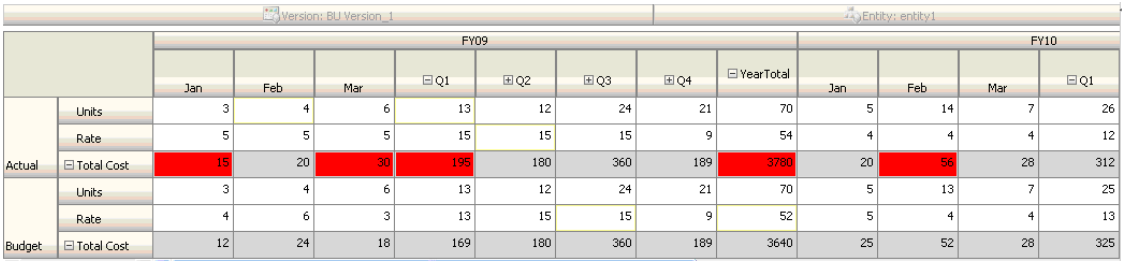
Form Layout at Design Time:



Data Validation Rule at Design Time:



Form at Data Entry Time with Data Validations Applied:



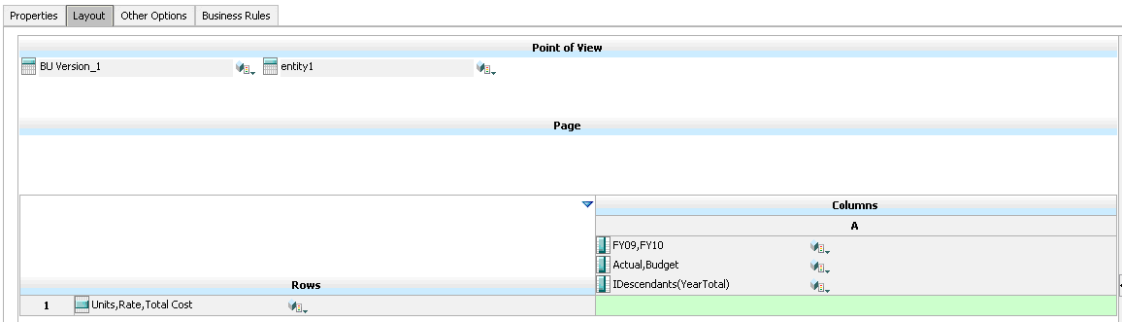
Tips:

- John can split Total Cost into its own segment and apply the data validation rule at that segment for a slight performance gain. However, doing so would increase maintenance as new accounts and scenarios were added to the form.
- If the requirements changed such that only the YearTotal Period in Actual had to be flagged in red, John would have two options. The best option is to add an IF entry to check if the Period member is YearTotal. Another option is to split the YearTotal member into a separate column for better performance. However, doing so would break the spreading logic, the column header for Year would repeat, and the form would be harder to maintain as new years were added.

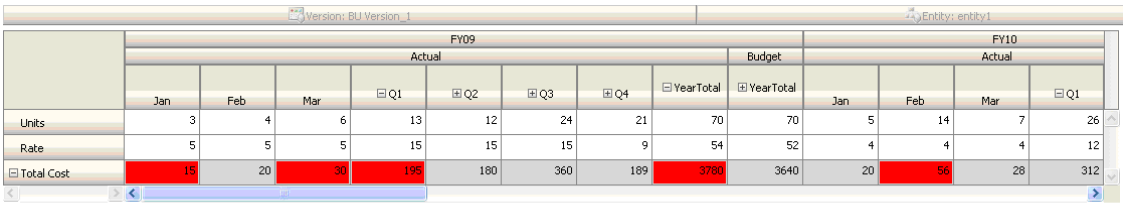
Scenario 2

After reviewing the form designed by John in Scenario 1, Acme decides that they want Budget on the column instead of the row. To implement this requirement, John can move members within the axes to change the form layout. However, he doesn't need to update the data validation rules. John updates the form as shown in the following figure.

Form Layout at Design Time:



Form at Data Entry Time with Data Validations Applied:



Scenario 3

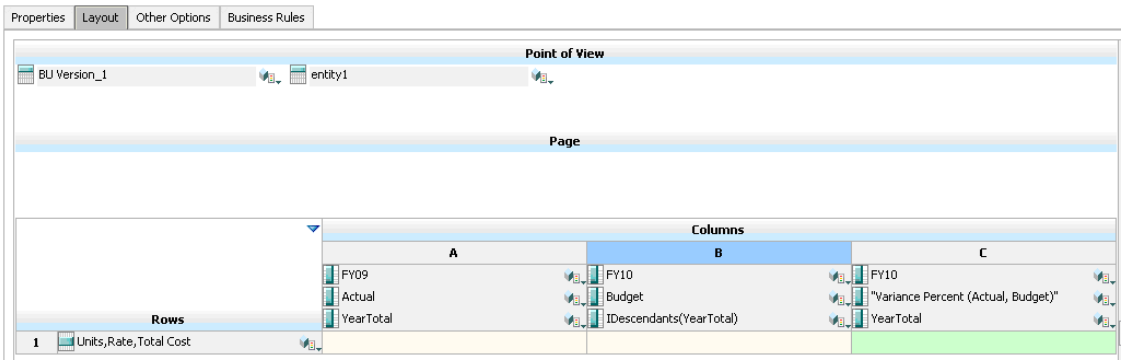
Following the successful rollout of these forms, John is asked to implement the next policy, which is to ensure that this year's Budget amounts are not significantly higher than previous year's Actual amounts. If the difference is greater than 5%, then flag the difference in red.

John decides to use a member with a member formula to calculate the variance between this year's Budget and the previous year's Actual amount. He adds this member formula:

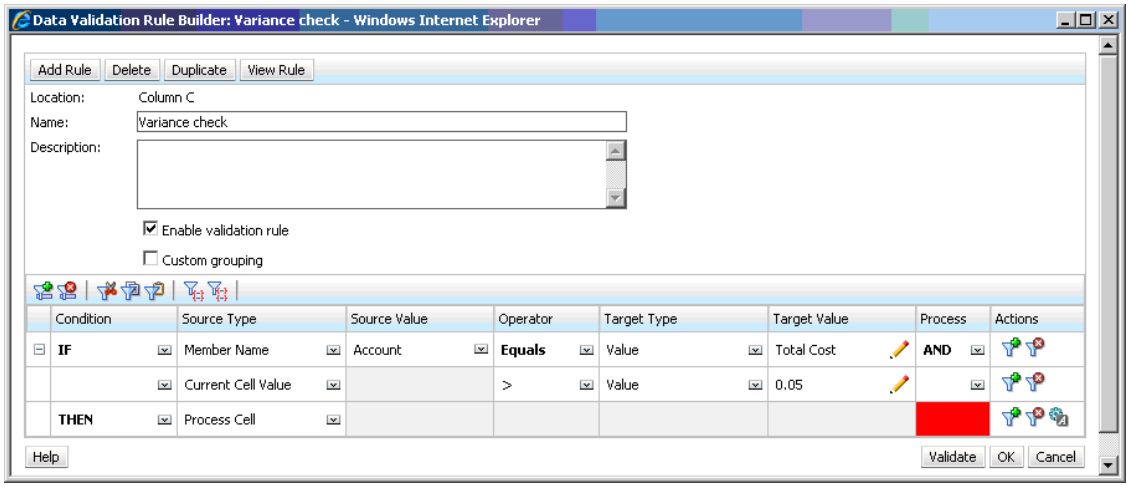
```
@varper(@Prior("Actual", 1, @Relative("Year", 0)), budget)/100;
```

John designs the form and adds a data validation rule at cell level, as shown in the following figure. He uses Member Name to apply the validation only to Total Cost.

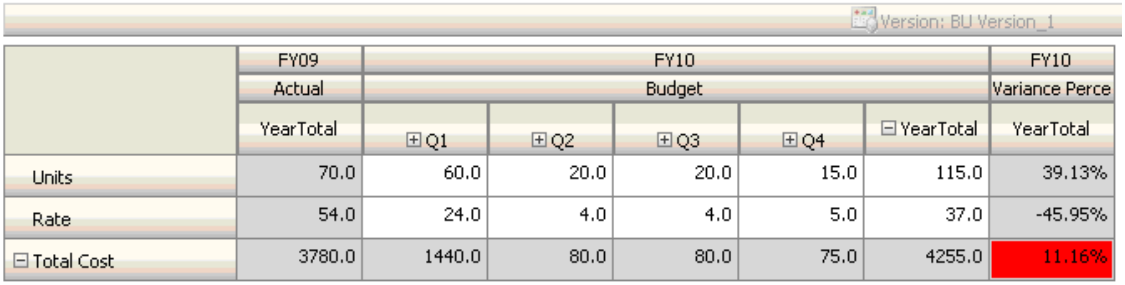
Form Layout at Design Time:



Data Validation Rule at Design Time:



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%

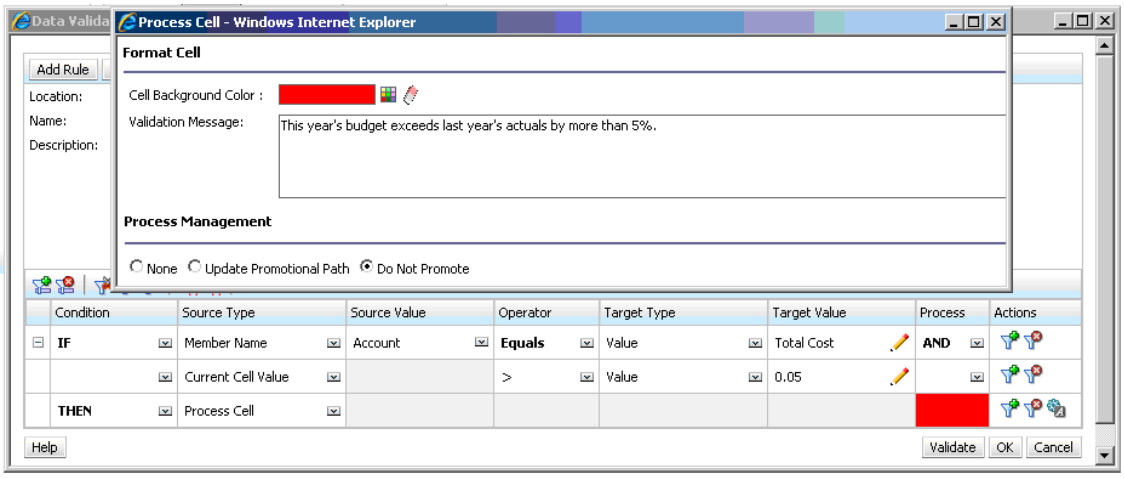
Tips:

- If John isn't allowed to change the outline, or if he experiences performance issues related to member formulas, he can use a formula column. See Designing Forms with Formula Rows and Columns.
- John defines the rule at the Variance Percent column for these reasons.
  - It improves performance. The rule is evaluated only on the cells in the Variance Percent column. If the rule had been assigned to YearTotal, it would have to be evaluated for all Time Periods for the current year budget.
  - It helps users respond to the data validation message. John can add a message to the Variance Percent column stating that the variance is higher instead of adding it to YearTotal. This way, users don't have to look for Variance Percent to determine the difference.
- John could have flagged both YearTotal and Variance Percent in red if this had been part of the requirement.

Scenario 4

In addition to flagging the cell in red, the rule is also required to prevent anyone from promoting the approval unit if this year's Budget is significantly higher (> 5%) than the previous year's Actual amounts. To implement this requirement, all John needs to do is edit the data validation rule's processing instructions and select **Do Not Promote**, as shown in the following figure.

Data Validation Rule at Design Time:



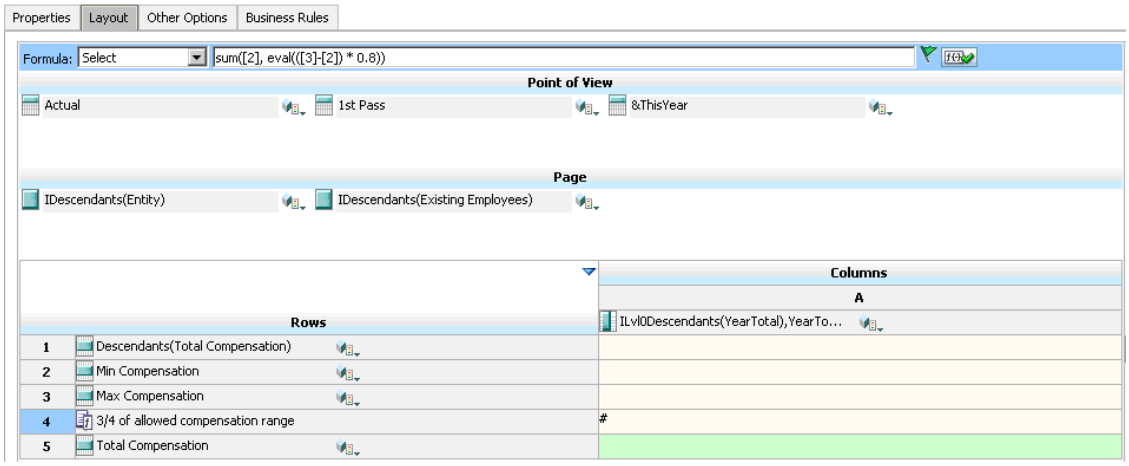
Scenario 5

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.

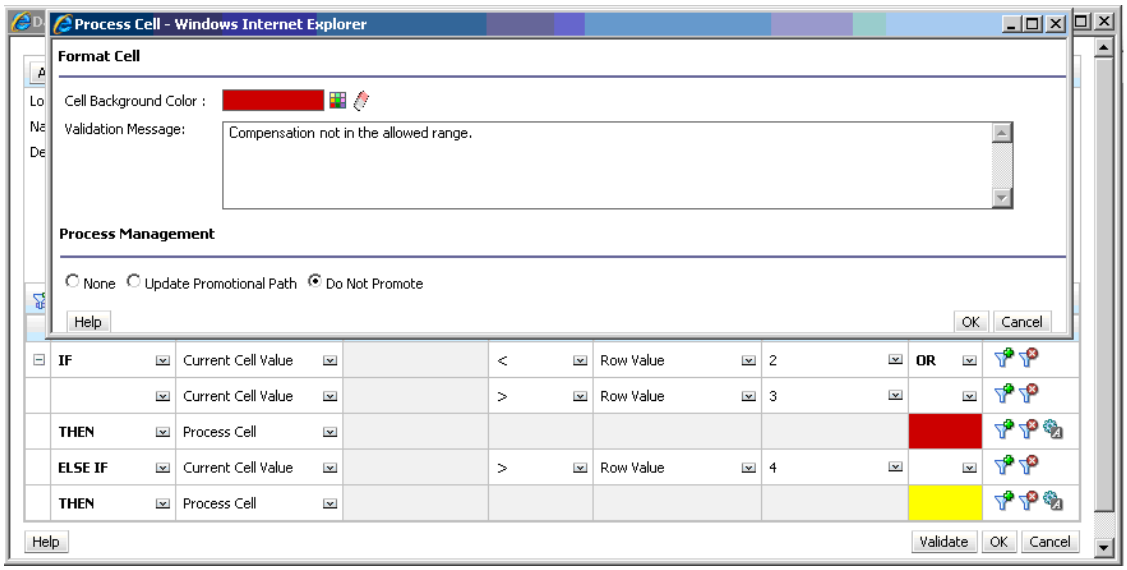
If Total Compensation is greater than ¾ of the compensation range, a validation message is provided, and the approval units must be approved by a human resource manager. If the value is less than Min and greater than Max, an error is generated, and users can't promote their approval units.

John opens the Employee Expenses Summary form in the Form Management dialog box. The form has employees and departments on the page, accounts (such as Total Compensation) on the row, and time period on the column. To make validations easier to build, John adds a calculated row to calculate 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.

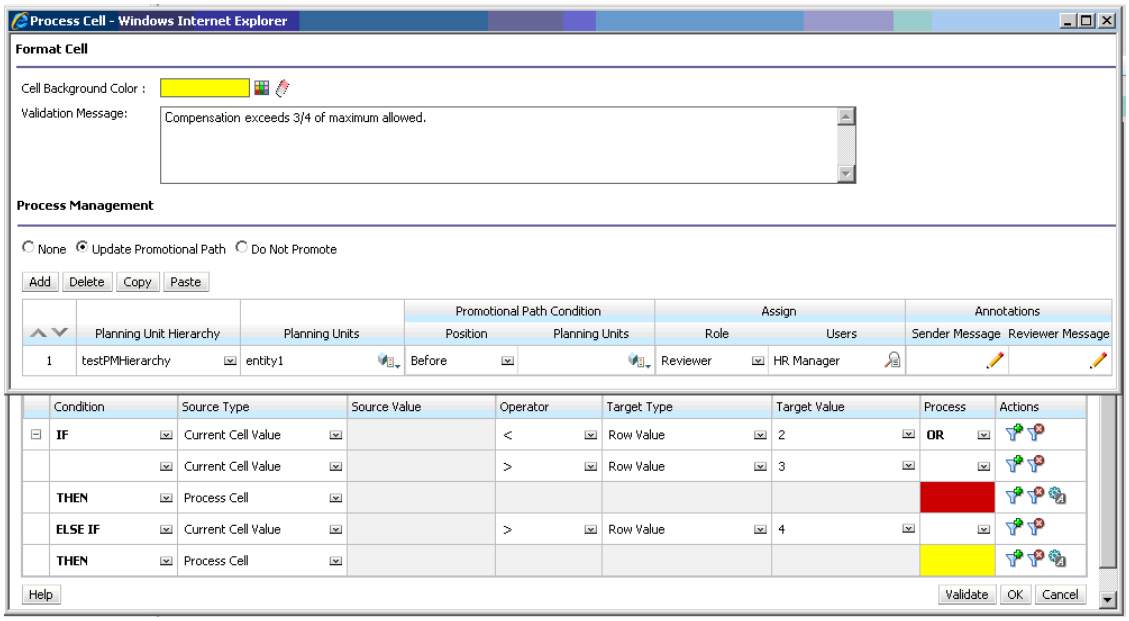
Form Layout at Design Time:



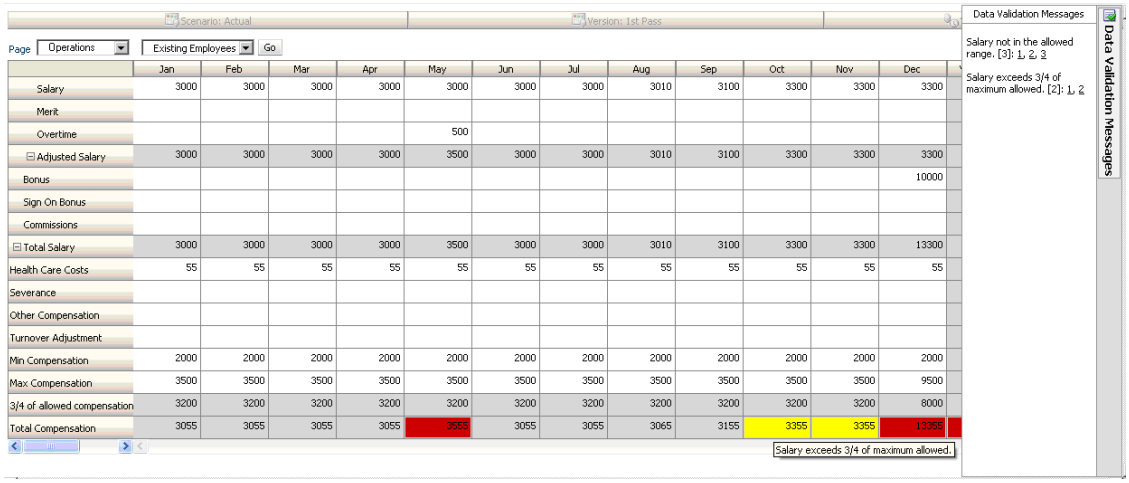
Data Validation Rule to Stop Promotion of Approval Units:



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



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



18

Managing Application and System Settings

Set application defaults and system settings such as number formatting, notifications, number of items displayed on a page, aliases, date formatting, and assigning application ownership.

Related Topics

- What Application and System Settings Can I Specify?
- Defining User Variables
- Customizing Your Display
  Change the theme of your display or add your company logo or a background image to the Home page.
- Announcing Upcoming Events
- Specifying Artifact Labels

What Application and System Settings Can I Specify?

You can control many aspects of the application and the system, such as:

- How to display thousands, decimals, and negative numbers in forms
- Define the actions about which you want to be notified
- Display the full names of users rather than user IDs
- Set reporting options
- Assign application ownership to another Service Administrator

To change application and system settings:

1. Click **Application**, and then click **Settings**.
2. Specify defaults for the current application. For descriptions of the settings, see the Application Settings table below.
3. Specify system settings. For descriptions of the settings, see the System Settings table below.
4. Click **Save**.

Table 18-1 Application Settings

Application Setting	Description
Alias Setting	For option descriptions, see Specifying a Default Alias Table and Setting Member and Alias Display Options.
Notifications	Enable notifications for task lists, and job console.

Table 18-1 (Cont.) Application Settings

Application Setting	Description
Page	Set defaults for indenting members on a page and setting the number of items on the page drop-down. Note that the **Number of Items on the Page Drop-down** option lets you shorten the member list so that the **Search** box can be more easily seen. If the list is shortened to 10 members, for example, then you won't need to scroll to see the **Search** box.
Other Options	Set these other configuration options: • **Date Format** • **Attribute Dimension Date Format** • **UI Display**: Choose **Standard Interface** or **Simplified Interface**. • **Partial Grid Fetch Size (Rows,Columns)**: Enter number of rows and columns in the format `nn,nn` • **Suppress Application Management Options in Smart View** • **Enable Data Load for Ad Hoc Read-Only Role**: Default is **No**. Select **Yes** to enable data load for users with Adhoc read-only role. • **Set Number of Seconds Before Rules Run in Background**: Enter a value between 0 and 600. **Note:** Rules that are set to automatically run when a form is loaded or saved never run in the background. • **Client Log Level**: All error, warnings, and informational messages from the connected data source are displayed when they occur, but you can choose which of these message levels to record in a browser console log file. This setting is only available for web interfaces using Oracle JET technology. Select a message level to display and record: – **None**: Suppress all messages. – **Information**: All messages, including warnings and errors—recommended to diagnose problems. May adversely impact performance. – **Warning**: Warnings and error level messages. May adversely impact performance. – **Error**: Error messages only—recommended for general use. Has minimal impact on performance. – **General**: Information-level messages plus all server responses and requests. Adversely impacts performance. • **Enable User Formulas in Ad Hoc**: If you select **Yes**, in an ad hoc grid, when you hover over a cell with a member formula attached, you see the member formula associated with the cell and you can perform calc on the fly. • **Filter Out Excluded Members in Segment Drop-down**: Default is **Yes**. Choose **No** to display the excluded members in the row drop-down POV in forms. • **Forms Version**: Specify a forms version. For existing applications, the default is **Forms 1.0**. Enabling **Forms 2.0** is only supported if Redwood Experience is enabled. Newly created applications will default to **Forms 2.0**. This setting is application-specific and can vary across applications.

Table 18-1 (Cont.) Application Settings

Application Setting	Description
Digital Assistant Settings	If you have implemented Oracle Digital Assistant for Enterprise Performance Management, specify configuration settings for the assistant. **Note**: These settings are part of a larger configuration process to enable you to work with Oracle Digital Assistant for Enterprise Performance Management. You must complete the configuration steps before using the assistant. See Configuring Applications in *Getting Started with the Digital Assistant for Enterprise Performance Management* . • **Channel ID**: Enter the channel ID generated when you created the Oracle Web channel for Oracle Digital Assistant for Enterprise Performance Management. • **Service Name**: Enter the Digital Assistant URL, which is the Oracle Digital Assistant Service URL that you see when you log on to the service. Enter the URL without either http:// or https:// preceding it.
Smart View Add-on	Select the **Google Sheets** check box to enable support for Smart View in Google Sheets. When selected, users can connect to their applications from Google Sheets and use Smart View features to view and analyze data in Google Sheets.

Table 18-2 System Settings

System Setting	Description
Display Users' Full Names	When selected, the system displays the user's full name (for example, Max Hennings). When cleared, the system displays the user's ID (for example, VHennings).
Include Shared Members in Cube Refresh	When selected, shared members will inherit the highest security access based on a combination of the access assigned to the base member and parent of shared member. When cleared, shared members will inherit the security access assigned to the base member.

Table 18-2 (Cont.) System Settings

System Setting	Description
Default Ancestor Access	Select an option to specify the default user access to ancestor members in ad hoc grids and the Member Selector: • **None**: By default, users are not able to see the ancestor members of members that they have access to unless they are explicitly granted access to them. When this option is selected, the system works the same way it did prior to the 24.03 update. • **Read**: By default, users are granted Read Only access to the ancestor members of members that they have access to unless they are explicitly granted **Write** or **Display** access, which would override this default. • **Write**: By default, users are granted Write access to the ancestor members of members that they have access to unless they are explicitly granted **Read** or **Display** access, which would override this default. • **Display**: By default, users are granted Display Only access to the ancestor members of members that they have access to unless they are explicitly granted **Read** or **Write** access, which would override this default. **Note:** Members with Display Only access will be displayed, but instead of data values in the cells associated with those members, users will see #NoAccess. See Accessing Ancestor Members in Ad Hoc Grids in *Working with Oracle Smart View for Office* for more information on working with the **Default Ancestor Access** setting.
Email Character Set	Select **UTF-8** or **Regional Setting**.
Business Rules Notification	If set to **Yes**, notifies users or groups when rules (which are enabled for notification in Calculation Manager) are completed or encounter errors. In **Notify These Users**, select the users or groups to notify.
Allow Drill Down on Shared Members in Ad Hoc	• **Yes** enables drilling on shared members in an ad hoc grid. When set to **Yes**, the **Drill to All Levels in Base** check box is displayed. Note that block suppression is not supported when this option is set to **Yes**. To disable block suppression: – In Oracle Smart View for Office, in **Options**, **Data Options**, disable **Suppress Missing Blocks**. – In the web, in **Preferences**, **Ad Hoc Options**, disable **Missing blocks on rows**. • **No** disables drilling on shared members in an ad hoc grid.

Table 18-2 (Cont.) System Settings

System Setting	Description
Drill to All Levels in Base	Allows ad hoc grid users to drill down (or zoom in) from shared members to all levels within the base hierarchy. This check box option applies to Standard-mode applications only. The **Drill to All Levels in Base** check box appears when **Allow Drill Down on Shared Members in Ad Hoc** is set to **Yes**. A shared member is often a parent member in its base hierarchy. With the **Drill to All Levels in Base** check box enabled, users can drill down from the shared member to the base hierarchy using the zoom options, **All Levels** or **Bottom Level**. When the **Drill to All Levels in Base** check box is disabled, the user may zoom in to the various levels of the base hierarchy one level at a time using the **Next Level** zoom option. Multi-cell zoom-in on shared members is not supported.
Enable Use of the Application for	Determines whether users can access the application in administration mode, such as during backups. When you select **Administrators**, if any non-administrative users are logged on to the application, they are forced off the system and will not be able to log on. To restore access to an application for all users, select **All users**.
Assign Application Owner	Assigns ownership of the application to another Service Administrator. For more information about application ownership, see Managing Application Ownership.
Enable the Display of Substitution Variables	Set how substitution variables display in the Member Selection dialog box when users respond to runtime prompts in business rules. **Display All** displays all substitution variables. **Display None** displays no substitution variables. **Enable Filtering** displays only substitution variables that are valid for the runtime prompt.
Suppression Mode	Choose a suppression behavior for ad hoc grids in the web and in Smart View for cases where rows and columns contain missing data or zeroes. • **Suppress Missing values only** (default): Suppresses rows, or columns, or both that contain No Data/Missing. • **Suppress Missing also Suppresses Zeros**: Suppresses rows, or columns, or both that contain both No Data/Missing and Zero.
Smart View Ad Hoc Behavior	All new and recreated applications are automatically set to **Standard**. For existing and migrated applications, choose to enable enhanced ad hoc features and behaviors in Smart View: • **Native** (default): Does not enable enhanced ad hoc features. Supported for all Smart View releases. • **Standard**: Enables enhanced ad hoc features. Supported for Smart View release 11.1.2.5.900 and later. For a complete description of the enhanced ad hoc features available in Standard mode, see Smart View Behavior Options in EPM Cloud in *Working with Oracle Smart View for Office*.
Use All Alias Tables on Refresh	Specify whether alias names entered into ad hoc grids are evaluated using the currently selected alias table or against all alias tables • **Yes**: The input is evaluated against all alias tables and all member names. • **No**: The input is evaluated against the currently selected alias table and all member names. This is the default setting. If the system cannot identify the input as a valid member name or alias, the input will be displayed as a comment.

Table 18-2 (Cont.) System Settings

System Setting	Description
Export EPM Cloud Smart List textual data during daily maintenance for incremental data import	Choose whether to perform a complete export during the daily maintenance process or to create an application backup: • **Yes**: Performs a complete export, such that data, including the business process Smart List data, can be incrementally imported to an application (this option may lengthen the maintenance process duration) • **No** (default): Creates an application backup during the maintenance process, such that data can be used as part of a full restoration Note that this setting applies only to Oracle Essbase that does not support hybrid. For more information, see Exporting Smart List Textual Data During Daily Maintenance for Incremental Data Import in *Oracle Enterprise Performance Management Cloud Operations Guide*.
Link Accounts by Default	For block storage (input) cubes, select whether to XREF linked account members by default: • **Yes** (default): XREFs will be created on account members, and the application will work the same way it has in earlier releases. • **No**: XREFs will not be created for account members, which may improve the application's performance. With **No** selected, after **Cube Refresh** is run, all existing XREFs on account members will be deleted, and non-source cubes will no longer show data from the source cube. Note that `HSP_LINK` and `HSP_NOLINK` UDAs on specific account members override the XREF setting for those account members. For example, if this option is set to **No** and you use the `@XREF` function to look up a data value in another cube to calculate a value from the current cube, you can add the`HSP_LINK` UDA to such members to create the `@XREF` function only for these specific members. If this option is set to **Yes**, HSP_NOLINK works the same way it worked in earlier releases and prevents XREFs from being created on specific members.
Attribute Dimension Reorder Threshold	Enter a threshold value between 0 and 500 (500 is the default value). For attribute dimensions, when the number of members under a given parent exceeds the specified threshold value, then the members will be reordered at the end of a metadata load instead of during the load. Depending on the specific shape of the attribute dimension, adjusting this number can sometimes have an impact on performance. In general, this setting can be ignored unless attribute dimension load times degrade beyond acceptable levels.
Set Reporting Options	If you're using the next-generation Reports reporting solution, Oracle recommends you use this setting to upload any TrueType fonts that your company uses to produce reports. To upload TrueType fonts, click **Report Settings**, click **Manage Fonts**, click  , and then select the font files to upload. See Working with the Reports Reporting Solution in *Working with FreeForm*.

Defining User Variables

You can define user variables to help users focus on particular members, such as their department's expenses. For example, you can create a form with entities on the rows, and a

user variable called Department. You can limit the number of rows displayed on the form by selecting a member for the Department user variable, such as Sales. Later, you can select another value for Department, such as Marketing.

To update user variables:

1. Click **Tools**, and then click **Variables**.
2. Click the **User Variables** tab.
3. Click  next to the variable to change.
4. On **Member Selection**, select members.

Customizing Your Display

Change the theme of your display or add your company logo or a background image to the Home page.

On the **Appearance** page, you can change the general look and feel of your Oracle Enterprise Performance Management Cloud environment.

All newly created or re-created EPM Cloud services, business processes, and applications use Redwood Experience as the default theme. Redwood Experience not only provides an attractive look and feel, but it also includes certain features, such as dynamic tabs, that are not available in the other themes. For more information about Redwood Experience, see Using Redwood Experience in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

If you opt not to use Redwood Experience, you can choose instead from a list of predefined classic themes with different background colors, icon styles, and so on. You can also add a branding logo and background images to the Home page and hide the business process name. For general information about using the Home page, see About the Home Page.

> **Note:**
>
> You can set your profile picture to display at the top of the Announcements panel of the Home page in **User Preferences**. Click **Tools**, and then **User Preferences**.
>
> For more information, see Setting Your Profile Picture in *Working with FreeForm*.

To customize your display, see Configuring EPM Cloud Appearance in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*.

Announcing Upcoming Events

Create and send announcements to alert users about upcoming events, such as system maintenance or the running of jobs. Announcements are displayed in the Announcements area on the application's Home page.

For more information about the Announcements area, see About the Home Page

To create an announcement:

1. Click **Tools**, and then click **Announcement**.

2. Click **Create**, and enter information such as:
   - A subject that summarizes the purpose of the announcement
   - The start date; when to send the announcement. The end date is optional.
   - The content. You may need to select an editing mode (rich text or source code) first.

Specifying Artifact Labels

The **Artifact Labels** page on the **Tools** cluster enables Service Administrators to customize artifact labels (artifact names, descriptions, and so on) based on the user's browser locale.

A few examples:

- If you create a form with a cryptic name that you don't want displayed to the user, you can define a meaningful name for the form that is displayed in the language of the user.
- If you want to create a useful instruction for an artifact that only displays in the language of the user, for example:

  "This formula calculates the number of regular employees away on a Leave of Absence."

Related Links

- Which Artifact Labels Can be Localized?
- Working With the Artifact Labels Grid
- Adding Languages and Defining Localized Artifact Labels
- Exporting and Importing Artifact Labels for Editing

Which Artifact Labels Can be Localized?

The application supports changing the language for the following artifacts:

- Card
- Cluster
- Dashboard
- Data Map
- Data Validation Rule
- Dimension
- Folder
- Form
- Member
- Menu
- Menu Item
- Navigation Flow
- Cube
- Approval Unit Hierarchy
- Report
- Rule

- Ruleset
- Smart List
- Smart List Entry
- Tab
- Task
- Task List
- Template
- User Variable
- Valid Intersection

Working With the Artifact Labels Grid

The **Artifact Labels** page displays an Excel-style spreadsheet grid that is filtered by artifact and property type.

The row axis of the grid displays the artifacts and their properties.

The column axis of the grid displays the following columns:

- **Artifact**: The type of artifact (for example, Task List or Rule)
- **Property**: The artifact's property type (for example, Name, Description, and so on)
- **Default**: Displays the artifact labels that were defined when the artifact was created.

When a language is added, a new column displays to the right of the **Default** column.

To view and filter the **Artifact Labels** grid:

1. Click **Tools**, and then click **Artifact Labels**.
2. To filter:
   a. Click  , and then select the artifacts you want to work with. For some artifacts, you can further filter by property type.
   b. Click **Apply** to close the **Filter** window and display the artifact grid filtered by artifact type and property type.

Adding Languages and Defining Localized Artifact Labels

Service Administrators can add a language for a given artifact to the **Artifact Labels** grid from a list of supported languages. You can select only one language at a time. When you add a language, a new column for that language is added to the grid to the right of the **Default** column. The cells in the language-specific column are editable.

> **Tip:**
>
> Use this method to add labels directly in the artifact labels grid. This method is ideal if you only need to add or update a few labels at a time. For bulk changes or edits on artifact labels; for example, terminology changes that affect multiple labels, use the export feature to edit in Excel, then import. See Exporting and Importing Artifact Labels for Editing.

To add a language:

1. Click **Tools**, and then click **Artifact Labels**.
2. Click  , and then select the artifacts you want to work with. For some artifacts, you can further filter by property type.
3. Click **Apply**.
4. Click **Add Language**.
5. Select from the list of supported languages.
6. In the language-specific column, enter artifact labels into the editable cells for each artifact property (Name, Description, and so on).

> **Note:**
>
> Using Ctrl+C (Copy) and Ctrl+V (Paste) are not supported in the artifact labels grid.

7. Click **Save**.

> **Note:**
>
> When you define a localized artifact label for the Default navigation flow (for example, editing the name of an icon on the Home page), your update will automatically propagate to all navigation flows. However, if you define a localized artifact label for another navigation flow that isn't the Default flow, then that update will override the label coming from the Default flow.

Exporting and Importing Artifact Labels for Editing

You can export all the artifact labels in a given language to edit them. Only artifacts that already have labels will be exported. The labels are exported in an Excel file format (XLSX). After you edit the labels, you can import them back into the application.

> **Tip:**
>
> Use this method for bulk changes or edits on artifact labels by language; for example, terminology changes that affect multiple labels. For updates to individual artifact labels, you can edit them directly in the artifact grid. See Adding Languages and Defining Localized Artifact Labels.

To export all artifact labels by language for editing and then import them:

1. Click **Tools**, and then click **Artifact Labels**.
2. Export the XLSX file containing all of the artifact labels:
   a. Click **Actions**, and then **Export**.

b. Select the target location for the export file:
   - **Local**: Saves the export file to a location on your local computer.
   - **Outbox**: Saves the export file to the server. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.

c. Choose a language.

d. Click **Export**.

3. Edit the labels in the XLSX file.

4. Import the XLSX file:

   a. Click **Actions**, and then **Import**.

   b. Select the location of the import file:
      - **Local**: Loads the import file from a location on your computer. For **Source File**, click **Browse** to select the import file on your computer for the artifact you're importing.
      - **Inbox**: Loads the import file from the server. Enter the name of the file in **Source File**. See Uploading and Downloading Files Using the Inbox/Outbox Explorer.

   c. Click **Import**.

19

Accessing More Administrative Tasks

Use the information in this chapter to perform additional administrative tasks.

Related Links

- About the Navigator Menu
- Administering Data Load Settings
- Importing Using Data Integration
- Administering Action Menus
- Administering Alias Tables
- Administering Dimensions
- Administering Forms
- Administering Rules
- Administering Rules Security
- Administering Smart Lists
- Administering Task Lists
- Setting User Preferences
- Administering Variables
- Clearing Cell Details
- Copying Data
- Administering Application Diagnostics

About the Navigator Menu

You can access additional administrative tasks from the Navigator Menu.

Clicking **Navigator** on the Home page displays a list of links that connect you to more business process functionality.

> **Note:**
>
> Some of the links are available only if you're accessing the business process from the desktop.

Administering Data Load Settings

Specify parameters to enable data to be loaded directly into an application database. Optionally, you can use advanced settings if you want to load details to child members of parent dimension members based on unique identifiers for the driver dimension.

For example, a company might load the Employee dimension member with account data for Start Date, Position, Salary Basis, and Pay Type. Because the human resource data includes placeholders for new and existing employees, the company could set up the following advanced settings:

- Data load dimension parents: New Employees, Existing Employees
- New Employees unique identifiers: Start Date, Position
- Existing Employees unique identifiers: Salary Basis, Pay Type

During data load, child members of New Employees and Existing Employees are evaluated for data updates. The unique identifiers Start Date, Position, Salary Basis, and Pay Type determine if existing data load dimension values are updated, or if new values are added: If the unique identifier's data values are the same, data is updated. If the data value is different, the next available child member is used.

To specify parameters for loading data:

1. From the Home page, click **Navigator**, and then under **Integration**, click **Data Load Settings**.
2. For **Data Load Dimension**, select the dimension (such as Employee) for which data is loaded for the application.
3. For **Driver Dimension**, click to select the dimension into which data is loaded.

   For example, if you're loading data to Employee, the driver dimension might be Account.
4. Select the members of the driver dimension.

   For example, if the driver dimension is Account, driver dimension members might include Start Date, Grade, Position, Salary Basis, and Pay Type.
5. **Optional:** To use advanced settings, complete these steps.
   a. Add a row by clicking .
   b. To the right of the new field, click and select a parent member.

      For information about selecting members, see Using the Member Selector.
   c. To the right of the parent member, under **Driver Dimension Unique Identifiers**, select members as unique identifiers. (Members selected for this field must be included in the list of selected Driver Dimension members at the top of the page.)

      Each parent member must include at least one unique identifier member. These members determine if existing data load dimension values are updated, or if new values are added.
   d. If necessary, continue adding rows by repeating the previous steps.
   e. To duplicate or delete a row, click within a row, and click or .

Importing Using Data Integration

Data Integration is the mechanism by which integration processes are performed in Oracle Fusion Cloud EPM. Designed for busy administrators and users, you can define file-based and direct integration sources, create mapping rules to translate source data into the required target format, and execute and manage the periodic data loading process. Common integration tasks are done using an easy-to-navigate interface that supports and conforms to how you work.

For a general understanding of the integration process, see Defining a Data Integration in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

Tutorials

Tutorials provide instructions with sequenced videos and documentation to help you learn a topic.

Your Goal	Learn How
Learn how to register applications, define period mappings, define category mappings, and then create and run a file-based data integration.	Loading Data Using Data Integration

Administering Action Menus

Related Topics

- Creating and Updating Action Menus
- Working with Action Menu Items
- Defining Action Menu Items

Creating and Updating Action Menus

Service Administrators can create right-click (or action) menus and associate them with forms, enabling users to click rows or columns in forms and select menu items to:

- Launch another application, URL, or business rule, with or without runtime prompts
- Move to another form
- Open Job

The context of the right-click is relayed to the next action: the POV and the Page, the member the user clicked on, the members to the left (for rows), or above (for columns).

When designing forms, use **Other Options** to select menus available for Form menu item types. As you update an application, update the appropriate menus. For example, if you delete a business rule referenced by a menu, remove it from the menu.

To create, edit, or delete action menus:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Action Menus**.
2. Perform an action:

- To create an action menu, click  , enter the menu's name, and then click **OK**.
- To edit details of an action menu, select it, and then click  . See Working with Action Menu Items.
- To delete action menus, select them, click  , and then click **OK**.

Working with Action Menu Items

The **Edit Menu** page displays menu items on the current action menu, including names, labels, required dimensions, icon, and type, such as URL, Form, Business Rule, Menu Header, Form, and Job.

To work with action menu items:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Action Menus**.
2. Select a menu, and then click  .
3. **First time only:** To add the first item to the menu, click **Add Child** and **Save**.
4. Select a menu item and:
   - To add menu items below the selected item, click **Add Child** (available for Menu Header menu types).
   - To add menu items at the same level as the selected item, click **Add Sibling**.
   - To edit menu items and define the menu item properties, click **Edit Menu Item**.
   - To delete menu items, click **Delete Menu Item**.
   - To change the order of menu items within the same level, click **Move Up** or **Move Down**. You can move multiple items.
5. Click **Save**.

   Click **Save As** to save the current selections under a new menu name.

Defining Action Menu Items

To define action menu items:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Action Menus**.
2. Select a menu and then click  .
3. Select the menu item, and then click **Edit Menu Item** or **Add Sibling**.
4. Define the menu item:

Table 19-1 Edit Menu Item Options

Item	Description
Menu Item	Enter a unique name containing only alphanumeric and underscore characters, with no special characters or spaces

Table 19-1 (Cont.) Edit Menu Item Options

Item	Description
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.
Icon	This option is not currently supported.
Type	Select the menu item type to determine available properties. **Note:** No properties are available for Menu Header. • **URL**: Create a menu that opens the specified URL. • **Form**: Create a menu that launches the 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. When launched from the web, the form opens in the web; when launched from Oracle Smart View for Office, the form opens in Smart View. Action menus referencing forms enabled as flex forms will open these forms as flex forms in Smart View and as simple forms in the web. • **Business Rule**: Create a menu that launches the selected business rule. • **Dashboard**: Create a menu that launches the selected dashboard. The cell context is retained when users right-click in the source form, and the dashboard is opened in a new dynamic tab. **Note:** The dynamic tabs feature is available only if Redwood Experience is enabled. • **Menu Header**: Create a menu under which you can create children menu items. To display a separator bar on the menu at this item, enter one hyphen as the Label. In this case, the Required Dimension list isn't available. • **Previous Form**: Create a menu that returns the user to the previous form. • **Copy Version**: Create a menu that opens Copy Version to enable administrators to copy data for the current form. **Note:** **Copy Version** is not available to Power Users and Users.

Table 19-1 (Cont.) Edit Menu Item Options

Item	Description
Required Parameters	Select a dimension and member, 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. **Note:** In Smart View, for action menus attached to forms (simple forms or flex forms), the Page and POV options are not supported.

5. Define menu item properties, which differ for menu item types:

Table 19-2 Options for Menu Item Types

Type	Options

URL	a. In **URL**, enter the complete URL to which to direct the user. For example: `http://`*server name*`/HFM/Logon/HsvLogon.asp`. URL type action menus automatically launch a new tab.
	b. Select **Use Form Context** to replace the bracketed dimension name in the URL (for example, `<Entity>` or `<Account>`) with the member name of the corresponding dimension from the form's page or POV.
	For example, to return Entity, Scenario, Version, and Account dimension's members in the URL, enter the URL and enable **Use Form Context**:
	`http://yourcompanyurl/EntDim=<Entity>&test['VERSION']=<Version>&Acc=<Account>`
	The URL will open in a new tab as:
	`http://yourcompanyurl/EntDim=<410>&test['VERSION']=<working>&Acc=<1110>`
	If **Use Form Context** is disabled, then the URL will be launched in a new tab as is without the context replacements.

Form	a. In **Form Folder**, select the folder containing the destination form. b. In **Form**, select the form.

Table 19-2 (Cont.) Options for Menu Item Types

Type	Options
Business Rule	a. In **Cube**, select the cube for which the business rule is available. b. In **Business Rules**, select the business rule to launch. c. In **View Type**, select how to display runtime prompt pages: • **Classic View**: Use the default application view • **Streamline View**: Display each runtime prompt on a different line d. **Optional**: In **Window Title**, enter a title to display instead of Runtime Prompts. e. **Optional**: In **OK Button Label**, enter the text to display for the OK button. f. **Optional**: In **Cancel Button Label**, enter the text to display for the Cancel button. g. **Optional**: In **Launch Confirmation Message**, enter text to display when the business rule is invoked, but before it's launched. This option enables Service Administrators to provide meaningful messages to users about the consequences of launching business rules.
Dashboard	In **Dashboard**, select the dashboard.
Previous Form	Enter the name of the menu item that will return the user to the previous form.

6. Click **Save**.

Administering Alias Tables

Related Topics

- About Aliases
- About Alias Tables
- Working with Alias Tables
- Specifying a Default Alias Table and Setting Member and Alias Display Options

About Aliases

You can assign alternate names, or aliases, to Account, Currency, Entity, Scenario, Period, Version, Years, and user-defined dimension members. The application allows up to 30 aliases per dimension member, including the default alias. Aliases can have the same name within an alias table and across alias tables.

Aliases can also have:

- The same name as a member
- The same alias on members that are parent and child
- The same name for members from different dimensions or from the same dimension

> **Note:**
>
> - You can't have the same alias for two members that are siblings because there would be no way to uniquely identify the member. This rule is enforced by the application for base members, but not for shared members.
>
> This rule isn't enforced for shared members because you can't directly set aliases for shared members; shared member aliases are inherited from the alias of their base member. It's possible to create an alternative hierarchy where you can have two shared members that have the same alias and are siblings. However, this situation is discouraged if you want to reference these members by their aliases from an adhoc grid because, when you type the alias into the grid and submit it to the application, the application can't uniquely resolve this member and it will return an error. If you're using the alias for display purposes only then there will be no issue, but this design is discouraged because, visually, there is no way to differentiate between these two members.
> - Although aliases can have the same name as a member, be careful not to set the alias of member1 to be the same name as member2. This can lead to unintended results and will cause confusion on the form grids.
> - Member names must be unique so that they can be used in rules and form designs.

About Alias Tables

You can create and update alias tables, and set a default alias table for the application. Follow naming conventions in Naming Restrictions.

Multiple alias tables support, for example, these language combinations:

- English, French, German, Spanish, and Italian
- Japanese and English
- Korean and English
- Turkish and English

> **Note:**
>
> Alias table support isn't restricted to these language combinations.

You can set alias tables to display members in an application. Users can set alias tables in preferences.

Working with Alias Tables

You can add, edit, rename, and delete alias tables or clear alias table values. You can also copy alias table content from one table to another.

To work with alias tables:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Alias Tables**.
2. Choose a task:
   - If adding an alias table, click , and then in **Add - Alias Table**, enter a name.
   - If editing or renaming alias tables, select the alias table, then click , and then for **Edit - Alias Table**, enter a name.
   - If deleting alias tables, select the alias table, and then click **Delete**.

     > **Note:**
     >
     > You can't delete the Default alias table.

   - If clearing the values in an alias table, select the alias table to clear, then click **Clear Values**.

     > **Note:**
     >
     > Clearing the alias table removes the content of the table, but doesn't remove the table.

   - If copying alias table content, select the alias table, click **Copy**, then select the destination alias table, and then click **Copy**.

     > **Note:**
     >
     > The destination alias table must exist. Copying doesn't create tables.

3. Click **OK**.

Specifying a Default Alias Table and Setting Member and Alias Display Options

If you create alias tables with aliases for Account, Currency, Entity, Scenario, Period, Version, Years, and user-defined dimensions and members, you can select a default alias table for the application. Users can set preferences for which set of aliases (stored in an alias table) to use for displaying member and dimension names.

To select the application's default alias table:

1. From the Home page, click **Application**, and then click **Settings**.
2. For **Alias Table**, select an alias table.
3. For **Display Member Label as**, select the option that enables the kind of member data to be displayed in the member selector throughout the application:
   - **Default**: The data determined by the form, grid, or dimension settings

- **Member Name** : Just member names
- **Alias**: Just member aliases, if defined
- **Member Name:Alias**: Names followed by aliases, if defined
- **Alias:Member Name**: Alias, if defined, followed by the names

4. Click **Save** or **Reset**.

Administering Dimensions

This section documents how to edit dimensions using the Classic Dimension Editor which is accessed by using the **Dimensions** link in the Navigator menu. In the 17.05 (May 2017) update, we released the Simplified Dimension Editor.

For detailed information about using the Simplified Dimension Editor, see Editing Dimensions in the Simplified Dimension Editor.

Related Links

About Dimensions

Working with Dimension Hierarchies

About Custom Dimensions, Entities, Accounts, Periods, and Cubes

Adding or Editing User-Defined Custom Dimensions

Working with Members

Working with Attributes

Working with Attribute Values

Customizing Calendars

Setting up Dynamic Time Series Members

Working with UDAs

Working with Member Formulas

About Dimensions

Dimensions categorize data values.

FreeForm enables you to create an application with the cubes and dimensions of your choice without being constrained by the cube and dimension limitations imposed by standard applications. You can add up to 12 total cubes with any combination of aggregate storage and block storage. See Understanding FreeForm.

Members are components of dimensions.

For complete information about dimensions, see Dimension Overview.

Working with Dimension Hierarchies

Related Topics

- Filtering the Dimension View by Cube
- Sorting Members

- Viewing a Member's Ancestors
- Determining Where Members Are Used in an Application

Filtering the Dimension View by Cube

You can filter the dimension view by cube. When you select a cube, only dimensions used in that cube are displayed on the **Dimensions** page.

To filter the dimension view by cube:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. For **Cube**, select the cube.

The application displays only the dimensions used in the selected cube.

Sorting Members

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

To sort members:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select the dimension for the members.
3. On **Dimensions**, select the members whose children or descendants you want to sort.
4. For **Sort**, select children or descendants.

   Sorting by children affects only members in the level immediately below the selected member. Sorting by descendants affects all descendants of the selected member.
5. Click to sort by ascending order or to sort by descending order.
6. Click **OK**.

   The next time you create or refresh the database, the outline is generated with members in the order that is displayed.

Viewing a Member's Ancestors

To view a member's ancestors:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select a dimension.
3. Select the member in the dimension hierarchy.
4. Click .
5. Click **OK**.

Determining Where Members Are Used in an Application

To view where members are used in an application:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension whose member's usage you want to view.
3. Click .

About Custom Dimensions, Entities, Accounts, Periods, and Cubes

See the following topics for more information:

- About Custom Dimensions
- About Entities
- About Accounts
- Accounts, Entities, Periods, and Cubes

Adding or Editing User-Defined Custom Dimensions

User-defined custom dimensions must conform to guidelines listed in Naming Restrictions.

Table 19-3 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 an alternate name for the dimension. See About Aliases.
Description	**Optional**: Enter a description.
Valid for Cubes	Select cubes for which the dimension is valid. Clearing this option makes all members of the dimension invalid for the deselected cube.
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**.

To add or change user-defined dimensions:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Click or select an existing dimension and click .
3. Specify any of the properties listed above.
4. Click **Save**.
5. Click **OK**.

   Click **Refresh** to revert to the previous values and keep the page open.

- Setting Dimension Properties
- Setting Dimension Density and Order
- Setting the Evaluation Order

Setting Dimension Properties

Dimension properties must conform to guidelines listed in Naming Restrictions.

Table 19-4 Dimensions Properties

Property	Value
Dimension	Enter a name that is unique across all dimensions.
Description	**Optional**: Enter a description.
Alias Table and **Alias**	**Optional**: Select an alias table. Enter an alternate name for the dimension. See Administering Alias Tables.
Valid for Cubes	Select cubes for which the dimension is valid. Clearing this option makes all members of the dimension invalid for the deselected cube.
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.
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**.
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.

Setting Dimension Density and Order

The **Performance Settings** tab enables you to set dimensions as sparse or dense and set their order of precedence.

To manage performance settings:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the **Performance Settings** tab.
3. For each dimension, set its **Density** as **Dense** or **Sparse**.

   See About Sparse and Dense Dimensions.

   > **Note:**
   >
   > The **Density** column is hidden for applications that contain only ASO cubes. If an application contains a BSO cube or a BSO and an ASO cube, the **Density** column displays for both cubes.

4. Set the order of precedence by selecting a dimension and clicking or next to the **Position** column heading.

Setting the Evaluation Order

The **Evaluation Order** tab enables you to specify which data type prevails when a data intersection has conflicting data types.

To set evaluation order:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select **Evaluation Order**, and then select the cube.
3. From **Available Dimensions**, select dimensions and move them to **Selected Dimensions**:
   - moves selected dimensions
   - moves all dimensions
   - removes selected dimensions
   - removes all dimensions

   You need select only dimensions whose members have specific data types (that is, their data type isn't "Unspecified"). The data type "Unspecified" doesn't conflict with another data type.
4. If you select multiple dimensions, set the order of precedence by clicking or .
5. Click **Save**.

Working with Members

You can assign access rights to members, rearrange the dimension member hierarchy, share members of the Entity, Account, and user-defined custom dimensions, and enable creation of dynamic members "on-the-fly."

- Finding Dimension Members
- About Assigning Access to Members
- Adding or Editing Members
- Deleting Members
- Deleting Parent Members
- Working with Shared Members
- Creating Shared Members
- About Dynamic Members

Finding Dimension Members

To find dimension members in dimension hierarchies:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select the dimension for the member.

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

About Assigning Access to Members

Service Administrators can assign permissions to members.

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

For complete information, see Assigning Access to Dimension Members. To enable access to members, see Editing Dimension Properties.

Adding or Editing Members

Members must conform to guidelines listed in Naming Restrictions. Shared members must be consistent with Working with Shared Members.

Table 19-5 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 in **Alias**. See Administering Alias Tables.
For Account members only: **Account Type**	Select **Expense**, **Revenue**, **Asset**, **Liability**, **Equity**, or **Saved Assumption**. For descriptions, see Account Types.
For Account members only: **Variance Reporting**	If the account type is **Saved Assumption**, select **Expense** or **Non-Expense**. Designate the saved assumption as a revenue, asset, liability, or equity account.
For Account members only: **Time Balance**	Select **Flow**, **First**, **Balance**, **Average**, **Fill**, **Weighted Average - Actual_Actual**, or **Weighted Average - Actual_365**. For descriptions, see Time Balance Property.
For Account members only: **Skip**	If the account type is **Asset**, **Equity**, or **Liability**, select **None**, **Missing**, **Zeros**, or **Missing and Zeros**. For descriptions, see Setting Account Calculations for Zeros and Missing Values.
For Account members only: **Exchange Rate Type**	Select **Average**, **Ending**, or **Historical**. For descriptions, see Data Type.
For Account members only: **Data Type**	Select **Percentage**, **Date**, or **Text**. For descriptions, see Data Type.
For Account members only: **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.

Table 19-5 (Cont.) Member Properties

Property	Value
Hierarchy Type	Hierarchy Type is available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be stored. **Note:** For members with a stored hierarchy type, the only valid cube aggregation options are Addition or Ignore. In a stored hierarchy, the first member must be set to Addition. For members with a dynamic hierarchy type, all cube aggregation options are valid. Stored hierarchy members that are not children of Label Only members must have Addition set as the consolidation operator. Children of Label Only members can be set to Ignore.
Data Storage	Select a data storage property. The default is Never Share for new custom dimension members (except root members).
Two Pass Calculation	Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with Dynamic Calc or Dynamic Calc and Store properties.

Table 19-5 (Cont.) Member Properties

Property	Value
Plan Type	Select the plan types (or cubes) for which the member is valid. **Note:** A member can belong to both aggregate storage and block storage cubes. Select an aggregation option for each selected cube. You can select a source cube only if multiple cubes are valid for the member. Only cubes and aggregation options for which the member's parent is valid are available. If the parent isn't valid for a cube or aggregation option, neither is the child member. Deselecting a cube for an account or entity parent member deselects it for all descendents of that parent. For members with a stored hierarchy type, the only valid aggregation options are Addition or Ignore. See Aggregation Options. **Caution:** Deselecting a cube for dimension members after data is entered into an application may result in loss of data when an application is refreshed. For account members, data is lost if the deselected cube is the source cube. Members of a custom dimension and a Period dimension can set usage by cube, similar to the Account and Entity dimensions.
For Account members only: **Source Cube**	Select the source cube for the member. A shared member is a pointer to the base member and isn't stored; this is disabled for shared members. The source cube of a shared Account member matches the source cube of the base member, even though Source Plan field is unavailable because it doesn't apply to shared members.
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).
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.

Table 19-5 (Cont.) Member Properties

Property	Value
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. (A Service 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 a Service Administrator changes these settings, they affect only future dynamic members; they don't retroactively affect dynamic members.

To add or edit members:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension.
3. Perform one action:
   - To add a child member, select the parent level of the dimension hierarchy to which to add a member and click .
   - To add a sibling, select the level of the dimension hierarchy to which to add a sibling and click .
   - To edit a member, select that member from the dimension hierarchy and click .

   Note:

   To add an All Years parent member that includes all members of the Years dimension, select the Years dimension and then click . The All Years parent member enables users to view the accumulated data across multiple years, for example, a project's total cost up to its end date. The All Years member doesn't include the No Year member, if one is defined for the application.
4. On **Member Properties**, set or change member properties described in Table 1.

   If you don't see the new member on the page, click **Next**.

5. Click **Save** to save information to the relational database and see changes in the dimension hierarchy.
6. Refresh the database so edited members are visible to users entering data.
7. After creating a dimension member, you typically complete these tasks:
   - Assign access. See Assigning Access to Dimension Members.
   - Specify attributes.

Deleting Members

Each data value is identified by a set of dimension member values and a cube. Deleting dimension members or deselecting the cube results in data loss when refreshing an application.

Before deleting members, understand where in the application they are used (in which forms, exchange rates, and so on) by using **Show Usage**. See Determining Where Members Are Used in an Application.

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

To delete members:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension whose member you want to delete.
3. From the dimension hierarchy, select the entity member to delete.
4. Click .

   Deleting a base member also deletes its shared members.
5. Click **Yes**.
6. Update and validate business rules and reports.

Deleting Parent Members

Data values are identified by a set of dimension member values and a cube. Deleting dimension members or deselecting the cube results in data loss when refreshing the application.

To delete a parent member and all its descendants from the dimension hierarchy:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension whose member and descendants you want to delete.
3. Select the member whose branch to delete.
4. Click .
5. Click **Yes**.

Working with Shared Members

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

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

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

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

Shared members are displayed similarly to base members in the dimension hierarchy for member selection in Oracle Smart View for Office.

Creating Shared Members

You create shared members the same way as other members, with these differences:

- The base member can't be the parent of the shared member.
- You can't add a shared member as a sibling to the base member.
- You must give the shared member the same name as its base member. It can have a different description.
- You must select Shared as the Data Storage for the shared member.

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." A Service Administrator enables end users to create dynamic members under a parent member and must refresh the database to create the required placeholders in Essbase. In business rules with runtime prompts, users can then create members by entering desired member names in the runtime prompts. Subsequent database refreshes will rename used dynamic children to the names specified by the end users and recreate the required placeholders in Essbase. For information about working with business rules and dynamic members, see *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*.

If a parent member is enabled for adding dynamic children, users can create new members by entering their name in the runtime prompt.

Enabling Parent Members for Adding Dynamic Children

To enable a parent member for adding dynamic children:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Dimensions**.

2. Edit the parent member and select the option **Enable for Dynamic Children** (see Adding or Editing Members).
   - **Optional**: Set the member property **Number of Possible Dynamic Children** (the default is 10). This setting determines the number of placeholders that are created for dynamically adding or loading members under the parent. If all placeholders are used, subsequently added children are added as normal members and can't be used until the database is refreshed.
   - **Optional**: Set the member property **Access Granted to Member Creator** (the default is Inherit).
3. Refresh the database to create the placeholders for dynamic members in the databases for the cubes in which the members are used.
4. In Calculation Manager:
   a. Create a business rule with a runtime prompt (of the Variable type Member). Under the **Dynamic Member Parent** column, use the **Member Selector** to select the parent member that you enabled for dynamic children in the application.
   b. Select the option **Create dynamic members**.
   c. Deploy the business rule.

      For information about working with business rules, see *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*.

> **Note:**
>
> - In Calculation Manager, selecting both the **Create dynamic members** and the **Delete dynamic members** options enables temporary dynamic members to be created for calculations, and those temporary dynamic members will be deleted after the business rule completes.
> - Selecting the **Delete dynamic members** option alone presents you with the **Member Selector** in the runtime prompt so you can delete any member that you created dynamically under the parent (if you have write access to that member). This enables you 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.

Results

If all the above conditions are met, when users run the business rule with the runtime prompt, they enter the name of the dynamic member, and then click **Launch**. If the business rule runs successfully, the member is created in the dimension hierarchy under the dynamic member's parent.

Child members that you import under parent members enabled for dynamic children are added as dynamic child members if there are dynamic member placeholders in Essbase. After the placeholders are full, any remaining children are added as normal members, and can't be used until the database is refreshed.

> **Note:**
>
> If you load a parent member that is enabled for dynamic children and its child members during the same import, the child members are loaded as normal members. This is because the database must be refreshed to create the placeholders in Essbase.

Considerations

When you create a dynamic member and data is captured against that member in multiple cubes (either through direct data entry, through calculations, data load, or data maps/Smart Push), then you must ensure that you clear the data from each of these areas before you remove the member using the **Delete dynamic members** operation. Performing the **Delete dynamic members** operation does not remove the data; the member is removed but the cube(s) will retain the data without the member showing up in the interface, and any rollup on the cube will reflect incorrect totals.

Working with Attributes

Use attributes to group members using the same criterion. You can assign attributes to sparse dimensions only. You can't assign attributes to label-only members. Attribute dimensions don't have aggregation properties because parents are dynamically calculated.

The Account dimension is usually defined as dense, so you can't assign attributes to it unless it's changed to sparse for all cubes. If you change a dimension from sparse to dense, all attributes and attribute values for that dimension are automatically deleted.

Attributes can have data types of text, date, Boolean, and numeric, as described in Understanding Attribute Data Types. Attribute names must conform to guidelines listed in Naming Restrictions. When attributes are defined, you can use the **Member Selection** dialog box to select attribute functions, such as **Equal** and **GreaterOrEqual**.

> **Note:**
>
> This topic documents how to work with attributes using the Classic Dimension Editor. For information about using the Simplified Dimension Editor to work with attributes, see Working with Attributes.

To create and change attributes, attribute values, and aliases:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select a sparse dimension for which to define an attribute, attribute value, or alias.

   > **Note:**
   >
   > Only sparse dimensions can contain attributes.

3. Click .

> **Note:**
>
> If the dimension isn't sparse, **Custom Attributes** isn't available.

4. Select options.

   - To create attributes, click  . Type an attribute name, and select a data type: **Text**, **Date**, **Boolean**, or **Numeric**.

     > **Note:**
     >
     > – You can't modify the type after the attribute is created.
     > – Before working with date attributes, you must select the **Attribute Dimension Date Format** in **Application Settings** and save it.

     See Understanding Attribute Data Types.

   - To modify attributes, click  , and update the attribute name.
   - To set aliases for attributes, select an attribute and an attribute value, click  . Select an alias table, type an alias name, and click **Save**.

5. Click **Close**.

   When you click **Close**, the hierarchy is validated and an error displays if issues are detected. For example, date attribute values must be entered in the correct format, and numeric and date attribute dimensions must have at least one attribute value defined.

6. Update and validate business rules and reports.

Related topics:

- Understanding Attribute Data Types
- Deleting Attributes

> **Note:**
>
> Instead of explicitly filtering by an attribute (like Red, for instance), you can create a user variable for the attribute dimension, and then use the user variable as the filter. Then you can enable the user variable as a dynamic user variable which would allow users to change the value of the filter at runtime. This is a useful technique that allows for dynamic filtering. See Managing User Variables.

Table 19-6 Tutorials

Your Goal	Learn How
Learn how to report on data with attribute hierarchies. You can view and report on data organized by attribute on the web on forms and dashboards, with MR reports, or in Oracle Smart View for Office with ad hoc analysis.	Reporting on Data with Attribute Hierarchies

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 the business process, 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**.

> **Caution:**
>
> Before you can work with date attributes, you must toggle the **Attribute Dimension Date Format** selection in **Application Settings** and save it. From the Home page, click **Application**, and then click **Settings**. In **Attribute Dimension Date Format**, select a date format (**MM-dd-yyyy** or **dd-MM-yyyy**), and then click **Save**. After you select and save the date format, you must manually change any existing attribute date values to the supported format. If any date attributes in your application use an unsupported format, the system will provide you with a list of dimension attribute values that you will need to fix.
>
> Also, if the **Attribute Dimension Date Format** setting is changed in the application settings, you must change any date attributes in the original format to the new format before you can add or save a new attribute.

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.

> **Note:**
>
> This topic documents how to delete attributes using the Classic Dimension Editor which is accessed by using the **Dimensions** link in the Navigator menu. In a previous update, we released the Simplified Dimension Editor.
>
> For information about using the Simplified Dimension Editor to delete attributes, see Deleting Attributes.

To delete attributes:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Dimensions**.
2. Select the sparse dimension for which to delete an attribute, and click  .
3. Select the attribute to delete.
4. Above the **Attributes** column, click  .
5. Click **OK**.
6. Update and validate business rules and reports.

Working with Attribute Values

Attribute values provide users with another way of selecting dimensions members when using forms. Data values for attribute values are dynamically calculated but not stored.

> **Note:**
>
> This topic documents how to work with attribute values using the Classic Dimension Editor. For information about using the Simplified Dimension Editor to work with attribute values, see Working with Attribute Values.

- Creating Attribute Values
- Assigning Attribute Values to Members
- Editing and Deleting Attribute Values

Creating Attribute Values

You can define attribute values for sparse dimensions, which are typically the Entity and user-defined custom dimensions. After you define an attribute value for a dimension, you can assign it to members of that dimension.

To create attribute values:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the sparse dimension for which to create an attribute value.
3. Select the top level in the dimension hierarchy.
4. Click .
5. On the **Manage Attributes and Values** page, select the attribute for which to specify a value.
6. Above the **Attribute Values** column, click . If the options are available, you can click or .
7. Enter a name and choose a data type.
8. Click **Save**.

Assigning Attribute Values to Members

You can assign attribute values members of a dimension that are defined as sparse for all cubes. Attribute values must be assigned to the same-level sparse dimension members. Otherwise, errors display during refresh.

To assign attribute values to members:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the sparse dimension for whose member you want to assign an attribute value.
3. In the **Dimension** hierarchy, select a member to which to assign an attribute value.
4. Click .
5. For members assigned attribute values, click to change a member's attribute value.
6. Select attribute values to assign to the member.
7. Perform an action:
   - To assign the value to the selected member, click .
   - To remove a value from the selected member, select the value to remove and click .
   - To remove all values from the selected member, click .
8. Click **Save**.

Editing and Deleting Attribute Values

> **Note:**
>
> When you delete an attribute value, it's removed from custom dimension members to which it's assigned.

To edit or delete attribute values:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Dimensions**.
2. Select the sparse dimension containing the attribute value you want to modify or delete.
3. Select the top level in the dimension hierarchy.
4. Click  .
5. For **Attributes**, select the attribute containing the value to modify or delete, and then select the attribute value.
6. Above **Attribute Values**, click  or  .
7. If editing, enter a name. If deleting, confirm the deletion.
8. Click **Save**.
9. Update and validate business rules and reports.

Customizing Calendars

Use the Period dimension to work with the yearly calendar rollup structure. When creating the application, the Service Administrator specifies the base time periods that span the application database. Use the Years dimension to add years to the calendar.

Related Topics

- Defining How Calendars Roll Up
- Creating and Editing Summary Time Periods
- Deleting Summary Time Periods
- Working with the Years Dimension
- Adding Years to the Calendar
- Editing Year Information
- Renaming Time Periods
- Assigning Aliases to Summary Time Periods
- Editing the BegBalance Member

Defining How Calendars Roll Up

Table 19-7 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 can't change the base time period or reduce the number of years in the calendar. Service Administrators can change the names, descriptions, aliases, and ranges of the summary time periods in the hierarchy.

Creating and Editing Summary Time Periods

You can change such aspects as name, description, alias, starting period, and ending period. However, you can't change the order of base time periods or skip base time periods. The range can't extend beyond the current fiscal year.

You must work from the top of the hierarchy to the bottom when creating summary time periods. (Otherwise, the application views the rollup structure as asymmetrical and you can't continue.) The summary time period displays in the hierarchy as a parent of the selected item. To enforce a balanced hierarchy, all base members must be the same number of levels from the root.

To create or edit summary time periods:

1. From the Home page, click **Navigator**  , and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Period**.
3. Perform one action:
   - To add a time period, select the time period in the dimension hierarchy to which you want to add a child or sibling, and then click  or  .
   - To edit a time period, select the time period and click  .
4. For **Name**, enter or change the name for the summary time period.
5. **Optional**: For **Description**, enter a description.
6. **Optional**: For **Alias Table** and **Alias**, select an alias table to use. Enter an alias name.

   The default table is used if you don't select one.
7. For **Start Period**, select the starting period.

   The range can't extend beyond the current fiscal year. For summary time periods, **Start Period** displays the first child, or all children except the first child of the sibling above it.
8. For **End Period**, select the ending period.

   For summary time periods, **End Period** displays the last child, or all children from the Start Period through the next sibling's children, except the last child.
9. Click **Save**.

Deleting Summary Time Periods

When you remove a summary time period from the hierarchy, its children are moved into another summary time period:

- If you delete the first summary time period, children are moved into the next sibling of the summary time period.
- If you delete the last summary time period, children are moved into the previous sibling of the summary time period.
- If you delete a summary time period from the middle of a hierarchy, children are moved into the previous sibling of the summary time period.

To delete summary time periods:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Period**.
3. In the dimension hierarchy, select the summary time period to delete.

   You can't delete base time periods.
4. Click .
5. Click **OK**.

Working with the Years Dimension

Use the Years dimension to work with calendar years.

Table 19-8 Years Tasks

Task	See Topic
• Add years before the Start year or after the End year of the calendar. • Add an All Years parent member that includes all members of the Years dimension (except No Year, if that member exists).	Adding Members
Add or update the description and alias for a year.	Editing Members

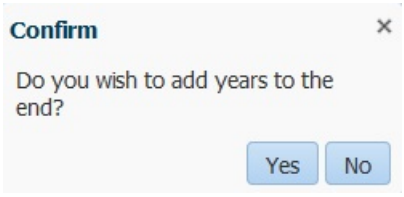
Adding Years to the Calendar

You can add years to the beginning or end of the calendar, but you can't reduce the number of calendar years without creating a database.

To add years to the calendar:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Years**.
3. Click .
4. In **Add Years**, enter the number of years to add to the calendar.
5. Click **OK**.

   The confirmation dialog asks you to confirm the following:



   - To add years after the End year, click **Yes**.
   - To add years before the Start year, click **No**.

> **Note:**
>
> To add an All Years parent member that includes all members of the Years dimension, click . The All Years parent member enables users to view the accumulated data across multiple years, for example, a project's total cost up to its end date. (This parent member doesn't include the No Year member, if one exists.)

Editing Year Information

You can add or update the description and alias for a member of the Years dimension.

To edit a member of the Years dimension:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Years**.
3. Select the year you want to edit, and then click .
4. Update the information for that year, and then click **Save**.

Renaming Time Periods

You can rename root-level, base time periods, and user-defined summary time periods.

To rename time periods:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Period**.
3. Select the time period you want to rename, and then click .
4. Rename the time period, and then click **Save**.

Assigning Aliases to Summary Time Periods

You can assign and change aliases to base time periods and summary time periods.

To assign or change the alias:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Period**.
3. Select the summary time period.
4. Click .
5. For **Alias Table**, select the alias table to use.
6. For **Alias**, enter an alias name.
7. Click **Save**.

Editing the BegBalance Member

You can edit the BegBalance member of the Period dimension. As the first time period in the application, the BegBalance member is useful for entering beginning data when you start a new application, fiscal year, or calendar year. You can rename and describe BegBalance and give it an alias.

To edit the BegBalance member:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Period**.
3. Select the first member, **BegBalance**.
4. Click .
5. For **Edit Period**:
   - Enter a name.
   - Enter a description.
   - Select an alias table to use for the BegBalance member, and enter an alias.
6. Click **Save**.

Setting up Dynamic Time Series Members

You can use Dynamic Time Series members to create reports that show period-to-date data, such as quarter-to-date expenses. Dynamic Time Series members are created automatically during application creation, and can be used with members of the Period dimension. To set up Dynamic Time Series, you enable a predefined Dynamic Time Series member and associate it with a generation number (and, optionally, an alias table and alias name). For example, to calculate quarter-to-date values, you can enable the Q-T-D (quarter-to-date) member and associate it with generation number 2. You can then use the Q-T-D Dynamic Time Series member to calculate monthly values up to the current month in the quarter.

> **Note:**
>
> Dynamic Time Series isn't supported for the Period dimension in an aggregate storage application.

The Dynamic Time Series members provide up to eight levels of period-to-date reporting. Your data and database outline determine which members you can use. For example, if the database contains hourly, daily, weekly, monthly, quarterly, and yearly data, you can report D-T-D (day-to date), W-T-D (week-to-date), M-T-D (month-to-date), Q-T-D, and Y-T-D information. If the database contains monthly data for the past 5 years, you can report Y-T-D and H-T-D (history-to-date) information, up to a specific year. If the database tracks data for seasonal time periods, you can report P-T-D or S-T-D (season-to-date) information.

Oracle recommends that you avoid assigning time balance properties (such as First and Average) to members set for dynamic calculations if you plan to use the members in Dynamic Time Series calculations. Doing so may retrieve incorrect values for parent members in your accounts dimension.

To set up Dynamic Time Series members:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Dimensions**.
2. Select the Period dimension, and then click .
3. Select **Enabled** for the DTS series to use:
   - Y-T-D: Year-to-date
   - H-T-D: History-to-date
   - S-T-D: Season-to-date
   - Q-T-D: Quarter-to-date
   - P-T-D: Period-to-date
   - M-T-D: Month-to-date
   - W-T-D: Week-to-date
   - D-T-D: Day-to-date
4. Select a generation.

   The number of generations displayed depends on the number of generations in the time dimension. You can't associate Dynamic Time Series members with the highest generation (the dimension root).
5. **Optional**: Add an alias name. In the **Edit Alias Name** column, click , enter an alias name, and then click **OK**.
6. Click **Save**.

Additional Supported Application Features

Additional features are supported for application dimensions. You can add children and siblings to Scenario, Version, and Period dimensions, and you can use the cut, paste, expand, and collapse features to work with their dimension hierarchies (see Working with Dimension Hierarchies). You can also use shared members for these dimensions, and can set two pass calculations at the root level for all dimensions. For example, you can:

Table 19-9 Features

Feature	More Information
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 and Editing the BegBalance Member.
Turn on two pass calculation at the root level, for example, for Account.	See Adding or Editing Members.

> **Caution:**
>
> Two pass calculation is ignored on any non-Account member not set to Dynamic Calc.

Table 19-9 (Cont.) Features

Feature	More Information
For attributes, create hierarchies and assign aliases.	See Working with Attributes.

Considerations for Alternate Hierarchies in Period Dimensions

If you create an alternate hierarchy in the Period dimension, the alternate hierarchy must follow the YearTotal member in the outline.

Working with UDAs

You can use user-defined attributes (UDAs), descriptive words or phrases, within calc scripts, member formulas, reports, and forms. 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 business process and create the member formula with a UDA. The syntax for this UDA is: `(UDAs: HSP_UDF)`.
- If you use the `@XREF` function to look up a data value in another database to calculate a value from the current database, you can add the `HSP_NOLINK` UDA to members to prevent the `@XREF` function from being created on all cubes that are not the source cube selected for that member.
- For a Product dimension with several product members, you can create a UDA called New Products and assign this UDA to the new products in the Product dimension hierarchy. Then you can base certain calculations on the designation New Products.
- When designing forms, you can use a UDA to select members for forms based on a common attribute. When you create forms with UDAs, any members that are assigned to the UDA are dynamically added to the form. For example, if you create a UDA called New Products and assign this UDA to the new products in the Product dimension hierarchy, the form will automatically display the new products at runtime. When selected in the form designer, a UDA is preceded by `UDA`; for example, `UDA(New Products)`.

UDAs are specific to dimensions. For example, creating a UDA for an Account member makes it available for non-shared Account members. Deleting it removes it for all Account members. To make UDAs available for multiple dimensions, create the same UDA for multiple dimensions. For example, create a UDA named New for Account and Entity dimensions to make the UDA named New available for Account and Entity members.

To work with UDAs:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension for whose members to associate the UDA.
3. From the dimension hierarchy, select a member and click .
4. Select **UDA**.
5. Select a task:
   - To create a UDA, click , enter a name, and then click **Save**.

> **Note:**
>
> Use no more than 60 characters when naming UDAs.

- To modify a UDA, click  , change the name, and then click **Save**.
- To delete a UDA, select the UDA, click  , and then confirm deletion.

> **Note:**
>
> Deleting the UDA removes it for the whole dimension. If you delete UDAs, you must update all member formulas, calc scripts, and reports that reference them.

- To clear UDA selections, click  .

6. To add or remove UDAs for the member, use the arrows to move UDAs to and from the **Selected UDA** panel.
7. Click **Save**.

Working with Member Formulas

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

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

To define member formulas:

1. Click the **Navigator** icon  , and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension for whose member to add or change a formula.
3. Select the member and click **Edit**.
4. Select the **Member Formula** tab.
5. Select options for the following fields:
   - **Cube**

> **Note:**
>
> With the exception of Account formulas, a formula entered for the default cube will be applied to all cubes unless it's overridden by a different formula entered for a specific cube.
>
> Also, a default formula on Account members will only be transferred to Oracle Essbase for the source cube and not for any of the other cubes. A default formula won't be transferred to any aggregate storage cubes.

- **Data Storage**: Select a data storage option. The default is **Store**.

> **Note:**
>
> The cube specific data storage field will not display the **Shared** or **Label Only** options. This is because a member can't be set to Shared or Label Only in one cube and not another.

- **Solve Order**: 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.

> **Note:**
>
> Solve order is available for aggregate storage cubes and also for block storage cubes that are enabled for Hybrid. Solve order for block storage cubes is only editable using the Simplified dimension editor. See Accessing Edit Member Properties.

6. In the text box, define formulas for the member.
7. **Optional**: To check the validity of the member formula, click **Validate**.
8. Click **Save**.

   Before you click **Save**, clicking **Reset** restores the previous member formula information.

Viewing Details of Formula Validation

To view details of the member formula validation:

1. On **Member Formula**, click **Validate**.
2. Click **Save**.

Working with Formula Expressions

In the text box on the **Member Formula** tab, you can include predefined formula expressions in member formulas, and test them with the **Validate** button. You can also load them.

You can update the dimension outline without updating the business rules and calc scripts that depend on the outline. Calculations become more independent of specifics in the outline. You

can use Smart Lists as objects in calculations. Performance isn't decreased when you use formula expressions because they are run only when you refresh the database.

To use a formula expression in a member formula:

1. Click the **Navigator** icon  , and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension with which you want to work.
3. Select the member and click **Edit**.
4. Select **Member Formula**.
5. Select options for the following fields:
   - **Cube**

     > **Note:**
     >
     > With the exception of Account formulas, a formula entered for the default cube will be applied to all cubes unless it's overridden by a different formula entered for a specific cube.
     >
     > Also, a default formula on Account members will only be transferred to Oracle Essbase for the source cube and not for any of the other cubes. A default formula won't be transferred to any aggregate storage cubes.

   - **Data Storage**: Select a data storage option. The default is **Store**.

     > **Note:**
     >
     > The cube specific data storage field will not display the **Shared** or **Label Only** options. This is because a member can't be set to Shared or Label Only in one cube and not another.

   - **Solve Order**: 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.

     > **Note:**
     >
     > Solve order is available for aggregate storage cubes and also for block storage cubes that are enabled for Hybrid. Solve order for block storage cubes is only editable using the Simplified dimension editor. See Accessing Edit Member Properties.

6. In the text box, define formulas for the member.

   You can include business process formula expressions and Essbase native formulas in the member formula.

   The business process provides predefined formula expressions that you can use in member formulas. You can't edit or create your own formula expressions.

7. **Optional**: To check the validity of the member formula, click **Validate**.
8. **Optional**: If there are errors in the member formula, click **Show Details** to view a description.
9. **Optional**: Click **Reset** to restore the previous member formula if you don't want to save the changes you made to the member formula.
10. Click **Save**.

Syntax

Member formula expressions support functions and variables. Follow these syntax rules for functions and variables when you create formula expressions:

- Enclose variables or properties with square brackets, [ ]. If you omit square brackets, the variable is treated as a native variable.
- Enclose member names with quotation marks.
- Characters in variables are case-insensitive, but can't have extra spaces or characters such as underscore (_).
- You can include subcalls to other functions within an expression.
- Don't enter text where a number is required.
- The order of the outline is important in a member formula. For example, don't reference a value that has not been calculated yet.

Including Smart List Values as Variables

You can include a Smart List as a variable in a formula expression, such as the formula expression, "Status"= [Status.Departed].

"Status" is the member name, Status is the Smart List name, and Departed is a Smart List entry. If the Smart List ID for Departed is 2, Status.Departed is replaced with a 2 in the member formula (the application treats Smart Lists as numbers). If the Smart List ID for Departed is 2, 2 is put in the calculation and 2 is stored in the database.

Write Smart Lists in this format: [SmartListName.SmartListEntry]

Formula Expressions

Formula expressions can include these predefined variables and functions.

Table 19-10 Variables in Formula Expressions

Variable	Description
NumberOfPeriodsInYear	Returns the number of time periods in the year
NumberOfYears	Returns the number of years in the application

Table 19-11 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 by adding the default prefix of "No" to each dimension name (except Currency, Period and Year), followed by the specified account. For example, in an application with the following dimensions: Account, Period, HSP_View, Year, Scenario, Version, Entity, and Product CrossRef("5800") returns: "BegBalance"->"No HSP_View"->"No Scenario"->"No Version"->"No Entity"->"No Product"->"5800";
CrossRef(accountName, prefix)	Generates a cross-reference by adding the specified prefix to each dimension name (except Currency, Period and Year), followed by the specified account. The prefix should be in double quotes. For example, in an application with the following dimensions: Account, Period, HSP_View, Year, Scenario, Version, Entity, and Product CrossRef("5800", "NoX") returns: "BegBalance"->"NoX HSP_View"->"NoX Scenario"->"NoX Version"->"NoX Entity"->"NoX Product"->"5800";
CrossRef(accountName, prefix, true)	Generates a cross-reference by adding the specified prefix to each dimension name, including Year (except Currency and Period), followed by the specified account. For example, CrossRef("5800", "NoX", true) returns: "BegBalance"->"NoX HSP_View"->"NoX Year"->"NoX Scenario"->"NoX Version"->"NoX Entity"->"NoX Product"->"5800";
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
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, review the formula details on the **Member Formula** tab. The most common error message is "Failed to execute." This occurs when you use parameters in the expression incorrectly. These actions cause "Failed to execute" error messages:

- Entering the wrong number of parameters in the formula expression
- Misspelling member names, functions, or variable names
- Not surrounding member names with quotation marks
- Including numbers where text is required

Administering Forms

Create forms for entering data based on your planning needs.

Related Topics

- About Forms
- Creating Forms
- Working with Forms and Form Components
- Managing Forms and Folders

About Forms

Forms are grids for entering data. You can create forms to meet your needs and then you can use the forms and other artifacts to design dashboards to summarize the data.

See Designing Dashboards.

Troubleshooting

For help with resolving form functional and performance issues, see Resolving Form Functional and Performance Issues in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Form Components

Point of View

Select point of view (POV) dimensions such as year, scenario, and version to define the context for pages, rows, and columns. For example, if Scenario is Budget, all data displayed and entered in pages, rows, and columns is written to the Budget scenario dimension member. Each POV dimension is set to one member, which users can't change.

To simplify a form's POV, or better tailor it to the needs and roles of users, you specify only relevant members or define user variables. See Defining Form Page and Point of View.

See also, Managing User Variables.

Page Axis

Use the page axis to specify member combinations that may span dimensions so that users can work with data in smaller, logical views. Each page axis item can have members selected from different dimensions. Users see only members they can access.

You can specify multiple page drop-down lists, and select members using relationship functions or attributes. Switch between member sets by selecting them from the page axis.

You display member names or aliases on the page axis. You can specify the number of members in a page dimension that enables a search drop-down list on the data entry page, useful if dimensions contain many members. See Defining Form Page and Point of View.

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.

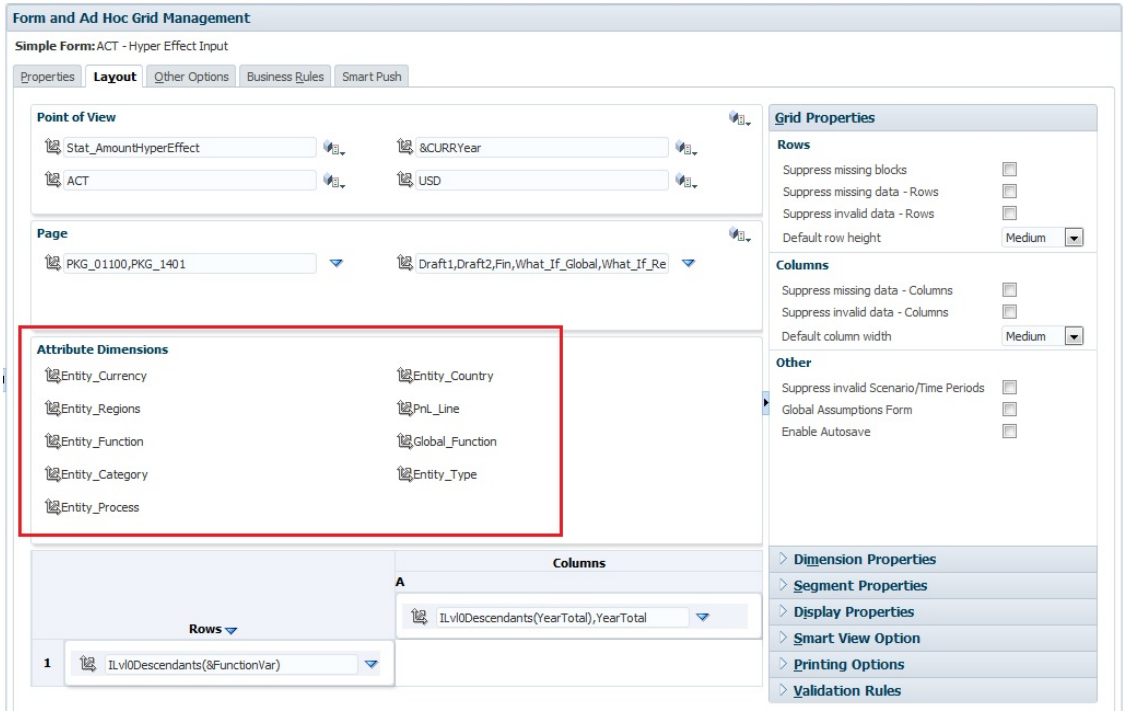
By default, forms have one set of rows and columns. You can add rows and columns to create asymmetrical combinations of members. See Creating Asymmetric Rows and Columns.

Attribute Dimensions

In addition to using attribute dimensions as filters, you can define attributes as unique dimensions within the application so that they can be placed on an axis other than their associated base dimension. This enables you to perform cross-dimensional rollups across attribute members.

Attribute dimensions are optional on form grids and are listed separately on the **Layout** tab of the Form Designer. To use an attribute dimension in a form, drag the attribute dimension to the point of view, page, row, or column.

> **Caution:**
>
> Don't drag the attribute on any axis when using it as a filter as you'll cause a cross-dimensional reference which will have a performance impact. When using an attribute dimension as a filter, there is a form option that shows the associated attribute (just like there is an option to show alias). See Defining the Layout.



Form Design Considerations

Forms and Cubes

When you create a form, you associate it with a cube, which determines the form's valid members. For example, if you assign a form to the Revenue cube, you can add only accounts that are valid for the Revenue cube. Entered data is saved to the selected cube's database.

> **Note:**
>
> - You can't change a form’s cube after assigning it.
> - You can edit form accounts only if their source cube matches the form's cube.
> - If you add an account to a form associated with a cube other than the account’s source cube, the account is read-only on that form.

Forms and Permissions

Assign permissions to a form to determine which users can modify its design (for example, layout and instructions) and input data. Users can edit forms only if they have permission to one secured dimension’s member. For example, if users have read-only permission to the Europe entity, the rows and columns that include the Europe entity are read-only. Users can change data only for members to which they have write permission.

Forms and Versions

For bottom-up versions, rows and columns with level 0 members allow data entry. Rows or columns set to a parent member are read-only. The point of view must also be set to the level 0

member to allow data entry on a bottom-up version. Target versions allow data entry in parent and children members.

Filtering Form Members by Attributes

You can select members by using attributes. For example, on the Entity dimension you can select members by a specific Region such as South. The resulting grid will only contain members that have the South attribute (for example, TX, NM, and so on). Values can be entered and saved into rows and columns filtered by attributes.

Forms and Shared Members

Because you can't select shared members individually, select them using a relationship function. For example, select an alternate functional rollup to include all members under that rollup. Users can enter values in rows or columns that display shared members, and data is saved to the base members in the database.

Forms and Calculations

To optimize calculations, select row members using relationships (such as Descendants or Children) instead of selecting individual children. For example, calculating individual parent-level totals could take several passes, so use a relationship instead.

Understanding Implied Sharing in Forms

Some members are shared even if you don't explicitly set them as shared. These members are implied shared members.

When an implied share relationship is created, each implied member assumes the other member’s value. The application assumes shared member relationships in these situations:

- A parent has only one child
- A parent has only one child that consolidates to the parent
- The data type of the parent is Label Only. The parent inherits the value of the first child, regardless of the child’s aggregation settings.

In a form that contains members with an implied sharing relationship, when a value is added for the parent, the child assumes the same value after the form is saved. Likewise, if a value is added for the child, the parent usually assumes the same value after a form is saved.

For example, when a calculation script or load rule populates an implied share member, the other implied share member assumes the value of the member populated by the calculation script or load rule. The last value calculated or imported takes precedence. The result is the same whether you refer to the parent or the child as a variable in a calculation script.

Creating Forms

Table 19-12 Form Creation Checklist

Task	Want to Know More?
Define the layout, including: • Adding rows and columns • Assigning dimensions to columns and rows • Assigning attribute dimensions to the point of view, page, columns, and rows • Selecting dimension members for users to work with • Setting grid properties • Setting dimension properties • Adding formula rows and columns • Setting Smart View form display options • Setting display properties • Setting printing options • Adding and updating validation rules	See Defining the Layout.
Define page axis and point of view	See Defining Form Page and Point of View.
Select members	See Using the Member Selector
Set form precision, context menus associations, and whether to enable dynamic user variables	See Setting Form Precision and Other Options.
Select business rules and set properties	See About Rules.
Using Smart Push	See Moving Data from One Cube to Another Cube Using Smart Push
Define access permissions	See Setting Up Access Permissions.
Design formula rows and columns	See Designing Forms with Formula Rows and Columns .
Design data validation rules	See Designing Forms with Data Validation .

To create forms:

1. Click the **Navigator** icon  , then under **Create and Manage**, click **Forms**.
2. Under Form and Ad Hoc Grid Management, click  .
3. On the **Properties** tab, enter a name of up to 80 characters, and an optional description of up to 255 characters.
4. Select the **Cube** associated with the form. See Forms and Cubes.
5. **Optional**: Enter instructions describing how to work with the form.
6. Click **Next**, and see Defining the Layout.

Defining the Layout

When you create forms, they initially contain only one row and column, and all dimensions, except for the optional attribute dimensions, are in Point of View. Add rows and columns as needed. When setting row and column layout:

- Assign at least one dimension to the row and column axis.
- You can't select the same dimension for multiple axes. To use dimensions on multiple axes, set user variables in the point of view.
- You can select a dimension from any axis and drag it to another
- You can exclude members of dimensions from the form after a dimension is assigned to a row or column.

> **Note:**
>
> Attribute dimensions are optional and are displayed in their own area on the **Layout** tab. Drag the attribute dimension to the point of view, page, row, or column to use it in the form.

> **Caution:**
>
> When working with attributes as filters, don't drag the attribute on any axis as you'll cause a cross-dimensional reference which will have a performance impact. When using an attribute dimension as a filter, there is a form option that shows the associated attribute (just like there is an option to show alias).

To define the layout:

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

   See Selecting and Opening Forms and Folders.

2. Click the  icon next to a dimension on the **Layout** tab, and then drag it to the desired axis (**Rows** or **Columns**). Optionally, you can also select the attribute dimensions to use on **Point of View**, **Page**, **Rows**, and **Columns**.
3. To edit dimensions added to rows and columns, click 

   next to a dimension:

   - Select **Edit** to launch the member selector and define the dimension members and variables you want to include on the form.

Table 19-13 Member Selection Options for Forms

Member Selection Option	Description
Members	Make selections by clicking a member in the member list. See Selecting Members for Forms.

Table 19-13 (Cont.) Member Selection Options for Forms

Member Selection Option	Description
Variables	Make selections for the following options: – **User Variables** (see Selecting User Variables as Members) – **Substitution Variables** (see Selecting Substitution Variables as Members) – **Attributes** (see Selecting Attribute Values as Members) – **UDAs** (see Selecting UDAs as Members)

- Select **Exclude** or **Exclude All** to use the member selector to define the dimension members you want to exclude:
  - **Exclude**: Selecting this option excludes members by ID. Only the specified members (base or shared) will be excluded.
  - **Exclude All**: Selecting this option excludes members by name. If a base member is specified, then the base and all of its shared members will be excluded. If a shared member is specified, then this member, its base member, and all other shared members of this member will be excluded.

  Service Administrators can also choose to display or hide excluded members to users in the segment drop-down member selector list using the application setting, **Filter Out Excluded Members in Segment Drop-down**.

  See What Application and System Settings Can I Specify?.
- Select **Clear** to remove selections.

4. **Optional**: To change the order of dimensions in rows or columns, click  , and then **Move Dimension Up** or **Move Dimension Down**.
5. Select a row header (such as 1 or 2) to specify row properties, or a column header (such as A or B) to set column properties described in Segment Properties.

   To enable users to transfer global assumptions from a test to a production environment for a form, select **Global Assumptions Form**, and then update the form to store global assumptions. See Designing Forms with Global Assumptions.
6. **Optional**: Add formula rows and columns. See Adding Formula Rows and Columns.

Subtopics:

- Segment Properties
- Setting Form Grid Properties
- Setting Dimension Properties
- Setting Display Properties
- Setting Smart View Form Options
- Setting Printing Options
- Including Data Validation Rules in Forms
- Enabling Drilling on Shared Members

Segment Properties

To access **Segment Properties** for a form, click the row number in the form layout.

Table 19-14 Segment Properties

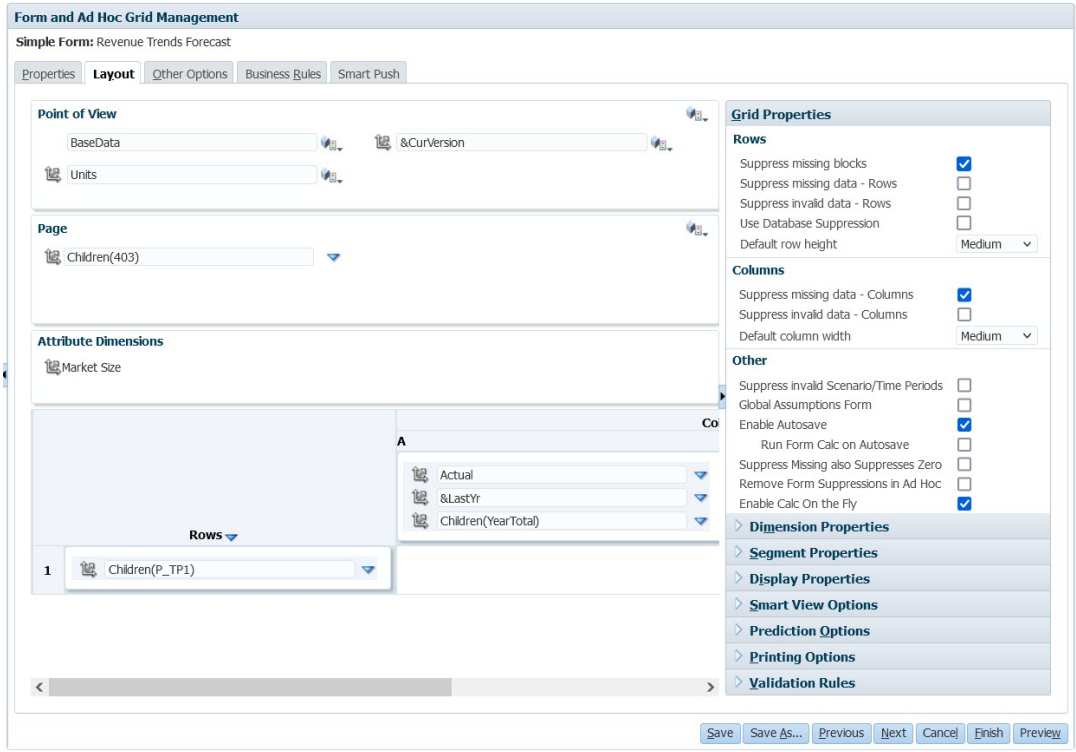
Option	Description
Apply to all rows	Clear to specify different properties for individual rows.
Apply to all columns	Clear to specify different properties for individual columns.
Hide	Conceal a column or row
Read-only	Create a read-only row or column, enabling users to compare old, read-only data with new, editable data.
Show separator	Create a bold border before the segment to visually distinguish it
Suppress hierarchy	Suppress indentation.
Suppress missing data	Hides empty rows or columns. Clear to display rows or columns with "#MISSING" in cells when data is missing.
Suppress invalid Scenario/ Time Periods	This option ties the column display to the start and end period for the scenario. If selected, it suppresses the display of time periods outside of the scenario time period range.
Column width	• **Default**: Use the column width defined at the grid level (under **Grid Properties**) • **Small**: 50px • **Medium**: 100px • **Large**: 300px • **Size-to-Fit**: Column will expand to fit data • **Custom**: Specify the number of pixels
Row height	• **Default**: Use the row height defined at the grid level (under **Grid Properties**) • **Medium**: Use standard height • **Size-to-Fit**: Row will expand to fit data • **Custom**: Specify the number of pixels
Enable drop-down for dimensions	Allows you to set up drop-down member selectors on row dimensions in Oracle Smart View for Office grids and business process web forms. When these drop-down selectors are enabled, form users can directly select a member from the drop-down list on the row dimension in the form. For example, suppose there are two dimensions placed on the row axes, Entities and Line item; the member selector drop-down may be enabled for the Entity dimension members, for the Line item dimension members, or for both. Depending on the settings chosen by the Service Administrator, the drop-down member selector may also allow users to add data to member rows that may have otherwise been suppressed or excluded. **Note:** You cannot enable a drop-down member selector on a form that already contains a server-side member formula.

Setting Form Grid Properties

Grid properties define how rows and columns display.

To set grid properties:

1. Open the form, and then click **Layout**. See Selecting and Opening Forms and Folders.



2. In **Grid Properties**, set row and column properties as follows:

Table 19-15 Form Grid Properties

Option	Description
Suppress missing blocks	On multiple rows: Greatly improves the efficiency of **Suppress missing data**. This option can degrade performance if no or few rows are suppressed. Test forms before and after using this setting to determine if performance is improved. Note that if you use this option: • Some suppressed blocks may ignore Dynamic Calc members • Row members may not indent
Suppress missing data	Hide rows or columns without data. Clear to display "#MISSING" in cells when data is missing.
Suppress invalid data	Hide rows or columns with invalid data. Cells with invalid data are read-only.

Table 19-15 (Cont.) Form Grid Properties

Option	Description
Use Database Suppression	Applies row suppression at the Oracle Essbase level instead of at the business process level, which reduces the amount of data on the business process side and eliminates the impact on query thresholds. **Note:** If you use this option and you have a formula column on the form that results in a #MISSING value, the row won't appear on the form because it will be suppressed in Essbase and won't be returned to the business process for formula evaluation.
Default row height	• **Medium**: Row will have medium height. • **Size-to-Fit**: Row will expand to fit data • **Custom**: Specify the number of pixels
Default column width	Specify the width in pixels: • **Small**: 50px • **Medium**: 100px • **Large**: 300px • **Size-to-Fit**: Column will expand to fit data • **Custom**: Specify the number of pixels
Suppress invalid Scenario/ Time Periods	This option ties the form grid display to the start and end period for the scenario. If selected, it suppresses the display of time periods outside of the scenario time period range.
Global Assumptions Form	To transfer global assumptions from a test to a production environment for a form, select **Global Assumptions Form**, and update it to store assumptions. See Designing Forms with Global Assumptions.
Enable Autosave	This option lets users use successive undo actions with Ctrl+Z. It also automatically saves their changes when they navigate between cells. **Note:** For optimal performance, use only dense dimensions on rows and columns.
Run Form Calc on Autosave	Only available if **Enable Autosave** is selected. If **Run Form Calc on Autosave** is selected, dynamically calculated cells that depend on modified and saved values (for example, a row with a formula that calculates a percentage of an aggregated parent value) are updated and display on a green background.

Table 19-15 (Cont.) Form Grid Properties

Option	Description
Suppress Missing also Suppresses Zero	When this option is selected along with the **Suppress missing data** option for forms, all rows or columns containing both #Missing and zeros are suppressed. When selected, this setting overrides the runtime suppression selections made for suppressing missing data or zeroes in Oracle Smart View for Office. **Note:** For Forms 2.0, filtering is not applied if the row or column has a combination of zeroes and #Missing.
Remove Form Suppressions in Ad Hoc	Select to allow Smart View users to perform ad hoc analysis on a form that has other suppression options specified. If there is missing data on the form, the base dimension members for the columns or rows are displayed on the ad hoc grid in Smart View. This allows users to continue ad hoc analysis on the form, even though suppression options were enabled in form design. **Note:** Selecting this option does not override any suppression options users may have set in the Smart View **Options** dialog, **Data Options** tab. If the **Remove Form Suppressions in Ad Hoc** option is enabled for the form, and users see an error when performing ad hoc analysis on the form, then they should clear the suppression options in Smart View.
Enable Calc On the Fly	When you select this option, in Forms 2.0, when you hover over a cell with a member formula attached, you see the member formula associated with the cell and you can perform calc on the fly. Calc on the fly allows you to perform calculations without having to submit your changes. Note that Forms 2.0 is available only if **Redwood Experience** is enabled.

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

Setting Dimension Properties

You can specify dimension display properties such as if to display the member names or aliases, and whether to permit users to view member formulas. These properties apply to row, column, page, and point of view dimensions.

To set dimension properties:

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

   See Selecting and Opening Forms and Folders.

2. Click in a point of view, page, row, or column dimension, then under **Dimension Properties** apply properties such as:

Table 19-16 Dimension Properties

Property	Description
Apply to all row | column | page dimensions	Apply properties to all row, column, or page dimensions
Apply to all POV dimensions	Apply properties to all point of view dimensions
Start expanded	For row or column dimensions: Expand and display the dimension member list.
Enable custom attributes	For row or column dimensions in Forms 1.0 only: Use custom attributes.
Drill on shared members	For row or column dimensions: Enable drilling on shared members when the shared member is on a parent member for the main hierarchy.
Flex beyond form definition	For row and column dimensions in flex forms: Select to allow Oracle Smart View for Office users to add row or column members that are not included as part of the form definition. Enable this option on a per-row or per-column basis. You may also select **Apply to all row dimensions**, but note that all selected properties in the current row dimension will be applied to all row dimensions in the form. This option is displayed only when **Enable flex form for rows** or **Enable flex form for columns**, or both, is selected in the **Smart View Options** section of the form definition (see Setting Smart View Form Options).
Row Width	For row dimensions: Select from the following column sizing options: • **Small**: 50px • **Medium**: 100px • **Large**: 300px • **Size-to-Fit** (default): Column will expand to fit content • **Custom**: Specify the number of pixels Note that this setting is retained if the form is viewed on its own or within a dashboard.

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

Setting Display Properties

Define display options on the **Layout** tab.

> **Note:**
>
> - Assign the Account dimension to a row axis.
> - Account, Entity, Versions, and Scenario dimensions can't be assigned to the column axis.
> - The Entity dimension can be assigned to any axis.
> - Assign Version and Scenario dimensions to the page or Point of View axis.

To set display properties:

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

   See Selecting and Opening Forms and Folders.

2. Select **Display Properties**, and then options such as:

Table 19-17 Display Properties

Option	Description
Make form read-only	Prevent users from editing the form.
Hide form	Conceal forms that are part of a dashboard, or accessed from menus or task lists.
Display missing values as blank	Display cells without data as empty cells. Clear to display "#MISSING". See Displaying #MISSING with Smart Lists.
Enable Mass Allocate	Users must have the Mass Allocate role to use this option. See Working with FreeForm .
Enable Grid Spread	See Working with FreeForm .
Enable cell-level document	Enable users to add, edit, and view documents in form cells, depending on access permissions. See Working with FreeForm .
Message for forms with no data	Enter custom text to display when invalid data exist. Leave blank to display `There are no valid rows of data for this form.`
Hide Save Confirmation Message	Prevents the form save confirmation message from being displayed to users.

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

Setting Smart View Form Options

Specify options for form display in Oracle Smart View for Office.

To set Smart View form options:

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

   See Selecting and Opening Forms and Folders.

2. Select **Smart View Options**, and then settings such as:

Table 19-18 Smart View Options

Option	Description
Disable spreading	Disables spreading options in Smart View, including default spreading, spreading data for time periods, spreading using grid spread, and spreading using mass allocations.
Disable formatting	Disables the **Formatting** options (Save, Clear, Apply) on the provider ribbon in Smart View.

Table 19-18 (Cont.) Smart View Options

Option	Description
Repeat member labels	Allows repeated member labels in forms. If this setting is cleared, you may override it in Smart View to allow repeated members in forms by selecting **Repeat Member Labels** in the Smart View **Options** dialog, **Formatting** tab. For forms with **Enable drop-down for dimensions** selected (Segment property), the form layout will automatically behave as if the **Repeat member labels** option is selected, regardless of whether or not it is selected. For flex forms (see **Enable flex form for rows** and **Enable flex form for columns** below), the **Repeat member labels** option is automatically enabled and cannot be cleared.
Disable sheet protection	Disables protection on data cells in forms in Smart View. Allows users to enter or paste data into cells, but does not allow users to submit the changed data. By default, this option is cleared.
Enable flex form for rows	Enables the form as a flex form on rows in Smart View. When this option is enabled, the **Repeat member labels** option is automatically enabled, and cannot be cleared. See Designing Flex Forms.
Enable flex form for columns	Enables the form as a flex form on columns in Smart View. When this option is enabled, the **Repeat member labels** option is automatically enabled, and cannot be cleared. See Designing Flex Forms.
Show invalid members for flex form	Retains invalid members on the flex form sheet after a refresh in Smart View. Invalid members can be the result of: • Mistyping or misspelling a member name • Errors when copying and pasting member names from one sheet to another, or within the same sheet • Entering a member name that is beyond the form definition When this check box is enabled, invalid members are retained on the sheet upon Refresh and highlighted, making them easy to find and correct. See Designing Flex Forms.

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

Setting Printing Options

To set printing options:

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

   See Selecting and Opening Forms and Folders.

2. Select **Printing Options**, and then settings such as:

Table 19-19 Printing Options

Option	Description
Include supporting detail	Print supporting detail as extra rows in PDF files as follows: **Normal Order**: In the same order as on the **Supporting Detail** page **Reverse Order**: Before the member associated with it. Supporting detail for children displays above parents, and the order of siblings is preserved
Show comments	Display cell text notes
Format data	Apply number format settings
Apply precision	Print data using a specific number of decimal places

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

Including Data Validation Rules in Forms

You can add and update validation rules for grids, columns, rows, or cells, on the Layout tab. When processed, rules may change the color of cells, and display validation messages to users during data entry.

> **Note:**
>
> Before adding data validation rules, consider the function it will perform. For information and best practices about using validation rules:
>
> See Managing Data Validation.

To include data validation rules in forms:

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

   See Selecting and Opening Forms and Folders.

2. Select **Validation Rules**, and then select validation rule options:

> **Note:**
>
> The right-click menu options that display are context-sensitive, and depend on if rules and menu options were already added.

Table 19-20 Validation Rules Options

Option	Description
Add/Edit Validation Rules	Create or modify existing rules in the **Data Validation Rule Builder** dialog box.
Copy | Paste Validation Rules	Copy selected rules and paste them in a new location.

Table 19-20 (Cont.) Validation Rules Options

Option	Description
Validate only for pages with existing blocks	Run validations only on page combinations that have potential blocks. The only exceptions being page combination having any Dynamic Calc, Dynamic Calc and Store, Label only, or Store with one child member settings. In this case, the page is loaded.
Validate only for cells and pages the user has access to	Run validations as the current user, using their security, and, not a Service Administrator.

3. Build and validate the rules.

   See Creating and Updating Data Validation Rules.

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

Enabling Drilling on Shared Members

Form designers often use shared members to create alternate rollup structures within an application. To increase usability for these alternative hierarchies, the application provides the ability to drill on a shared member for a given dimension. The option to drill is a dimension property which provides the flexibility to focus the drill behavior to a particular dimension. The drilling option is only available for dimensions placed on the row or column axis during form design. Page or POV dimensions don't have drill capability.

For example, using the following hierarchy representing the Entity dimension:

World			
	USA		
		West	
			CA
			AZ
			CO
		East	
			NY
			MA
			PA
		South	
			TX
			FL
			NM
		North	
			WA
			MI
			MN
	Southwest		
		South (Shared)	
		West (Shared)	
	Northeast		
		North (Shared)	
		East (Shared)	

A form with iDescendants(Southwest) defined on the row would return the following members:

- TX
- FL
- NM
- South
- CA
- AZ
- CO
- West
- Southwest

To enable drilling on shared members in a form:

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

   See Selecting and Opening Forms and Folders.
2. Click in a row or column dimension to expand **Dimension Properties**.
3. Under **Dimension Properties**, select **Drill on shared members**.
4. Click **Save**.

About Precision Settings

In **Other Options**, you can control data precision by applying minimum and maximum values for different account types. For example, you can truncate and round the decimal portion of longer numbers. Precision settings affect just value display, not their stored values. For example, if **Minimum Precision** is 2, and the value 100 from Q1 is spread to Jan, Feb, and Mar, the month cells display 33.33 when they are not selected. When selected, they display their actual values (for example, 33.33333333333333). Because the number of decimal places for storing values is finite, when the values for Jan, Feb, and Mar aggregate to Q1, 33.33333333333333 is multiplied by 3, and Q1's value is 99.99999999999998.

Setting Form Precision and Other Options

To set precision and other options:

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

   See Selecting and Opening Forms and Folders .

2. In **Precision**, specify the number of decimal positions displayed in a cell for **Currency Values**, **Non-Currency Values**, and **Percentage Values**.

   See About Precision Settings.

   Specify **Minimum** values to add zeros to numbers with few decimal places. Specify **Maximum** values to truncate and round the decimal portion of large numbers. For example:

Table 19-21 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
100.12345	2	4	100.1235
100	2	4	100.00

> **Note:**
>
> Your selections override the precision set for the currency member. To use the currency member's precision setting instead, select **Use Currency member precision setting**.

3. Under Context Menus, associate menus with the form by moving them from **Available Menus** to **Selected Menus**. For multiple menus, click a "move" options on the right to specify the display order.

   For information on how to create Context Menus (Action Menus):

   See Administering Action Menus.

4. Select **Enable dynamic user variables** to allow dynamic user variables in the form, and move **Available User Variables** to **Selected User Variables**.

   See Dynamically Setting User Variables.

Moving Data Using Smart Push

For more meaningful and complete reporting, you can use Smart Push to instantly move comments, attachments, and supporting detail from source cubes to a reporting cube while working in forms. Users can then do more analysis on the data coming from the different cubes.

To learn how to move data using Smart Push, see Moving Data from One Cube to Another Cube Using Smart Push in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

Troubleshooting

For help with resolving issues with Smart Push, see Resolving Issues with Smart Push in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Creating Asymmetric Rows and Columns

Asymmetric rows and columns contain different sets of members selected across the same dimensions. For example:

Row/Column A—Scenario = Actual, Time Period = Q1

Row/Column B—Scenario = Budget, Time Period = Q2, Q3, Q4

To create asymmetric rows or columns:

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

   See Selecting and Opening Forms and Folders.

2. Click  to select the dimension to modify.

3. Click  to the right of the dimension name, and then modify the members selected for this dimension.

   See Using the Member Selector.

4. Click **Save** to continue, or click **Finish** to save and close.

Adding Formula Rows and Columns

You can define formulas for rows and columns. For example, you can create a formula column to calculate the percentage variance between monthly sales columns. The formula for a row or column applies to all row or column dimensions. To define or assign existing formulas to forms, select the row or column on the **Layout** tab, and use the options under **Segment Properties**.

See Creating Formulas.

> **Tip:**
>
> Consider adding a formula row between two other rows to create a blank row. Blank rows help you visually separate subtotals and totals in a form.

To add formula rows and columns:

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

   See Selecting and Opening Forms and Folders.
2. On the **Layout** tab, right-click **Rows** or **Columns**.
3. Select **Add Formula Row** or **Add Formula Column**.
4. Click the new **Formula Label**, and then enter the formula name.
5. Click the row or column number of the Formula Label, and specify **Segment Properties** such as **Display formula on form** so users can view the formula when they click  .
6. Under **Segment Properties**, for each dimension in **Formula Data Type**, select how to display the formula result, such as a SmartList, Date, or in a Text format.
7. Enter the formula, and then click  .

   See Editing Formulas.
8. Click **Validate** to find and fix any errors.

Defining Form Page and Point of View

You can select dimensions and members for the page axis and point of view (POV). POV dimensions and members must be valid for the form cube, and not assigned to a page, column, or row axis. The POV sets the members that define data intersections and the form's context. If you use user variables, variable names displays in the POV.

See Managing User Variables.

To define page axis and point of view:

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

   See Selecting and Opening Forms and Folders.
2. Click  and then drag that dimension to **Page** .
3. Click  for each page axis dimension and select members.

   See Using the Member Selector.

> **Note:**
>
> To shorten the number of members displayed in the member list so that the **Search** box can be easily seen, go to the Home page, click **Application**, and then click **Settings**. For the **Number of Items on the Page Drop-down** option, enter a lower value.

4. Repeat Steps 2 and 3 as needed. Assign multiple dimensions to the page axis to enable users to change dimensionality while entering data. They can use **Display Properties** to use the most recent selection.
5. Specify **Dimension Properties**.

   See Editing Dimension Properties.
6. Click  and drag a dimension to **Point of View** to add them, or members, to the form's point of view.
7. Specify **Dimension Properties**.

   See Editing Dimension Properties.
8. Click **Save** to continue, or **Finish** to save and close.

Designing Specific Types of Forms

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

Designing Forms with Formula Rows and Columns

Formula rows and columns contain formulas that perform mathematical calculations on grid members. For example, you might want to rank values in a particular column or calculate variance between two rows. A formula consists of grid references, arithmetic operators, and mathematical functions. To define or assign existing formulas to forms, select the appropriate row or column on the **Layout** tab and then make selections under **Segment Properties**.

See Adding Formula Rows and Columns.

For information about creating formulas and using the mathematical functions:

See Form Formula Functions.

Designing Forms with Data Validation

You can design forms that include predefined data validation rules that help implement business policies and practices. You can specify cell colors and data validation messages that are generated on the form if entered data violates a validation rule. Data validation rules are saved as part of the form. See Including Data Validation Rules in Forms.

See also, Managing Data Validation.

Designing Forms with Global Assumptions

Form designers typically create a specific-purpose form that stores global assumptions (also called "drivers" or "driver data"), such as a depreciation value, tax rate, or unit price. This is useful if you want to migrate the form's driver data using Migration, for example, from a test environment to a production environment. Such forms are typically small, and are used as "lookup tables" that drive calculations.

> **Note:**
>
> Global assumptions can't be used with dashboards or forms with multiple segments.

To enable transferring global assumptions from a test to production environment during form design:

1. In the **Layout** tab, click **Grid Properties**.
2. Under **Other**, select **Global Assumptions Form**.
3. Update the form to store driver data such as a tax rate. When migrating with Migration, this enables migrating the driver data contained in the tagged form.

Designing Forms for Rolling Forecasts

Related Topics

- About Rolling Forecasts
- Creating Rolling Forecasts
- Modifying Rolling Forecast Variables

About Rolling Forecasts

In a traditional forecast, the forecast cycle is always tied to the fiscal year end, and the months in the forecast period keep reducing as the months in the fiscal year progress.

Rolling forecasts differ from traditional forecasts in that they are continuous without regard to the annual fiscal year end period. The periods in a rolling forecast roll along based on the predefined window for the rolling forecast. The periods are generally defined on a monthly or quarterly basis. Monthly rolling forecasts are generally in 12-month, 18-month, or 24-month cycles. In a 12-month cycle, the 12-month period constantly shifts each month, and every month the forecast is for the next twelve months without regard to the actual fiscal year end.

For example, assume a company has a fiscal calendar for July through June. In the first month of year (Jul, FY11) the company's users fill in the forecast scenario for the periods Jul 11 – Jun 12. In the next month (Aug 11), the users 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 19-1 12-Month 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 19-2 Quarterly Rolling Forecast

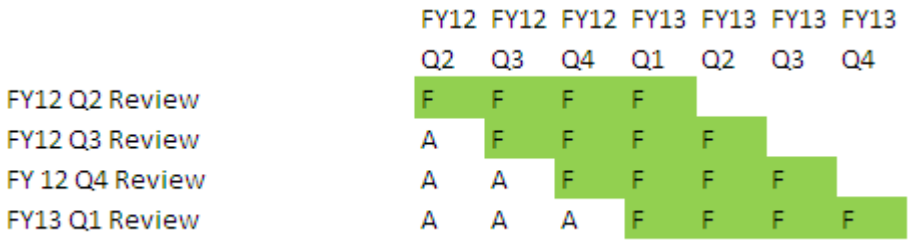
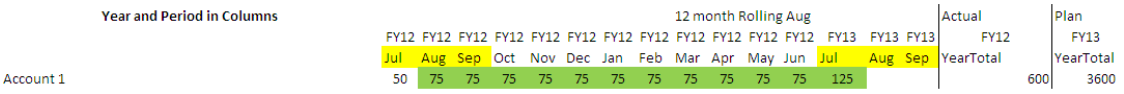


Figure 19-3 Quarterly Trailing Rolling Forecast (Rolling Quarters with a Cumulative Total)

	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 19-4 Rolling Forecast Where There are Additional Segments for Actual and Plan Year



Creating Rolling Forecasts

> **Note:**
>
> Only Service Administrators can create and work with rolling forecasts. This includes the ability to see the rolling forecast option when designing a form, the ability to move a rolling forecast from the user interface, and the ability to delete or modify substitution variables.

To design a form for a rolling forecast:

1. Create a new form.

   See Selecting and Opening Forms and Folders.
2. In the Layout tab, drop the **Years** and **Period** dimensions in the column axis.
3. Right-click the column segment header and select **Rolling Forecast Setup**.

The Rolling Forecast Setup menu option is only available when the Years and Period are in the same grid axis (row or column).

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

Table 19-22 Rolling Forecast Setup Options

Option	Description
Prefix	Allows rolling forecast substitution variables to be distinct from other substitution variables; for example, 4QRF designates that the forecast is a 4-quarter rolling forecast.
Reuse existing substitution variables	Select if you wish to specify a prefix that you have previously used.
Start Year	The year in which the rolling forecast starts; for example, FY11. Either enter the start year or click  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  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.

Note:

- Substitution variables are based on the periods selected for the Start Year and Start Period.

  See Selecting Substitution Variables as Members.

- When designing a form, if the Rolling Forecast Setup dialog box is brought up from a row or column with the Year and Level 0 time period selected (for example,FY12/Jan), the Start Year and Start Period are automatically filled in. Values are not automatically filled in if members in the column are selected using functions, variables, or non-level-0 members.
- To reuse the rolling forecast variables in a different form, right-click the column header in the new form to invoke the Member Selector.

Modifying Rolling Forecast Variables

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

To modify rolling forecast variables in a form:

1. From the Home page, click **Data**.
2. Open the rolling forecast form.
3. Select a column header in the form, click **Actions**, and then select **Set Rolling Forecast Variables**.
4. In the **Set Rolling Forecast Variables** dialog box, enter or edit values for the Years and Period dimensions.

   You can shift the values up or down by changing the selection next to **Shift Values By**. When you change the selection next to **Shift Values By**, the values for the Years and Period dimensions are automatically repopulated to show the resulting year and period values after the shift.
5. Click **Apply**.

   The new values flow through to all the forms where these substitution variables are used, and those forms will reflect the changes.

Designing Flex Forms

Flex forms are a form type that provides flexible row and column management in Oracle Smart View for Office.

Flex forms retain all regular form properties and features, such as running business rules attached to the flex form, running rules on save or before save, and enabling some ad hoc-specific functionality. However, with flex forms, dimension and member row and column cells and all data cells are unprotected. This means that Smart View users can rearrange row and column members, and sort or move rows or columns. Modified row and column order is maintained on refresh and during submit. Smart View users can also filter data using Excel's filtering functionality.

> **Note:**
>
> Flex forms are used only in Smart View and not in the web interface.

During form definition:

- Select **Enable flex form for rows**, or **Enable flex form for columns**, or both, to enable flex form-specific features. These options are located in the **Layout** tab, under **Smart View Options**.

  See Setting Smart View Form Options.
- If you have selected **Enable flex form for rows**, or **Enable flex form for columns**, or both, then you can enable the **Show invalid members on flex form** option. This option is located in the **Layout** tab, under **Smart View Options**.

  When entering member names in a flex form, errors may occur; for example, a user may mistype or misspell a member name; make an error when copying and pasting member names from one sheet to another, or within the same sheet; or enter a member name that

is beyond the form definition, resulting in an invalid member. By enabling the **Show invalid members for flex form** option, the flex form will retain any invalid members on the form after refresh. Invalid members are highlighted making them easy to locate and edit. After users make necessary corrections to member names and refresh again, they can then enter and submit data on the flex form.

See Setting Smart View Form Options.

- On a row-by-row basis, or column-by-column basis, or both, enable the **Flex beyond form definition** option. The **Flex beyond form definition** property is located under **Dimension Properties** in the **Layout tab** when a row or column is selected.

  When this option is enabled for a row or column dimension, Smart View users may add row or column members to the flex form that are not included as part of the form definition. Do this only for the rows or columns you require flex functionality.

  See Setting Dimension Properties.

- Define action menus to open another flex form in Smart View.

  To define action menus:

  See Administering Action Menus.

  To attach an action menu to a flex form:

  See Setting Form Precision and Other Options.

For information on using flex forms:

See Flex Forms in *Working with Oracle Smart View for Office*.

Working with Forms and Form Components

Related Topics

- Selecting and Opening Forms and Folders
- Previewing Forms
- Printing Form Definitions
- Searching for Forms
- Editing Forms
- Moving, Deleting, and Renaming Forms
- How Cell Formatting in Smart View Persists in FreeForm

Selecting and Opening Forms and Folders

Use these procedures to select and open form folders and the forms they contain.

To select and open form folders or forms:

1. Click the **Navigator** icon  , and then under **Create and Manage**, click **Forms**.
2. Perform one of the following steps:
   - To open a folder, expand Library beneath **Folders**, and then select a form folder.
   - To open a form, expand Library beneath **Folders**, select a form folder, and then select a form from the list that displays in the **Form and Ad Hoc Grid Management** pane.

3. After you select a form folder, use the buttons next to **Folders** to create, delete, rename, move, and assign access to the folder.
4. To manage forms, select one of the following actions in the **Form and Ad Hoc Grid Management** pane:
   - To open the form, select a form, and then click **Edit**.
   - To create a form, click **Create simple form** .
   - To edit, delete, move, assign access, rename forms, or show usage (for composite forms only), select a form, and use the corresponding button.

   The icon next to the form name indicates the type of form:

   -  Simple form
   -  Composite form

     > **Note:**
     >
     > Oracle has stopped supporting composite forms. However, your existing composite forms will continue to work. Oracle recommends that you shift your usage to dashboards instead of composite forms.

   -  Ad hoc grid

For information about setting up ad hoc grids:

See Using Ad Hoc for Data Entry and Analysis in *Working with FreeForm*.

Previewing Forms

While you're designing forms, you can preview the dimensions that are assigned to the Point of View, columns, rows, and page axes. Previewing displays member attributes, alias, and data associated with forms, although new data can't be entered.

Previewing completes regular form design validation checks, and checks for proper evaluation of any data validation rules included in the form. Data validation rules must be properly completed before the form can be saved. In addition, data validation rules are saved as part of the form. If you don't save changes to a form, any data validation rule changes made after the form was last saved are lost.

To preview a form's design:

1. Under **Form and Ad Hoc Grid Management**, select a form, and then click **Edit** to open the form.
2. With the form open, click **Preview**.

   The form opens in edit mode in a new tab.
3. Resolve any issues reported during the design validation checks, including any issues with data validation rules.
4. Save the form to ensure that updates are saved, including any changes to data validation rules.

Printing Form Definitions

Service Administrators can print form definition reports that include information on dimension members, business rules, access permissions, and other form components.

To create and print form definition reports:

1. Click the **Navigator** icon, and then under **Monitor and Explore**, click **System Reports**.
2. Select the **Forms** tab.
3. Under **Select Report Options**, select the form definitions to print by moving them from **Available Forms** to **Selected Forms**.
4. **Optional**: Select **Include Member Selection List** to include column and row members on the report.
5. **Optional**: Select **Include Business Rules** to include associated business rules.
6. Click **Create Report**.

   Adobe Acrobat generates a consolidated report, including:

   - Cube
   - Description
   - Column dimension and members and additional column definitions
   - Row dimension and members and additional row definitions
   - Page and Point of View dimensions
   - Form access permissions
   - Associated business rules
7. To print the report, select **File**, and then **Print** on the Adobe toolbar.

Searching for Forms

To search for forms:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Forms**.
2. Under Form and Ad Hoc Grid Management, enter part or all the form name in the **Search** box, and press **Enter**.

   Ignoring capitalization, Search finds the next match.
3. Click to search forward (down) or to search backwards (up).

Editing Forms

You can edit the layout, members, and properties of forms. For example, you can add formula rows or columns to a form.

> **Note:**
>
> Previously, you could create composite forms (forms that comprise multiple simple forms) to summarize data. Oracle has stopped supporting composite forms. However your existing composite forms will continue to work. Oracle recommends that you shift your usage to dashboards instead of composite forms.
>
> See Designing Dashboards.

To edit forms:

1. Select the form, then click  (see Selecting and Opening Forms and Folders).

   Alternatively, on the Home page, you can click **Data**. Expand a form folder, click the name of a form to open it, click **Actions**, and then select **Edit**.
2. Select:
   a. **Properties** to edit the description and instructions. See Creating Forms.
   b. **Layout** to edit form layout. See Defining the Layout.
   c. **Other Options** to edit form precision, change the context menus associated with the form, enable/disable dynamic user variables, and select user variables. See Setting Form Precision and Other Options.

      See also, Administering Action Menus and Dynamically Setting User Variables.
   d. **Business Rules** to change which business rules are associated with the form, or modify business rule properties.

      See Administering Rules.
3. Choose an option:
   - To save your work, click **Save**.
   - To save a variant of a form with a new name, click **Save As**, enter a **New Form Name**, and then click **OK**. The updated form with the new name is added to the form list. If Redwood Experience is enabled, the newly named form opens in a new dynamic tab.

Moving, Deleting, and Renaming Forms

To move, delete, and rename forms:

1. Select the form.

   See Selecting and Opening Forms and Folders.
2. Perform any of the following tasks:
   - To move a form, click **Move**, and then select the destination folder.

     > **Note:**
     >
     > You can move multiple forms simultaneously if they are in the same folder.
   - To delete a form, click **Delete**.

- To rename a form, click **Rename**, and then enter the new name.

3. Click **OK**.

How Cell Formatting in Smart View Persists in FreeForm

> **Note:**
>
> Before you perform the following steps:
>
> See Saving Native Excel Formatting to EPM Cloud in *Working with Oracle Smart View for Office 22.100*.

To select the formatting for a form:

1. In the form, right-click, and then select **Apply**.
2. Select:
   - **Cell Styles**: To use the business process formatting
   - **Custom Styles**: To use the formatting saved in Oracle Smart View for Office

For more information:

- On which Microsoft Excel formatting features are supported in Smart View and FreeForm:

  See About Smart View Formatting in Planning Forms in *Working with FreeForm*.
- On saving Excel formatting:

  See the Oracle Smart View for Office User's Guide.

Managing Forms and Folders

Use the Form and Ad Hoc Grid Management and the Business Rule Folders pages to manage folders and forms. To access the Form and Ad Hoc Grid Management page, click the **Navigator** icon, and then under **Create and Manage**, click **Forms**. You can access the Business Rule Folders by clicking the **Navigator** icon, and then selecting **Rules Security** under **Create and Manage**.

Table 19-23 Tasks

Task	Topic
Create folders	Creating Folders
Move, delete, or rename folders	Working with Folders
Create forms	Creating Forms
Assign permissions to forms and folders	About Assigning Permissions to Artifacts, Rules, and Folders
Move and delete forms	Moving, Deleting, and Renaming Forms

To view all forms or business rules in a folder, click the folder's name in the left-hand folders area. To select all the forms, select the check box at the top of the forms list.

Creating Folders

Use folders to hierarchically organize forms and business rules. You can move folders within the hierarchy, and give folders the same name if they are on different hierarchical levels. You can't:

- Delete folders unless they are empty
- Select multiple folders
- Rename, move, or delete the top-level folder. For Forms, the top-level folder is called Library. For Business Rules, the top-level folder is called CalcMgrRules.
- Add files to the top-level folder unless you're an administrator

To create folders:

1. Perform an action:
   - For form folders: Click the **Navigator** icon, and then under **Create and Manage**, click **Forms**.
   - For Calculation Manager business rule folders: Click the **Navigator** icon, and then under **Create and Manage**, click **Rules Security**.
2. Select the folder under which to create the folder.
3. Above the folders list, click **Create**.
4. Enter the folder name.
5. Click **OK**.

Working with Folders

To move, delete, or rename folders:

1. For form folders: Click the **Navigator** icon, and then under **Create and Manage**, click **Forms**.

   For business rule folders: Click the **Navigator** icon, and then under **Create and Manage**, click **Rules Security**.
2. Select the folder to move, delete, or rename.
3. Perform a task:
   - To move, click **Move**. Select the destination folder to which to move the selected folder.

     > **Note:**
     >
     > When you move folders, all nested folders, forms, and Calculation Manager business rules within them are also moved.

   - To delete, click **Delete**.
   - To rename, click **Rename**, and then enter the new name.

4. Click **OK**.

> **Tip:**
>
> Did you know that folder names can be localized based on the user's browser locale? To localize a folder name (for example, from English to Japanese), you can customize the folder name on the **Artifact Labels** page in the **Tools** cluster. For more information:
>
> See Specifying Artifact Labels.

Administering Rules

- About Rules
- Adding and Removing Rules in Forms
- Setting Business Rule Properties
- Viewing Rules Usage
- About Runtime Prompts
- Using Groovy Rules

About Rules

With appropriate access, users can launch business rules from the application. Rules can also prompt users for input when rules are launched.

To launch business rules from within data forms or independently in the application, the rules must be deployed from Calculation Manager. For optimum performance, business rules that run in forms should be designed to execute within three minutes. For rules with a longer execution time, you can schedule batch processing or run the rules during non-peak hours.

You can diagnose performance issues with business rules and pinpoint and address any issues using activity reports. Activity reports identify which business rules are taking the longest to execute. You can then open the rules in Calculation Manager, review the log, and optimize the steps within the rule to improve performance.

For information on:

- Creating, updating, and deploying business rules, see Deploying Business Rules and Business Rulesets from the Deployment View in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*
- Using runtime prompts, see About Runtime Prompts
- Generating a report that details where rules are used, see Viewing Rules Usage
- Viewing an activity report to diagnose rules performance issues, see Viewing Activity Reports
- Optimizing business rules, see Optimizing Business Rules in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*

Troubleshooting

For help with designing rules to avoid common execution errors and optimizing slow rules, see Troubleshooting Business Rule Errors and Performance in *Oracle Enterprise Performance Management Cloud Operations Guide*.

Adding and Removing Rules in Forms

You can associate one or more rules with a form, by cube. Users can launch associated rules from the form to calculate and allocate values. You can set whether each rule associated with a form automatically launches when the form is opened or saved. You can also change the order in which the rules are launched or remove rules from a form.

Note the following:

- Rules must be deployed to the business process from Calculation Manager before you can add rules to business process forms.

  See About Rules.
- Likewise, if a rule is deleted in Calculation Manager, the rules must be redeployed to the business process in order for them to no longer be used in the business process. Optionally, you can manually remove a rule from a form using the procedure in this topic.
- Rules that are set to automatically run when a form is loaded or saved never run in the background.

To add, update, or remove business rules in forms:

1. Take an action:
   - To update the current form, click **Actions**, and then click **Business Rules**.
   - To open a form for editing so you can associate rules, go to the Home page and click the **Navigator** icon , and then under **Create and Manage**, click **Forms**. Select the form, click , and then click **Business Rules**.
2. From the **Cube** drop-down menu, select the cube.
3. From the **Business Rules** list, select the rules to associate with the form, and add them to **Selected Business Rules**. To remove associated rules, select the rules under **Selected Business Rules** and remove them.

   By default, the Calculate Form rule is selected. Calculate Form is automatically created for forms to calculate subtotals. You can remove Calculate Form to prevent users 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.

   > **Caution:**
   >
   > The order in which business rules launch is important and may affect data.
5. To set business rule properties, click **Properties**.

See Setting Business Rule Properties.

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.

Setting Business Rule Properties

You can specify whether business rules associated with forms launch automatically when users load (open) or save the form. If business rules have runtime prompts, you can set whether the default members in the runtime prompt match the members selected in the page and Point of View axes.

To set business rule properties:

1. Take an action:
   - To update the current form, click **Actions**, and then click **Business Rules**.
   - To open a form for editing, go to the Home page and click the **Navigator** icon, and then under **Create and Manage**, click **Forms**. Select the form, click , and then click **Business Rules**.
2. There are four phases in a form where a rule can be assigned and each may have information relevant to that phase. Select from the following:
   - **Run Before Load**
   - **Run After Load**
   - **Run Before Save**
   - **Run After Save**

   > **Note:**
   >
   > - **Run After Load** and **Run Before Save** are enabled only for Groovy rules. You can't select these options for calc scripts or graphical rules. For more information about Groovy rules, see Using Groovy Rules.
   > - Some actions such as saving the grid, changing the page, or launching an action menu item will also reload the page after the operation completes. This ensures the grid contains the newest data. When this happens, the before and after load actions will run as they do in normal page loads.
   > - While hidden runtime prompts are not supported for calc script rules on the before and after load options, they are supported for Groovy rules.
   > - For information about where Groovy rules are supported, see About Creating a Groovy Business Rule.:
3. **Optional**: If a business rule has runtime prompts, select **Use Members on Form** to match the default member selection on the runtime prompt window to the current members in the page and Point of View axes of the open form.

   To learn how this option interacts with other settings and conditions, see Understanding Runtime Prompts.
4. **Optional**: To hide the runtime prompt value from the user, select **Hide Prompt**, which automatically selects **Use Members on Form**.

After saving the form, the next time you return to this page, **Use Members on Form** displays as selected.

You can hide runtime prompts if:

- All runtime prompt member values are filled in (appropriate dimension members can be read from form's Page/Point of View)
- No dimensions are repeated in the runtime prompt

5. Click **OK**.

Viewing Rules Usage

Business rules can be used in the following artifacts:

- Forms
- Rulesets
- Menus
- Task lists

Service Administrators can generate a report, called the Rules Usage Report, that details where rules are used. Filters enable you to refine the details by rule name, rule type, and cube.

> **Note:**
>
> The only rules listed in the Rules Usage Report are those rules that are deployed in the business process.

To view rules usage:

1. Click the **Navigator** icon, and then under **Monitor and Explore**, click **System Reports**.
2. Click the **Rules Details** tab.
3. If filtering, select filter criteria, and then click **Apply Filter**:
   - **Rule Type**: Choose one or more options: **All**, **Rules**, and **Rulesets**
   - **Cube**: Choose from a list of available cubes.
   - **Name Filter**: Enter a rule or ruleset name. Optionally, you can use wildcard characters.

   The **Rules** list displays the rules that are deployed in the business process and that match the filter criteria.
4. For **File Format**, select one of the following formats:
   - **XLSX** (default)
   - **PDF**
   - **HTML**
   - **XML**

> **Note:**
>
> Choosing XML format enables you to configure the Rules Usage Report with the help of Oracle BI Publisher Add-in for MS Office. You can have greater control over which columns to display in your report, the order in which the columns are displayed, whether to apply sorting, and so on.

5. Click **Create Report**.

The Rules Usage Reports provides the following information:

- Rule Name
- Rule Type
- Cube
- Primary Association (Type and Name)
- Secondary Association (Type and Name)

Primary association refers to the rule that is associated with the artifact directly; for example, a rule linked to a form or to a task. Secondary association refers to the rule that is associated with an artifact indirectly; for example, a rule that is linked to a menu item, and the menu item is linked to a form, or a rule that is linked to a task for a given task list.

Rules in the generated report are displayed in alphabetical order as follows:

- Rule name
- Primary association type
- Primary association name

Example 19-1 Sample Rules Usage Report Showing Primary Associations (PDF Format)

ORACLE EPM Cloud

Rules Usage Report

Rule Name	Rule Type	Cube	Primary Association		Secondary Association	
			Type	Name	Type	Name
Act_Agg_Ek	Rules	PSPPlan1				
Adhoc - Clear 5+7	Rules	PSPPlan1				
Agg	Rules	PSPPlan1	Form	_XX_Agg		
Agg - Roll up E and O from drop down, Customer in row, all others at none	Rules	PSPPlan1	Form	1 - Corp Prof Fees		

Page 1 of 82

1/10/2018 18:00 PM

Example 19-2 Sample Rules Usage Report Showing Primary and Secondary Associations (PDF Format)

ORACLE EPM Cloud

Rule Name	Rule Type	Cube	Primary Association		Secondary Association	
			Type	Name	Type	Name
psp_Agg_budget	Rules	PSPPlan1				
psp_Agg_Fcst_All Dims	Rules	PSPPlan1	Task	Aggregation	Task List	Execute Royalty Calc
psp_Agg_FC@AER_All Dims	Rules	PSPPlan1				
psp_Agg_budget_All Dims	Rules	PSPPlan1				
psp_Rowfocused_Agg	Rules	PSPPlan1	Form	1.0 psp_Agg		
psp_Agg_Actuals_All Dims	Rules	PSPPlan1				

Page 81 of 82

1/10/2018 18:00 PM

About Runtime Prompts

When launched, business rules can prompt users for such variables as members, text, dates, or numbers. Prompts should be specific and tell users what type of data is expected. For example:

- Select a month.
- Enter the expected number of customer visits per quarter.
- What percentage change in earnings do you expect next month?

If, in Calculation Manager, the option **Create dynamic members** is selected for the business rule and the parent member is enabled for adding dynamic children, then users can create new members by entering their name in the runtime prompt.

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

Understanding Runtime Prompts

The display and values of runtime prompts are affected by such aspects as:

- Whether the **Use as Override Value** property is set at the rule or the ruleset level at design-time
- Whether there are valid members on the form's Page/Point of View and whether the **Use Members on Form** and **Hide Prompt** options on the **Business Rule Properties** tab are selected (see Setting Business Rule Properties)
- Whether the **Runtime Prompt** option is set during form design or set when designing the runtime prompt (see *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*)
- Whether the **Use Last Value** property is set when designing the business rule
- Whether, in Calculation Manager, the option **Create dynamic members** is selected for the business rule and the parent member is enabled for adding dynamic children. If yes, then users can create new members by entering their name in the runtime prompt.

> **Note:**
>
> In the business process, runtime prompt values aren't replaced inside double quoted text. Alternately, you can use Groovy rules to achieve the same. However, launching a rule from Calculation Manager will replace runtime prompt values in double quoted text.

Principles:

1. If the **Use as Override Value** property is set at the rule or the ruleset level at design-time, the value overridden at the rule level or the ruleset level will take precedence over the values of members in the Page/Point of View and the last saved value. This occurs regardless of where the rule is launched (from the form or from the **Rules** link on the **Navigator** menu), and regardless of whether the runtime prompt is hidden during design.

The **Override Value** can be set as a user variable, in which case the rule will be launched with the current value of the variable.

2. When launched from a form, the values of members in the Page/Point of View take precedence over the last saved value if the **Use Members on Form** option is selected, regardless of whether the runtime prompt is hidden during design. The business rule is run without displaying the hidden runtime prompt to users, and the runtime prompt values are taken from the Page/Point of View members.
3. If the **Use Last Value** option is selected for the runtime prompt at design time, and if any of these conditions exist:
   - **Use Members on Form** isn't selected
   - A runtime prompt is launched from the **Rules** link on the **Navigator** menu
   - Values can't be pre-filled from the context

   Then the precedence of runtime prompt values is determined by:

   a. The last saved value takes precedence.
   b. If a ruleset is launched, the value overridden at the ruleset level at design-time is used.
   c. If a business rule is launched, the value overridden at the rule-level at design-time is used. If it's not overridden at the rule-level, the runtime prompt value at design-time is used.

   Runtime prompts that are hidden at design time never use the last saved value. In these cases, the **Use Last Value** setting is ignored.
4. The **Use Members on Form** and **Hide Prompt** options apply only to Member and Cross Dimension runtime prompt types (Cross Dimension runtime prompt types are available only for business rules created with Calculation Manager).

   The value set at the rule or the ruleset level at design-time when the **Use as Override Value** property is set participates in the **Hide Prompt** behavior.
5. For Cross Dimension runtime prompts: the runtime prompt isn't hidden unless all the prompts in the runtime prompt can be pre-filled from the **Override Value** or Page/Point of View. The runtime prompt is displayed with some values pre-filled from the **Override Value** or Page/Point of View and others filled according to Principles 1, 2 and 3.

This table describes the result on runtime prompts of these settings and conditions:

Table 19-24 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.

Table 19-24 (Cont.) 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	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 can't be pre-filled from the **Override Value** or Page/Point of View context, then all runtime prompts display, with values pre-filled wherever possible. All others follow Principles 1 and 3.
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 isn't set and the member isn't 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 can't be passed because the dimension of the runtime prompt is on rows or columns, so the **Hide Prompt** setting is ignored and the runtime prompt displayed.
Use as Override Value isn't set and the member isn't available on the Page/Point of View to use as the runtime prompt value.	Yes	No	Yes	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 isn't 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 can't 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 isn't set and the member isn't 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.

Table 19-24 (Cont.) 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, and the member isn't 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 isn't set and the member isn't 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 isn't set and the member isn't 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 can't be passed in (because the dimension is on the row or column, for example), hidden runtime prompts are displayed.
- With hidden Cross Dimension runtime prompt types, if all prompts can't be passed in from the Override Value or context, the runtime prompt displays with values pre-filled from **Override Value** or context values and design time values. For example, if the Cross Dimension has runtime prompts for Period, Entity, and Scenario, and Entity is defined on the row and Scenario has an **Override Value** set, then the runtime prompt displays with the Override Scenario, then design time value for Entity, then Page Period.
- If the **Override Value** is present, or context can be passed in for the runtime prompt value but it's out of limits, then the runtime prompt is displayed with the context value pre-filled.
- If there is more than one runtime prompt of type Member or type Cross Dimension combined, then the runtime prompts are displayed with the **Override Value** or context values pre-filled. For example, if there is a member type runtime prompt for the Entity dimension and a Cross Dimension type runtime prompt with one prompt for the Entity dimension, then both runtime prompts are displayed. This rule doesn't apply to Calculation Manager rulesets.
- When launched from the **Rules** link on the **Navigator** menu, runtime prompts are hidden and the design-time value (overridden at the rule or ruleset level) is used to launch the business rule. If the provided design-time value is out of limits, then the runtime prompt is displayed with the design-time value pre-filled.

- Runtime variables that are hidden during design never use the last saved value. The **Use Last Value** property is ignored, and the values are not saved to the database.

Using Groovy Rules

Create rules written in the Groovy scripting language that solve use cases that normal business rules can't solve.

Table 19-25 Where Can I Learn More About Groovy Rules?

Your Goal	Learn More
Learn the benefits of Groovy rules.	About Creating a Groovy Business Rule
View examples of business scenarios where Groovy business rules might be used.	Groovy Rule Business Scenarios
Watch videos and complete hands-on tutorials that teach best practices when implementing and using Groovy rules.	• Groovy Business Rule Tutorial Videos • Learning Groovy in Oracle EPM Cloud
Create Groovy business rules and a Groovy template using Calculation Manager.	• Creating a Groovy Business Rule • Creating a Groovy Template for a Planning BSO Cube
Connect to the Java APIs used for creating Groovy rules and view example Groovy scripts.	• Java API Reference for Groovy Rules • Groovy Business Rule Examples
Edit the script for a Groovy business rule or template using Calculation Manager.	Editing the Script For a Groovy Business Rule or Template
Secure Groovy templates for authorized users.	Assigning Access to Groovy Templates

About Creating a Groovy Business Rule

> **Note:**
>
> You can use Groovy rules only for applications of type "Enterprise" (available with Enterprise PBCS or PBCS Plus One licenses), Strategic Workforce Planning, Sales Planning, and FreeForm.

Groovy business rules allow you to design sophisticated rules that solve use cases that normal business rules can't solve; for example, rules to prevent users from saving data on forms if the data value is above a predefined threshold.

You create Groovy rules in Calculation Manager and execute them from any place that you can execute a calc script rule in a business process; for example, on the Rules page, within the context of a form, in the job scheduler, in dashboards, in task lists, and so on. Groovy rules are also supported in rulesets. You can have a combination of calc script rules and Groovy rules within a ruleset.

Oracle supports two types of Groovy rules:

- Rules that can dynamically generate calc scripts at runtime based on context other than the runtime prompts and return the calc script which is then executed against Oracle Essbase.

  For example, you could create a rule to calculate expenses for projects only for the duration (start and end dates) of the project.

Another example is a trend-based calculation that restricts the calculation to the accounts available on the form. You could use this calculation for various forms in Revenue, Expense, Balance Sheet, and Cash Flow. This allows for optimization and reuse.

- Pure Groovy rules that can, for example, perform data validations and cancel the operation if the data entered violates company policies.

Watch this tutorial video to learn best practices in moving modified data using Groovy Rules and Smart Push.

 Moving Modified Data Using Groovy Rules and Smart Push

Watch this tutorial video to learn how to improve calculation performance on business process forms by creating context-specific, dynamic business rules using the Groovy scripting language.

 Calculating Modified Data Using Groovy Rules

> **Note:**
>
> For Enterprise applications, a Java API reference is available to use as you create Groovy rules. To view the technical reference for designing Groovy rules, see the Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules on the cloud help center. You can also access this reference from the Oracle Enterprise Planning and Budgeting Cloud Academy. To access the Academy, sign in, and then click **Academy**.
> The Java API reference includes examples that demonstrate the syntax and power of the EPM Groovy object model. To view examples, open the Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules. Then take an action:
>
> - On the main page, scroll down to **Example Groovy Scripts**, and then click the word **here** in the sentence about sample Groovy scripts.
> - In the left pane, under **All Classes**, click the **StrategicModel** Class.

Java API Reference for Groovy Rules

For Enterprise applications, a Java API Reference is available to use as you create Groovy rules.

The Java API Reference includes examples that demonstrate the syntax and powers of the EPM Groovy object model.

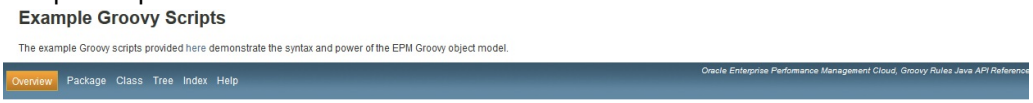
To view the Java API Reference, see the Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules on the cloud help center. You can also access this reference from the Oracle Enterprise Planning and Budgeting Cloud Academy. To access the Academy, sign in, and then click **Academy**.

Groovy Business Rule Examples

Example Groovy scripts are available.

To see example Groovy scripts:

1. See the Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules, http://docs.oracle.com/cloud/latest/epm-common/GROOV/.
2. Do one of the following:
   - Under **Example Groovy Scripts** on the main page, click the word "here" to view sample scripts:

   - Under **All Classes** in the left pane, click a class to see the examples for that class.

     For example, to see Strategic Modeling examples, click the StrategicModel class in the left pane.

Groovy Business Rule Tutorial Videos

Watch these tutorial videos for details and best practices when implementing and using Groovy Business Rules.

Your Goal	Watch this Video
See the training options for creating Groovy rules in Oracle Enterprise Performance Management Cloud:	Learning Groovy in Oracle EPM Cloud
Push data from a source location to a target location using Groovy rules and Smart Push.	Moving Modified Data Using Groovy Rules and Smart Push
Generate focused calculation scripts in the business process to calculate only data that has been edited, instead of the entire data entry form.	Calculating Modified Data Using Groovy Rules
Use Groovy rules to calculate incrementally loaded data in Data Management.	Calculating Incrementally Loaded Data in Data Management Using Groovy Rules
Use Groovy templates to improve usability and calculation performance for user actions.	Customizing Actions to Improve Performance Using Groovy Templates

Groovy Rule Business Scenarios

This section provides examples of business scenarios where you can use Groovy business rules.

These topics contain scenarios that show you how to use Groovy rules to perform focused data movement and smart calculations on business process data:

- Moving Modified Data Using Groovy Rules and Smart Push
- Calculating Modified Data Using Groovy Rules

Moving Modified Data Using Groovy Rules and Smart Push

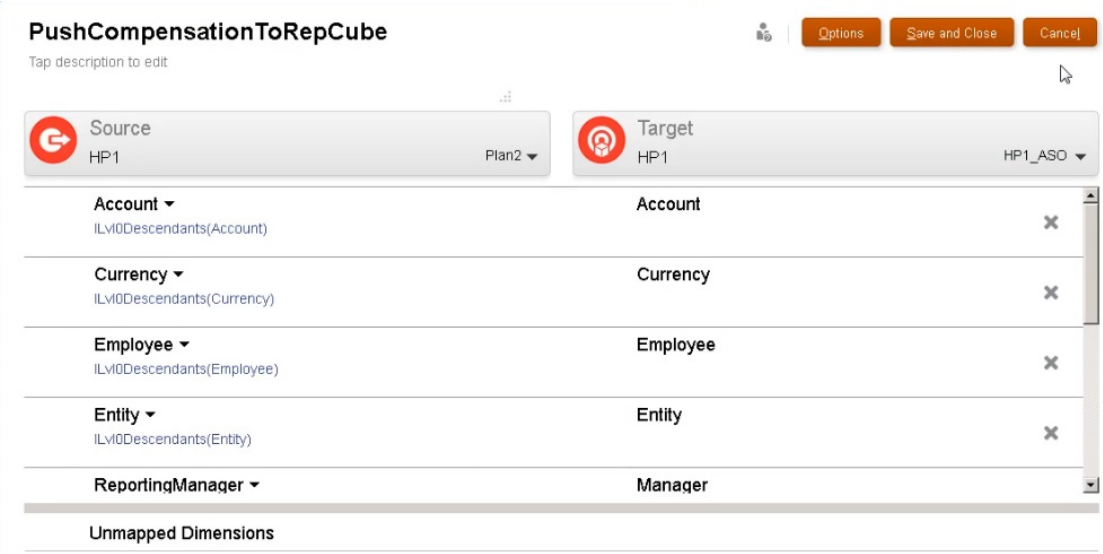
In this topic, we'll show you best practices on how you can use Groovy rules and Smart Push to move modified data into a reporting cube.

On a data form, users can modify employee salary and assign a Reporting Manager. After the form data is saved, Smart Push is run to copy data to the reporting cube for analysis. Then, dimensions are also consolidated, if necessary, as the data is being copied.

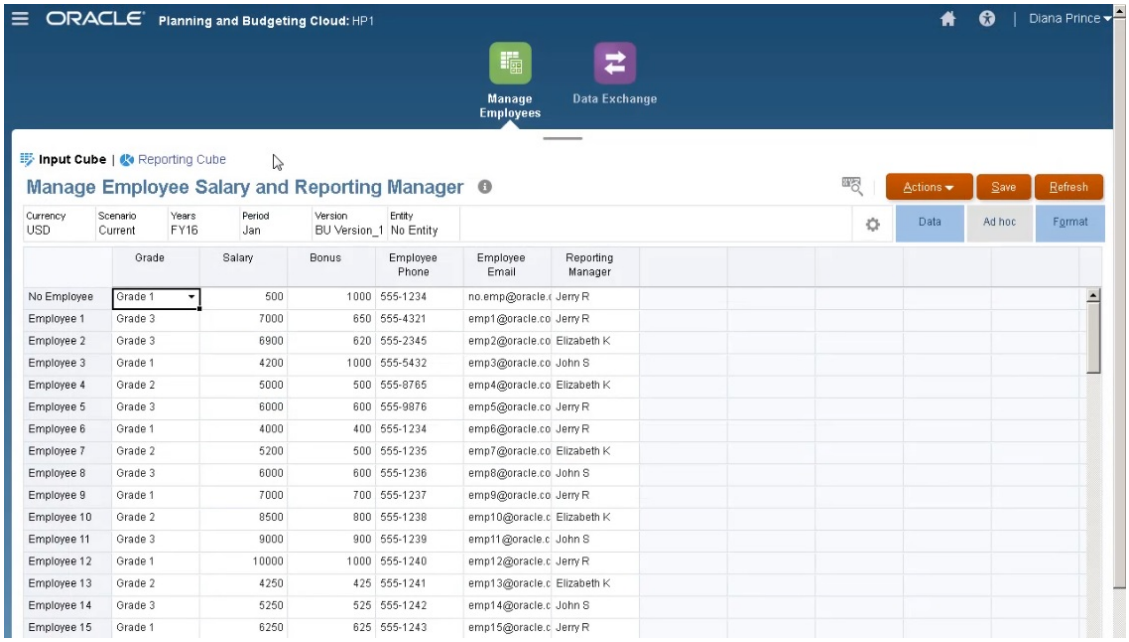
If the form contains hundreds of rows of data, all of that data is checked against the configured data map, processed for dimension consolidation, if any is set, and then pushed to the reporting cube. Depending on the amount of data on your form, this process may take a while to complete.

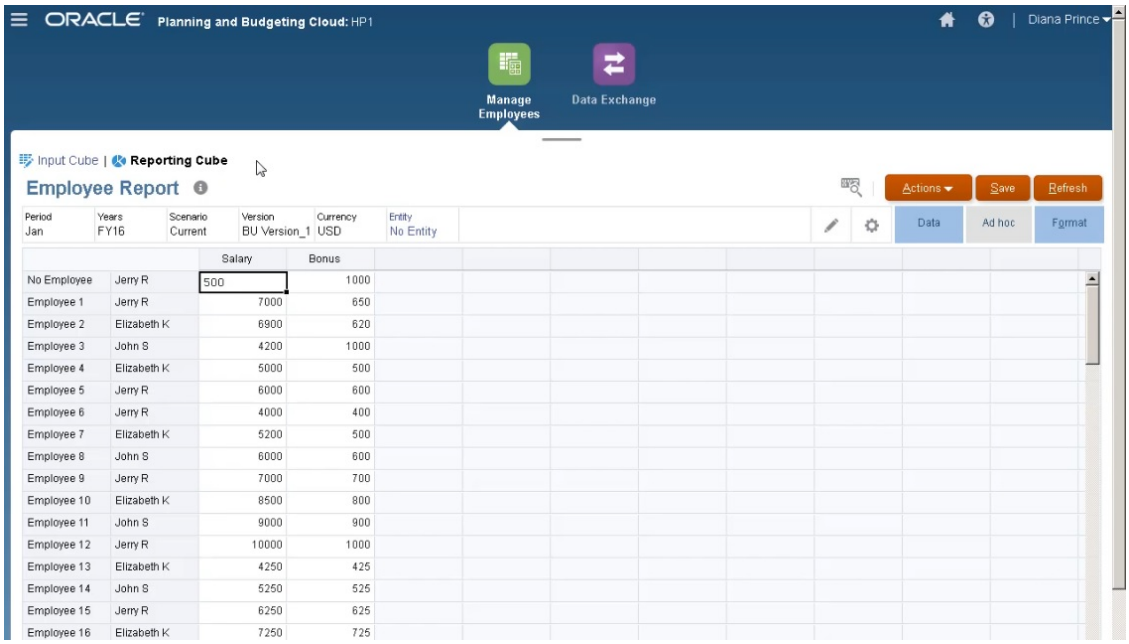
You can improve the performance of data synchronization between your input and reporting cubes by designing a Groovy rule that identifies and isolates the data you modified, and pushes only the isolated data into your reporting cube.

For example, we configured a simple data map for Smart Push that maps dimension members from our input cube to our reporting cube.

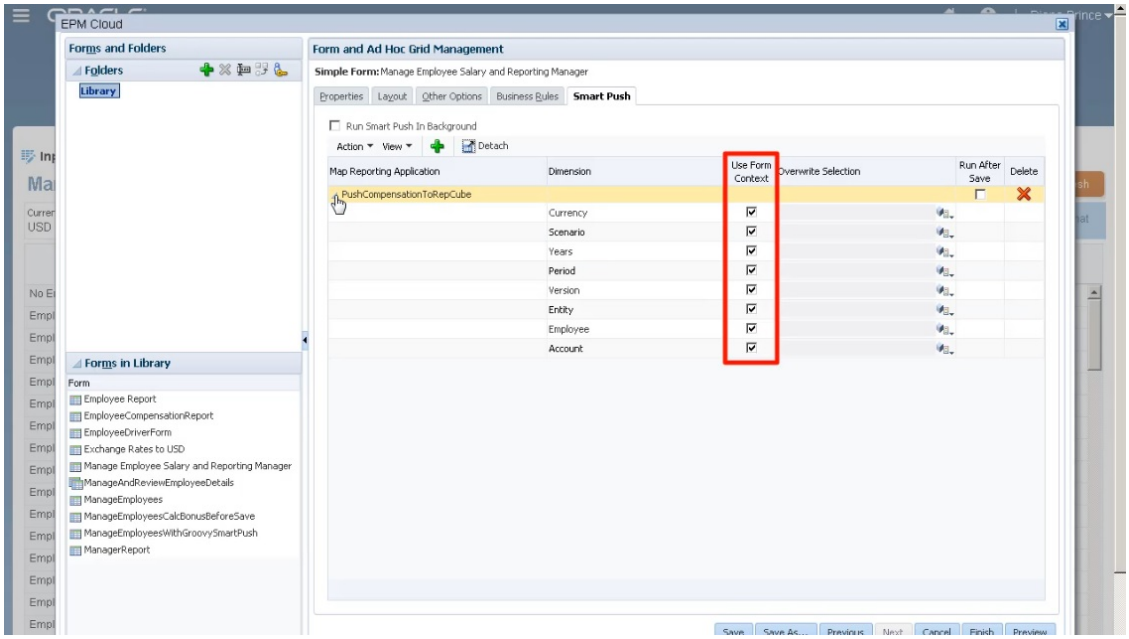


Then we prepared forms that display data from our input and reporting cubes.

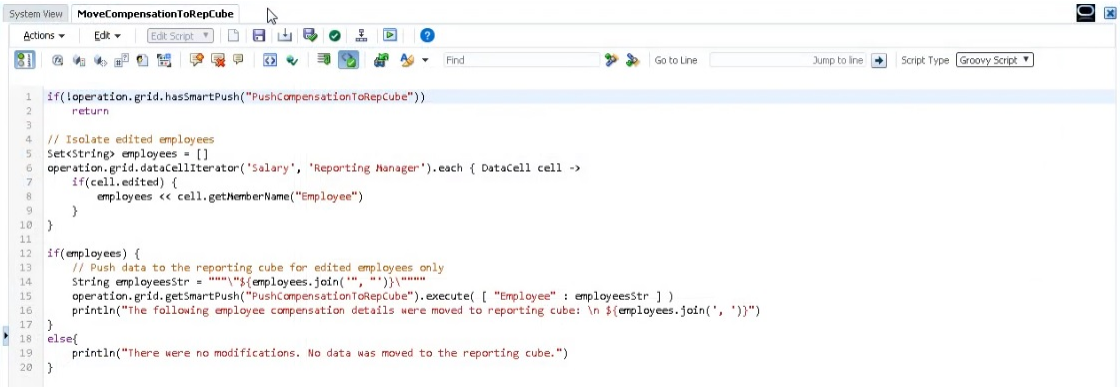




The form used for data entry has Smart Push enabled and uses the form's context to push data for all the members selected for the dimensions defined on the form.



For the Groovy rule, here's the entire script.



The script starts by identifying the associated data map for smart push.

```
if(!operation.grid.hasSmartPush("PushCompensationToRepCube"))
    return

// Isolate edited employees
Set<String> employees = []
operation.grid.dataCellIterator('Salary', 'Reporting Manager').each { DataCell cell ->
    if(cell.edited) {
        employees << cell.getMemberName("Employee")
    }
}

if(employees) {
    // Push data to the reporting cube for edited employees only
    String employeesStr = """\"${employees.join('", "')}\"""
    operation.grid.getSmartPush("PushCompensationToRepCube").execute( [ "Employee" : employeesStr ] )
    println("The following employee compensation details were moved to reporting cube: \n ${employees.join(', ')}")
}
else{
    println("There were no modifications. No data was moved to the reporting cube.")
}
```

This section of the script uses the `dataCellIterator` method to identify edited cells and isolates them.

```
if(!operation.grid.hasSmartPush("PushCompensationToRepCube"))
    return

// Isolate edited employees
Set<String> employees = []
operation.grid.dataCellIterator('Salary', 'Reporting Manager').each { DataCell cell ->
    if(cell.edited) {
        employees << cell.getMemberName("Employee")
    }
}

if(employees) {
    // Push data to the reporting cube for edited employees only
    String employeesStr = """\"${employees.join('", "')}\""""
    operation.grid.getSmartPush("PushCompensationToRepCube").execute( [ "Employee" : employeesStr ] )
    println("The following employee compensation details were moved to reporting cube: \n ${employees.join(', ')}")
}
else{
    println("There were no modifications. No data was moved to the reporting cube.")
}
```

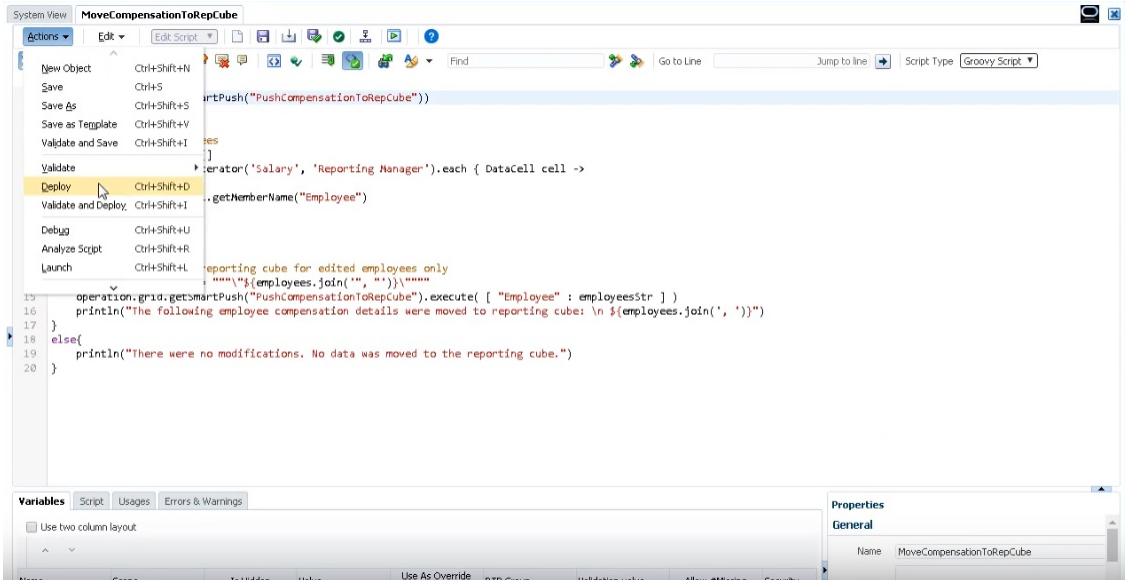
Then, this section takes the isolated data and runs the focused data movement using Smart Push and the data map configured and associated with the form. A message displays when modified data is moved to the reporting cube.

```
if(!operation.grid.hasSmartPush("PushCompensationToRepCube"))
    return

// Isolate edited employees
Set<String> employees = []
operation.grid.dataCellIterator('Salary', 'Reporting Manager').each { DataCell cell ->
    if(cell.edited) {
        employees << cell.getMemberName("Employee")
    }
}

if(employees) {
    // Push data to the reporting cube for edited employees only
    String employeesStr = """\"${employees.join('", "')}\""""
    operation.grid.getSmartPush("PushCompensationToRepCube").execute( [ "Employee" : employeesStr ] )
    println("The following employee compensation details were moved to reporting cube: \n ${employees.join(', ')}")
}
else{
    println("There were no modifications. No data was moved to the reporting cube.")
}
```

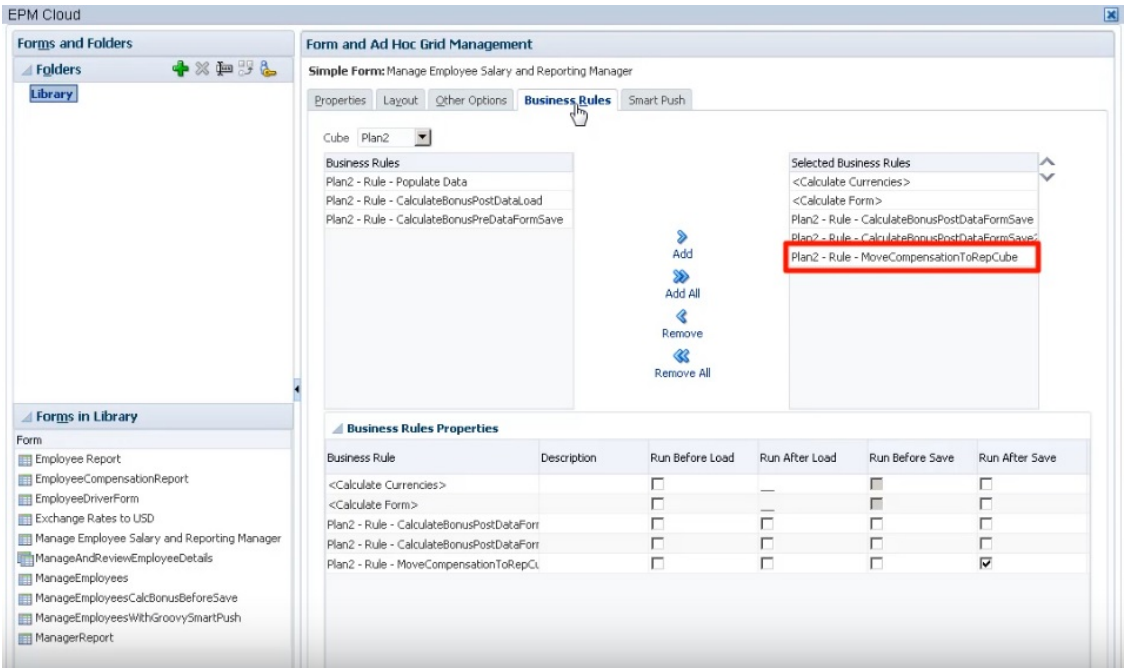
If there are no modifications, a message displays when there are no changes and data was not moved to the reporting cube.

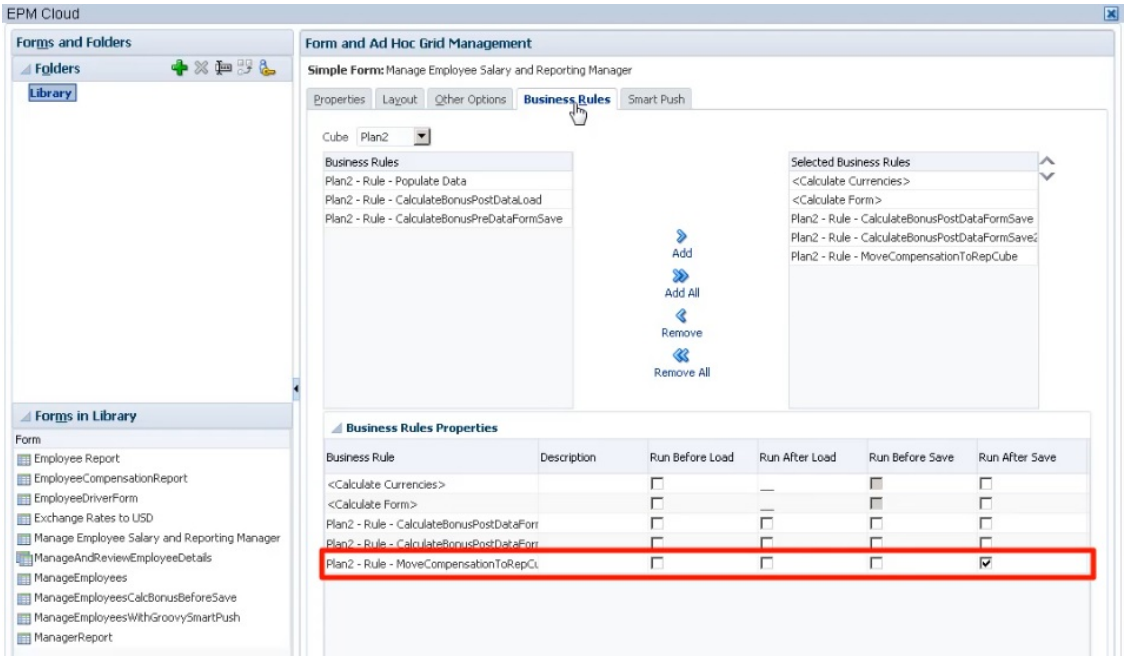
```
if(!operation.grid.hasSmartPush("PushCompensationToRepCube"))
    return

// Isolate edited employees
Set<String> employees = []
operation.grid.dataCellIterator('Salary', 'Reporting Manager').each { DataCell cell ->
    if(cell.edited) {
        employees << cell.getMemberName("Employee")
    }
}

if(employees) {
    // Push data to the reporting cube for edited employees only
    String employeesStr = """\"${employees.join('", "')}\""""
    operation.grid.getSmartPush("PushCompensationToRepCube").execute( [ "Employee" : employeesStr ] )
    println("The following employee compensation details were moved to reporting cube: \n ${employees.join(', ')}")
}
else{
    println("There were no modifications. No data was moved to the reporting cube.")
}
```

After we created this Groovy rule, we deployed it the same way we deploy business rules.

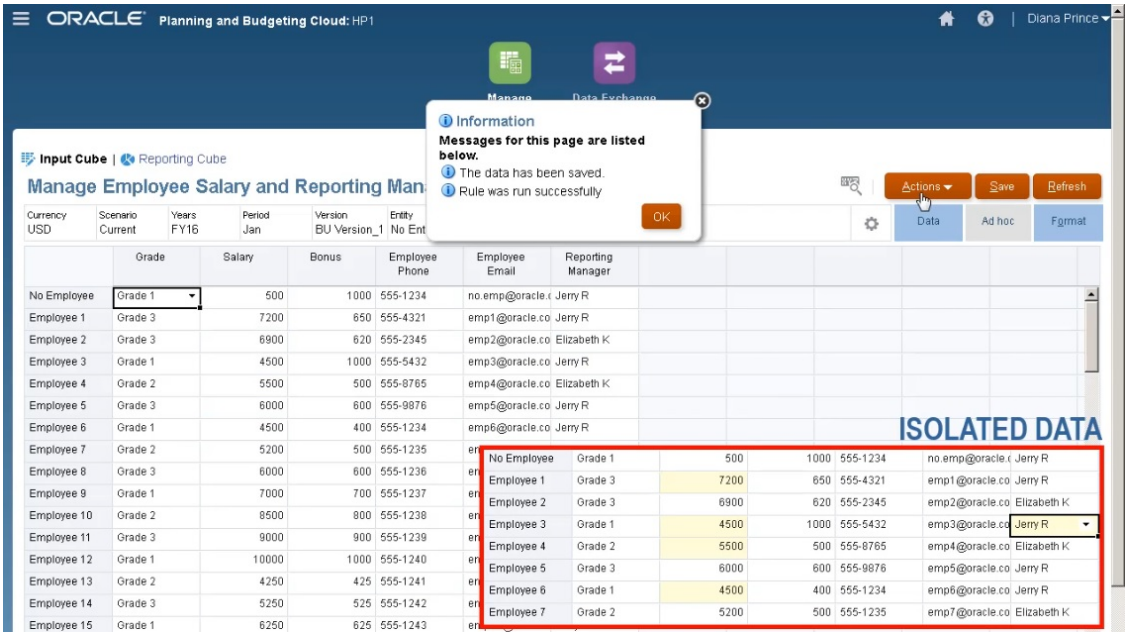


Then, in form management, associated it with the form. The Groovy rule runs after the form data is saved.





To run the Groovy rule, open the form in **Input Cube**, make some changes, and then click **Save**.



Once saved, the updated data will display in the reporting cube.

Calculating Modified Data Using Groovy Rules

In this topic, we'll show you how to improve business process calculation performance by using Groovy rules to calculate only the data that's been modified.

After you update data in a form, it needs to be calculated. When you use a business rule to calculate a standard calculation script, all the data for the form is calculated. With large forms or calculations that affect large areas of the database, you might see slower calculation performance.

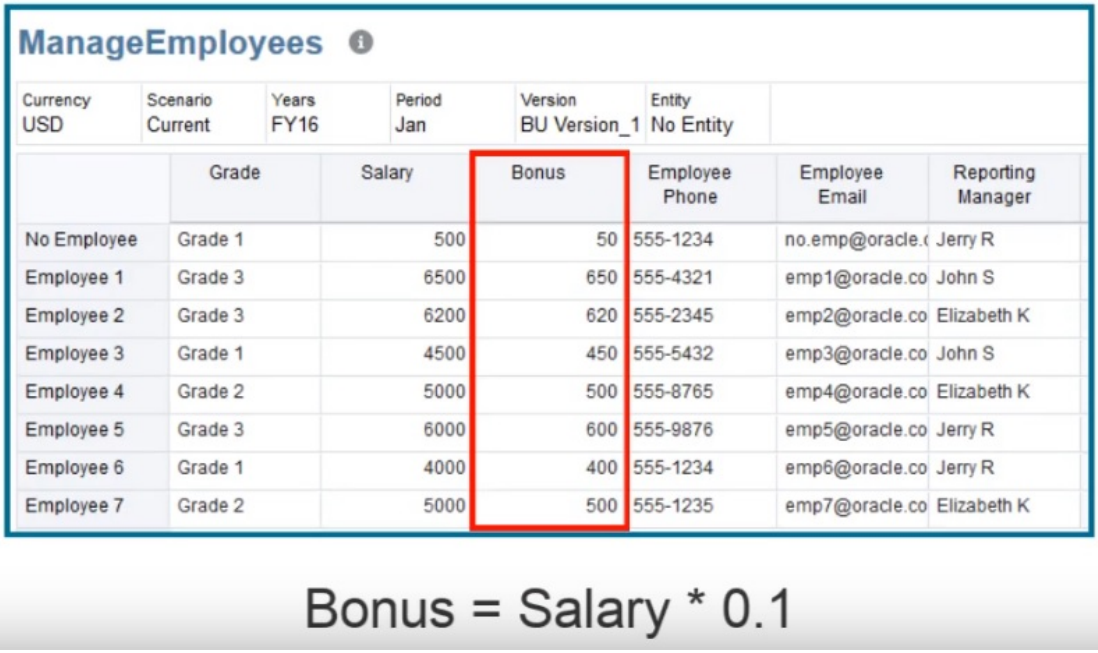
You can manually calculate one row at a time by right-clicking on the row. This can provide faster calculation performance, but isn't a good option when making changes to multiple rows of data.

Groovy rules can dynamically focus on the cells that you update.

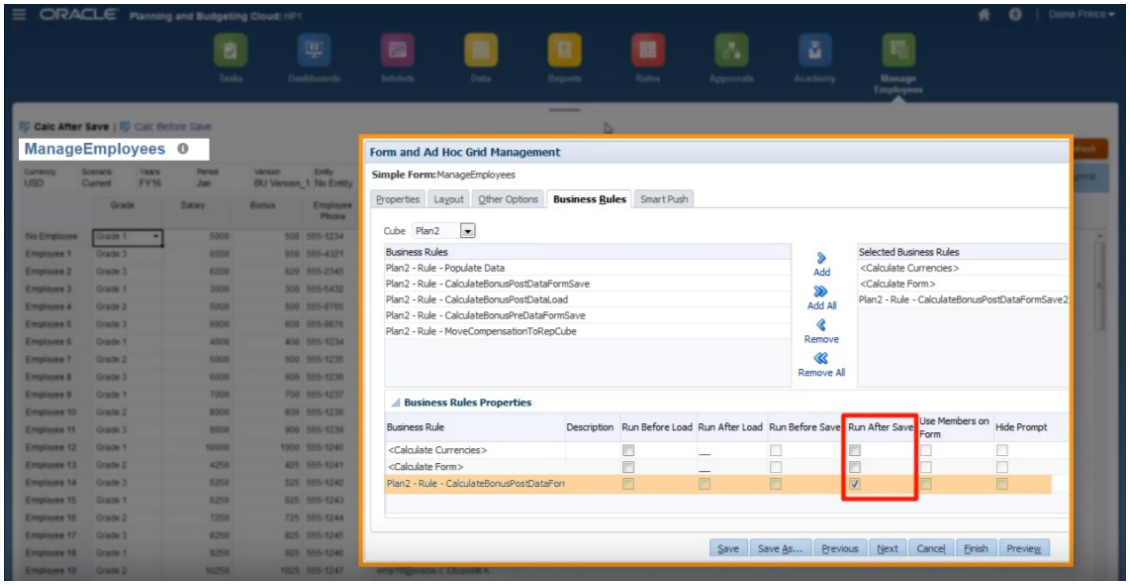
You can use Groovy rules on forms in two ways:

- When you calculate data on save, the Groovy rule generates a calculation script with a FIX statement based on the cells you updated. Then the calculation is run on the server against the modified data instead of all records on the form. As a best practice, use this option with larger forms where you are updating a lot of data.
- When you calculate data before saving, the Groovy rule calculates modified data in memory and displays the results on the form before saving them to the database. As a best practice, use this option when you have a small number of calculations to perform, or when you want to perform complex procedural calculations that are difficult to build into the outline.

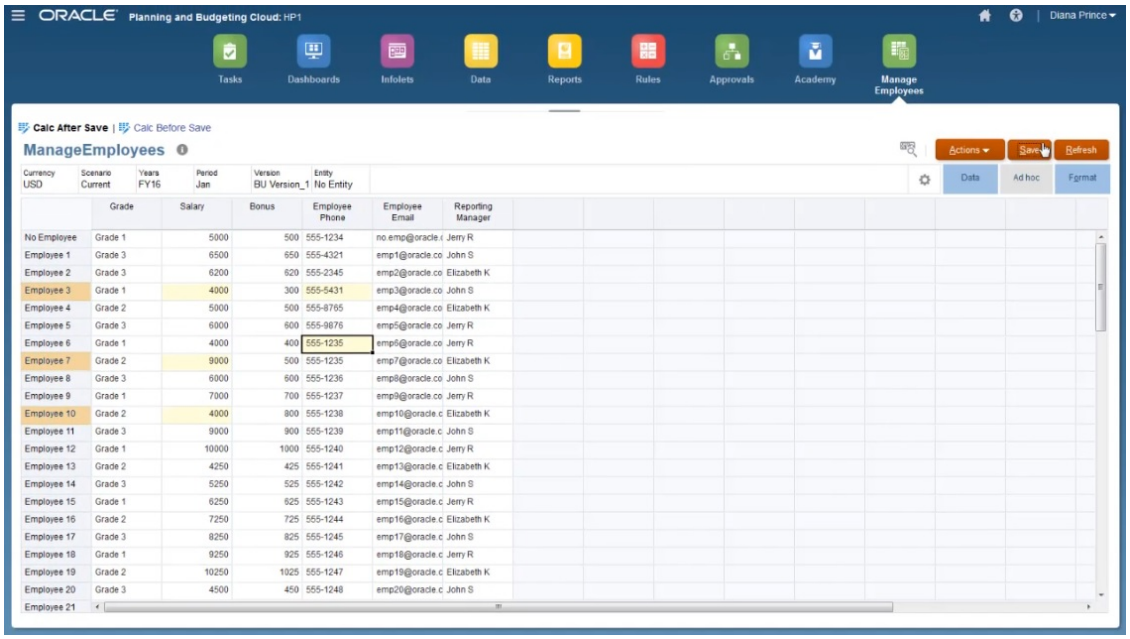
In this example, bonus calculations are calculated as a percent of salary.



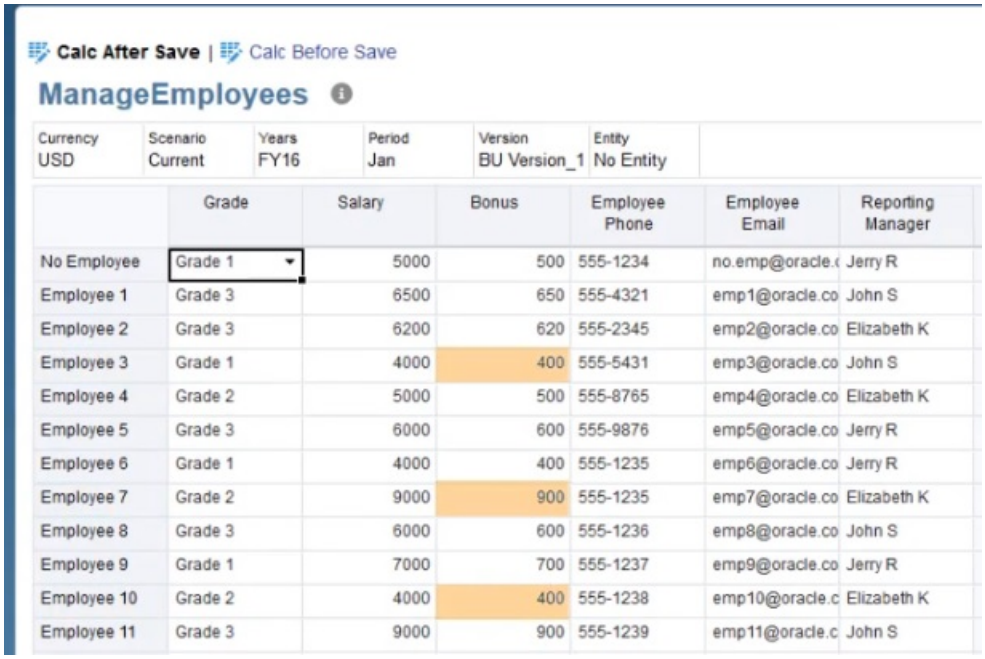
Let's look at calculating on save. Notice that this form is defined with a Groovy rule to calculate on save.



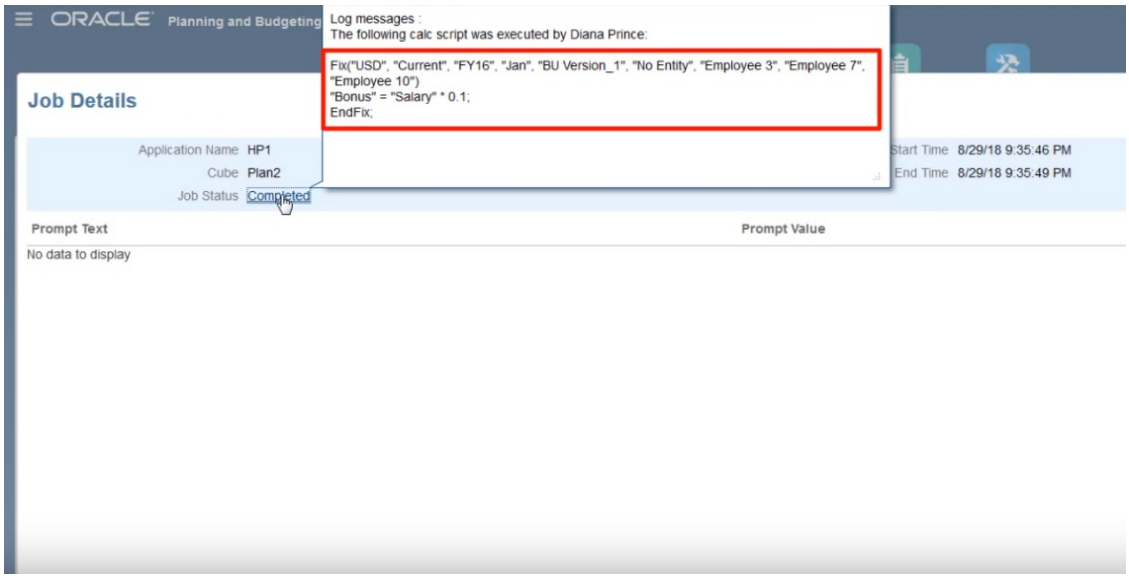
After updating some of the salary data and some of the phone numbers on the form, the phone numbers are saved but don't affect the calculation.



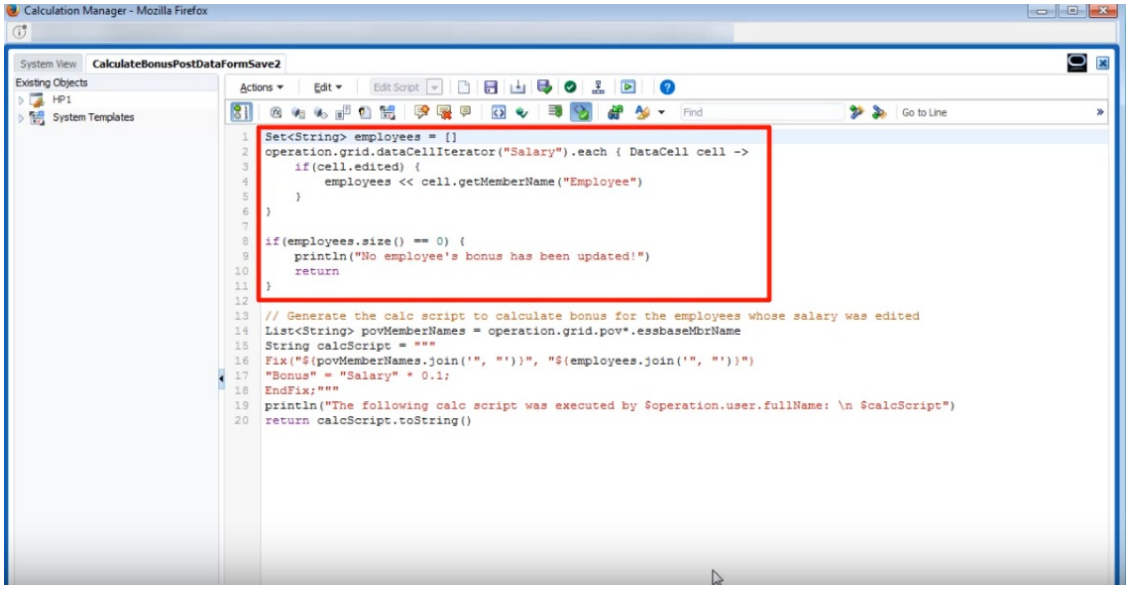
When you save the form to run the associated Groovy rule, the calculated bonus data is displayed on the form.



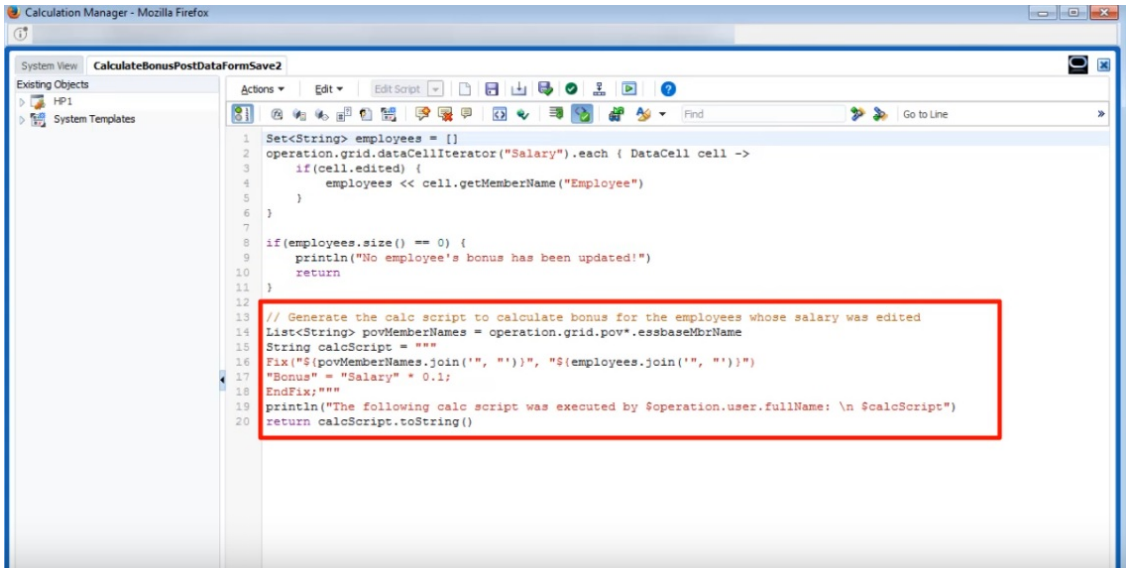
Viewing the job details, here's the calculation script that the Groovy rule created. Notice that the FIX statement includes only the modified employees.



Let's open Calculation Manager to see the Groovy rule. This first section iterates over the grid in the form to find Salary cells that have been edited and collects the related employee names.

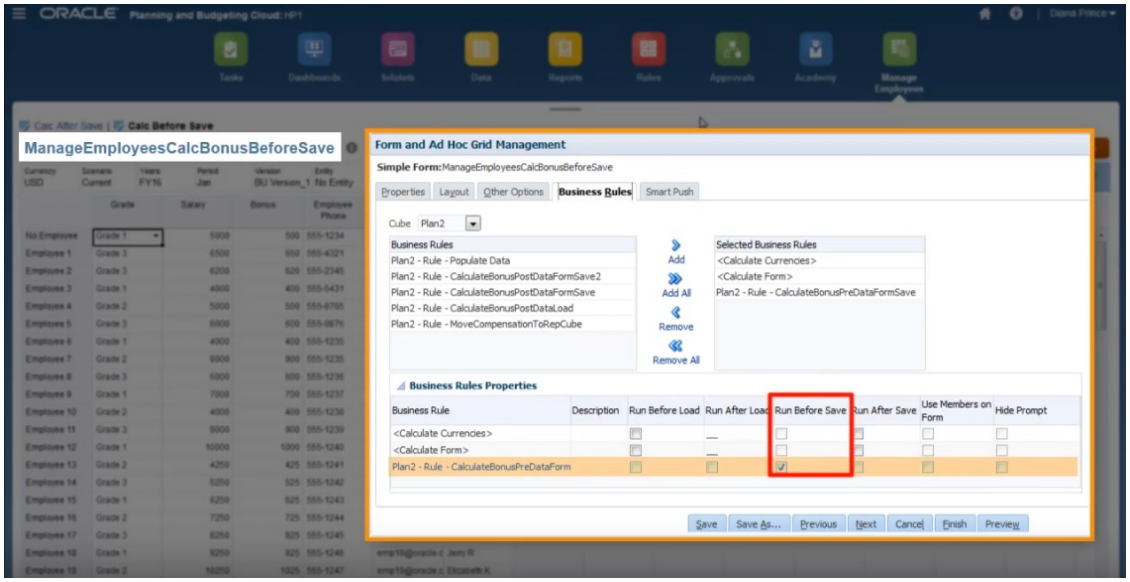


This section generates the calculation script; it finds the POV members from the form, then creates a FIX statement on the POV members and the edited employees.

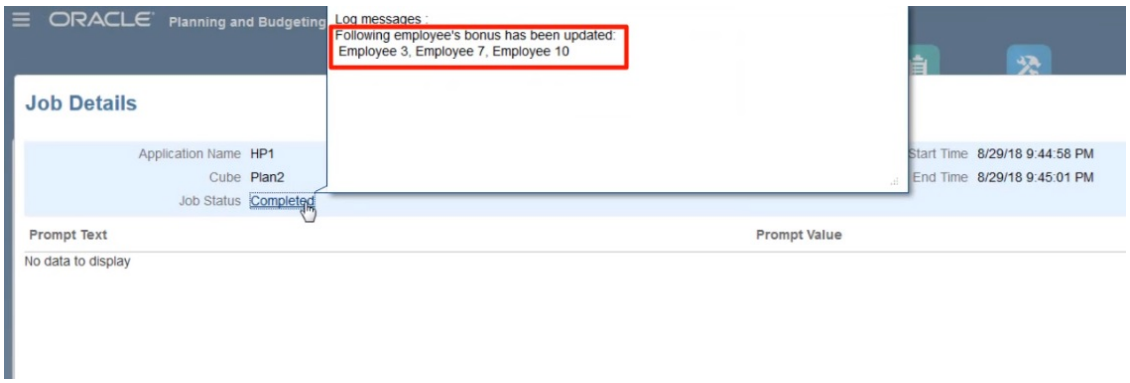


Now let's take a look at calculating before save.

This is the same form, but now it's defined with a different Groovy rule to calculate before save.

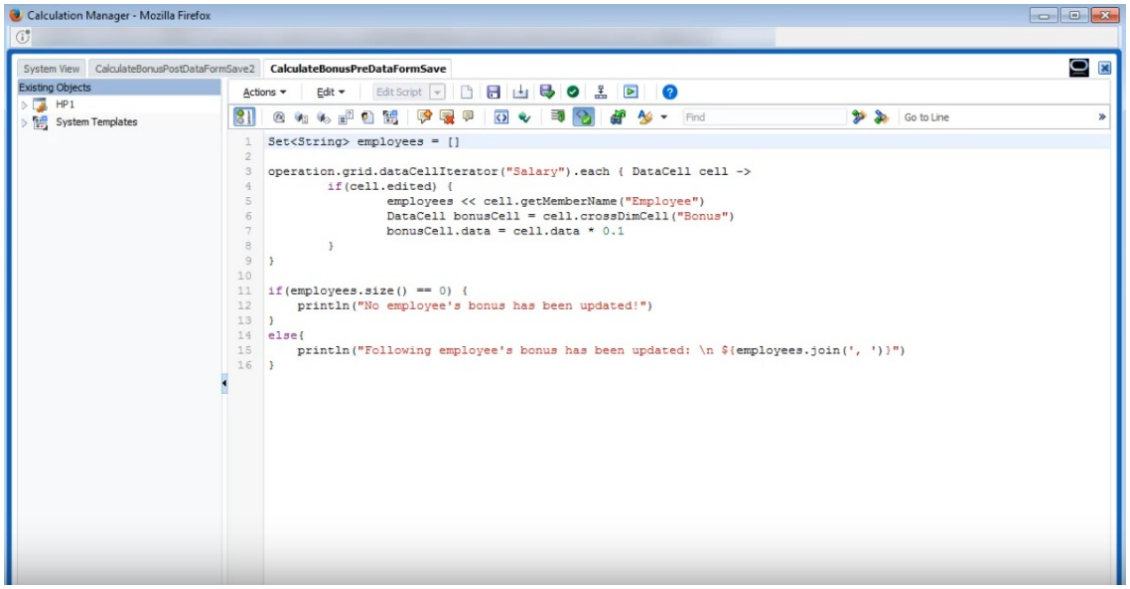


After updating the salary data and some of the phone numbers again, save the form to run the associated Groovy rule. The calculated bonus data is displayed on the form, but when you look at the job details, you can see that some of the bonus data was updated, but this time no calculation script was generated.

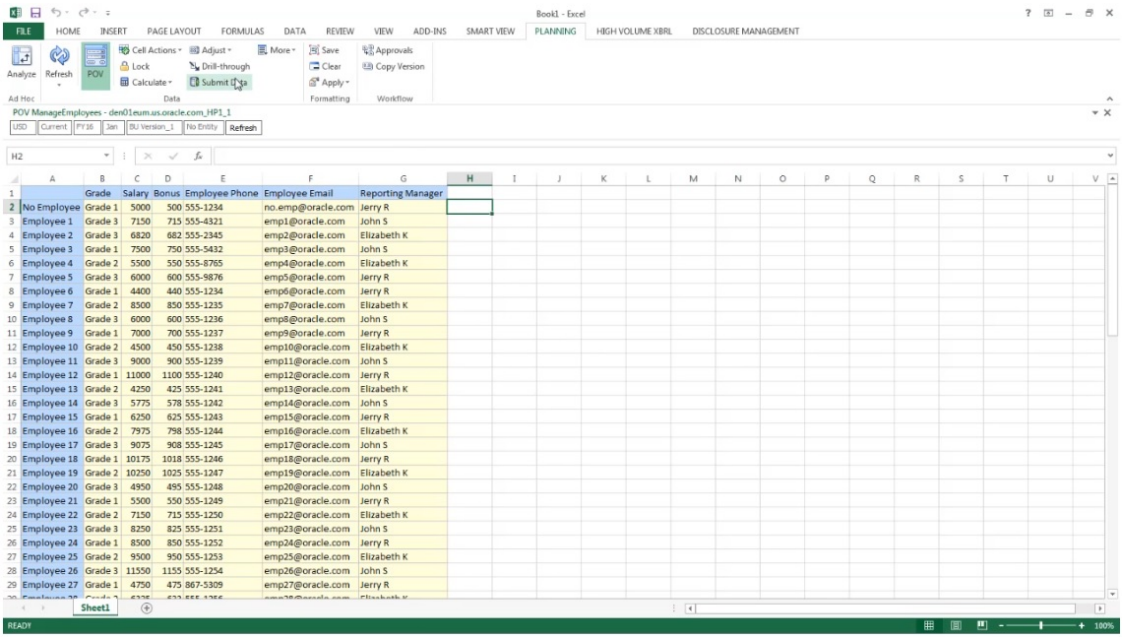


Open the Groovy rule in Calculation Manager.

Similar to the first rule, the script iterates over the grid in the form to find Salary cells that have been edited and collects the related employee names. Then it calculates bonus data for just the edited employees. Notice that no Oracle Essbase calculation script is generated.



Groovy rules also work with FreeForm forms in Oracle Smart View for Office. Smart View gives you all the power of Microsoft Excel to make mass adjustments your data, and any Groovy rules for the form set to run before or after saving are executed when you submit data.



Administering Rules Security

Related Topics

- Assigning Access to Rules
- Adding, Editing, and Removing Access to Rules
- Assigning Access to Groovy Templates

Assigning Access to Rules

Service Administrators can assign access to business rules.

To assign access to Groovy business rule templates, see Assigning Access to Groovy Templates.

To assign access to rules:

1. Click the **Navigator** icon, and then under Create and Manage, click **Rules Security**.
2. Under **Business Rule Folders**, select the folder containing the rules, and then select the rules.
3. Click .
4. Add, edit, or remove access.

   See Adding, Editing, and Removing Access to Rules.

Adding, Editing, and Removing Access to Rules

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

> **Note:**
>
> Rulesets inherit launch permissions from the rules included in the ruleset.

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

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Rules Security**.
2. Under **Business Rule Folders**, select the folder containing the rules, and then select the rules.
3. Click .
4. Perform a task:
   - To add access, click , and then select from the list of available users and groups.

     For **Type of Access**:
     - Select **Launch** to allow the selected users and groups to launch the selected rules.
     - Select **No Launch** to prevent the selected users and groups from launching the selected rules.
   - To edit access, click , and then select the applicable **Type of Access**.
   - To remove access, select the users or groups for which to remove access, and then click .

Assigning Access to Groovy Templates

To assign access to Groovy templates:

1. From the Home page, click **Rules**, then **Filter**, and then select a cube and artifact type.
2. Next to the rule or template, click **Permission**, and then click **Assign Permission**.
3. In **Assign Permission**, click **Permission**, and then assign permissions. See Setting Up Access Permissions for more information.

Administering Smart Lists

Smart Lists are custom drop-down lists that users access from form cells.

Related Topics

- Working with Smart Lists
- Synchronizing Smart Lists in Reporting Applications
- Setting Smart List Properties
- Defining Smart List Entries
- Previewing Smart Lists
- Displaying #MISSING with Smart Lists

Working with Smart Lists

Service Administrators use Smart Lists to create custom drop-down lists that users access from form cells. When clicking in cells whose members are associated with a Smart List (as a member property), users select items from drop-down lists instead of entering data. Users can't type in cells that contain Smart Lists. Smart Lists display in cells as down arrows that expand when users click into the cells.

Perform these tasks to create and administer Smart Lists:

- Define Smart Lists.
- Associate Smart Lists with members.
- Select dimensions for which Smart Lists are displayed.
- Optionally:
  - Use Smart List values in member formulas.
  - Set how #MISSING cells associated with Smart Lists display in forms.
  - Synchronize Smart Lists in reporting applications

> **Note:**
>
> For a Smart List to be viewable in forms, the dimension to which the Smart List type member belongs must be first in the evaluation order. See Managing Dimensions.

To create or work with Smart Lists:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Smart Lists**.
2. Perform one action:
   - To create a Smart List, click , and then set Smart List properties. See Setting Smart List Properties.
   - To edit a Smart List, select it, click , and then set Smart List properties. See Setting Smart List Properties.
   - To delete Smart Lists, select them, click , and then click **OK**. Deleting Smart Lists also deletes any associated mappings with dimension members and reporting applications.

     Data cells can display only one Smart List. If multiple Smart Lists intersect at cells, set which one takes precedence.
   - **Optional**: Click to synchronize Smart Lists in a reporting application. See Synchronizing Smart Lists in Reporting Applications.

> **Note:**
>
> When a member in a Smart List is deleted and then added back in again, the new member assumes a new unique ID. Any data that was associated with the deleted member will display as the deleted member's numeric ID.

Synchronizing Smart Lists in Reporting Applications

Synchronizing Smart Lists in reporting applications identifies dimensions in reporting applications to which Smart Lists are mapped, and adds level 0 members of the selected dimensions as new Smart List entries to the selected Smart Lists. To define data maps, see Defining Data Maps in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

To synchronize Smart Lists in reporting applications:

1. View the **Smart Lists** listing page.

   See Working with Smart Lists.
2. Click the Smart List you want to synchronize, and then select

   .

   During synchronization, values from reporting applications in all existing mappings are appended after the last Smart List item in the appropriate Smart list. If a Smart List is mapped to two dimensions, all members from the first mapping are inserted first, and then members from the second mapping are inserted. If a member already exists in a Smart List, it's not added again. Smart List members in the FreeForm Smart Lists are not deleted, even if the corresponding dimension members on the reporting application are deleted.

> **Note:**
>
> If Account is mapped as Smart List to Dimension, all of the level 0 members in the Account dimension are brought in as Smart List entries when the Smart List is synchronized. For example, Smart Lists may include entries such as HSP_Average and HSP_Ending. If this occurs, delete the extra entries from the Smart List.

3. Click **OK** in the dialog box that says "Smart List Synchronized Successfully."

> **Note:**
>
> Smart List names can't have spaces in them. If you're synchronizing Smart Lists in a reporting application, ensure that any new members don't have spaces in the name.

Setting Smart List Properties

Define Smart List properties on the Properties tab.

> **Note:**
>
> For a Smart List to be viewable in forms, the dimension to which the Smart List type member belongs must be first in the evaluation order. See Managing Dimensions.

Table 19-26 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. Note the following: • It displays as the first selection in the Smart List drop-down, allowing #MISSING as a selection in the form. • When the cell isn't in focus, this label displays only if **Drop-Down Setting** is selected in the next option. Otherwise, #MISSING or a blank cell is displayed, depending on the **Display Missing Values As Blank** selection for the form. • #MISSING labels determine only the display of cells with #MISSING data; #MISSING remains the stored value.

Table 19-26 (Cont.) Smart List Properties

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's 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 don't select this option, you can customize Smart List ID values.
Create from Members	Create a Smart List based on dimension hierarchies. Smart List values are dynamically updated when members are updated. Note that user security for the dimension is honored for Smart Lists created from the dimension's hierarchy.
Member Selection	Manually select the members to use in Smart Lists.

Use the **Entries** tab to define selections on Smart Lists. See Defining Smart List Entries.

Defining Smart List Entries

To define Smart List entries:

1. On the **Entries** tab, define drop-down list items:
   - **For first items only**: enter information into the first row.
   - To add an item, click  and enter the information.
   - To delete an item, select it and click  .
   - To edit an entry, change the information in its row:

Table 19-27 Smart List Entries

Entry Property	Description
ID	Unique number that sets the order for the displayed entry. The ID is customizable only if **Automatically generate ID** isn't selected on the **Properties** tab.
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).

> **Note:**
>
> Items highlighted in red are duplicates.

2. Click **Save**.

   Use the **Preview** tab to preview the defined Smart List. See Previewing Smart Lists.

Previewing Smart Lists

Preview the defined Smart List on the **Preview** tab. The tab shows the Smart List as displayed in a drop-down list or a table.

To preview a Smart List:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Smart Lists**.
2. Select a Smart List and click .
3. Click **Save**.

Displaying #MISSING with Smart Lists

Service Administrators set values displayed in Smart Lists and data cells, including the display when no data is in the cell. Cells can display no value, #MISSING, or (for cells associated with Smart Lists) a specified value.

Use these options to control the display of #MISSING when cells are not in focus:

Table 19-28 Display Options for #MISSING

Option	Guideline
Blank	When designing forms, select **Display Missing Values as Blank**. When setting Smart List properties, select **Form Setting**.
#MISSING	When designing forms, don't 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**.

Administering Task Lists

Guide users through the planning process with Task Lists.

Task Lists guide users through the planning process by listing tasks, instructions, and end dates. Service Administrators and Power Users create and manage tasks and task lists.

Related Topics

- Working with Task Lists
- Adding Instructions to Task Lists
- Adding Tasks to Task Lists
- Editing Task Lists
- Assigning Access to Task Lists

Working with Task Lists

Task lists organize groups of tasks for users. You must create task lists before creating tasks.

To create and rename task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. Perform a task:
   - To create a task list, click .
   - To rename a task list, select the task list, then click .
   - To move a task list, select the task list, then click .
   - To remove a task list, select the task list, then click .
3. Click **OK**.

To define the task list, see:

- Adding Instructions to Task Lists
- Adding Tasks to Task Lists

Adding Instructions to Task Lists

To add instructions to task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. For **Task List**, select the task list to modify, and then click .
3. Click the **Instructions** tab.
4. Enter instructions for the task list.
5. Click **Save**, and then click **Close**

Adding Tasks to Task Lists

You can set completion dates and alerts for tasks that users perform at runtime. You can also set up email messages, for example, to alert users that a task was not completed by its end date. Alert messages are sent after an "alert date" that you set, and are repeated until the end date is reached for a task.

To add tasks to task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. Perform a task:
   - Define a task list by clicking , and then entering a name.
   - Modify a task list to add a new task by selecting it, and then clicking .
3. Click .

4. For **Task**, enter a task name.
5. From **Type**, select the kind of task you're creating and what it should enable users to do. For example, if you're creating a task that enables Service Administrators to modify all dimensions at runtime, select **Dimension Editor**.

   Specify the information required for the task type using this table:

Table 19-29 Task Information

Task Type	Action
URL	Opens a specified URL Enter a fully qualified URL to associate with this task, such as `http://www.company_name.com`, and then go to step 6.
Form	Opens a form Select the form for users to complete, and then go to step 6. **Optional**: Select **Set Page Member Defaults** to select the member from each dimension to display as the default when the task is first opened. After you select this option, you can select the members for the page dimensions. The page member defaults apply until a user updates the form and returns to the task in another session. Where page member defaults are set, they override the most recently used settings in each session.
Dashboard	Opens a dashboard Select the dashboard for users to work with, and then go to step 6.
Business Rule	Launches a business rule Perform these tasks: • From **Cube**, select the cube associated with the business rule to execute. • From **Business Rules**, select the business rule to execute. • Go to step 6.
Descriptive	Enables Service Administrators to add a task with no task properties.
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 Perform these tasks: • In **Job Type** select the kind of job to be displayed in the console. • In **Status**, select the status of the task to be displayed in the console. • Go to step 6.
Dimension Editor	Enables Service Administrators to view and modify a dimension at runtime Perform these tasks, and then go to step 6: • 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.
Refresh Application	Enables Service Administrators to refresh the application at runtime to capture recent changes. After enabling, go to step 6.
Import and Export	Enables Service Administrators to import and export data and metadata using flat files. 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 6.

> **Note:**
>
> The Security Management, Process Management, and Form Management task types cannot be added to a task list or edited in the Simplified Interface. These task types are accessible only in desktop mode of the application.
>
> The Dashboard task type cannot be added to a task list or edited while in desktop mode. This task type is accessible only in the Simplified Interface.

6. For **Duration**, enter how long you expect it will take to complete the task. Enter a number in the first field and the unit of time in the second field; for example, if you think the task will take two weeks, type 2 in the first field and then in the second field, type a `w` or `W` to select **Week(s)**. Time unit options for the second field are as follows:
   - **Hour(s)**: Type `h` or `H` to select this option.
   - **Day(s)**: Type `d` or `D` to select this option.
   - **Week(s)**: Type `w` or `W` to select this option.
   - **Month(s)**: Type `m` or `M` to select this option.
   - **Year(s)**: Type `y` or `Y` to select this option.

> **Note:**
>
> Unit of time options are not case-sensitive. If the language you are using is not English, these entries are localized. Type the first letter of the localized word and then select the whole word that is displayed.

7. To ensure that the task is started on a certain date and then completed by a certain date, perform these tasks:
   a. For **Start Date**, click  , and then use the calendar to specify the year, month, day, and time by which the task must begin.
   b. For **End Date**, click  , and then use the calendar to specify the year, month, day, and time by which the task should be completed.

      To send email messages that prompt users to finish uncompleted tasks, select **Repeat Every**, and then specify how often to send these emails, such as once a day.
8. To send email messages that prompt users that the date by which they must complete the task is approaching, select **Alert**, and then:
   a. Click  to use the calendar to specify the date and time on which to start sending alert reminders.
   b. Click **Repeat Every** and specify the frequency for email reminders.

   For example, if a task's end date is June 9 2014 at 5 PM, and you want to remind users once a day starting the week before the task is due, select June 2 2014 at 5 PM, enter 1, and then select Weeks.
9. To make task completion dependent on completing another primary (or parent) task, click  next to **Dependency**, select the primary task, and then click **OK**.

> **Note:**
>
> You cannot make task completion dependent on completing a child task of another primary task. Dependencies across task list hierarchies are not supported when a task has children. For example, consider a case where Task 3 is dependent on Task 2 and Task 4 is, in turn, dependent on Task 3. When Task 3 is incomplete, Task 4 becomes incomplete automatically. However, when Task 2.1 is incomplete, its parent task (Task 2) changes to incomplete, but this is not recursive (Task 3 and Task 4 are not affected).
>
> - Task1
> - Task2
>   - Task2.1
>   - Task2.2
> - Task3
> - Task4

10. In the **Instructions** pane to the right, enter information that explains how to complete the task. Use the formatting buttons to customize the information. For example, you can modify the font, font size, and layout (as bullet points or numbered steps, for example).
11. Click **Save** and then click **OK**.

Editing Task Lists

Related Topics

- Editing Tasks
- Copying and Moving Tasks
- Moving and Reordering Task Lists
- Clearing Task Lists
- Deleting Tasks and Task Lists

Editing Tasks

Use the Edit Task dialog box to modify the type of task, its instructions, the end date and alert date, and email reminder messages.

To edit tasks:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Task Lists**.
2. Select a task list, and then select .
3. Select a task, and then select .
4. In **Edit Task**:
   - Modify the task name.
   - Select another task type.

- **Optional**: Modify the values for **Duration**, **Start Date**, **End Date**, Alert, and Dependency options.

For descriptions of the various options, see Adding Tasks to Task Lists.

5. For **Task - Instructions**, modify instructions for completing the task.
6. Click **Save**.

Copying and Moving Tasks

To copy or move tasks:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. Perform an action:

   To copy tasks:

   a. Select a task list, and then select .
   b. Select the task to copy, and then click **Save As**.
   c. Enter the name for the new task list, and then click **OK**.

   To move tasks:

   a. Select a task list, then select .
   b. Select a task, and then click .
   c. Select the task that will be a child of another task, and then select .
   d. Select the task that will be the parent task, and then select .
   e. Click **Save**.

   To cut and paste tasks:

   a. Select a task list, and then click .
   b. Select a task, and then click .
   c. To move the task to a new position, select the task to appear above it.
   d. Click .
   e. Click **OK**.

Moving and Reordering Task Lists

To move or reorder task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. Perform a task:

   To move a task list:

   a. Select a task list, and then click .
   b. Select the destination folder.

c. 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 doesn't delete the dates.

To clear task lists:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Task Lists**.
2. Select the task list you want to clear, and then click .
3. In the **Clear** drop-down list, select an option:
   - **Completion Status**: Clears completion status only
   - **Due Dates and Alerts**: Clears any alerts that are enabled, based on the end date set for the task
   - **Both**: Clears completion status, end dates, and alerts
4. Click .
5. Click **OK**.

Deleting Tasks and Task Lists

To delete tasks and task lists:

1. Click the **Navigator** icon , and then under **Create and Manage**, click **Task Lists**.
2. Perform a task:
   - To delete a task, select the task list with the task to delete, and then click . Select the tasks to delete, and then click .
   - To delete a task list, select the task list to delete, and then select .
3. Click **OK**.

Assigning Access to Task Lists

You can determine who can view and modify task lists. By default, Service Administrators can manage and assign access permissions for task lists.

> **Note:**
>
> Being assigned to a task list means being able to access and complete tasks in the task list. It doesn't mean being able to assign tasks to someone else.

- Adding Access to Task Lists
- Changing and Removing Access to Task Lists

Adding Access to Task Lists

To assign access to task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. Select the task list to modify.
3. Click .
4. Click .
5. Select the users or groups to access the task list.
   - Click **Users** to display all user names; click **Groups** to display all groups.
   - If there are multiple pages of users and groups, a pagination bar displays at the bottom of the page to help you navigate between the pages and to jump to a specific page.
6. For **Type of Access**, select how users or groups can use the task list:
   - **Assigned**: View and use
   - **Manage**: Modify
   - **Manage and Assign**: View, use, and modify
   - **None**: No access
7. Click **Add**.
8. Click **Close**.

Changing and Removing Access to Task Lists

To change or remove access to task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.
2. Select the task list to modify, then click .
3. Perform a task:

   To change access to task lists:

   a. Select users or groups, and click .
   b. For **Type of Access**, select:
      - **Assigned**: View and use
      - **Manage**: Modify

- **Manage and Assign**: View, use, and modify
- **None**: No access

c. Click **Set**.

To remove access to task lists:

a. Select the user or group, and click  .

b. Click **OK**.

4. Click **Close**.

Clearing Cell Details

You can clear these cell details for a cube: account annotations, supporting detail, cell text, and cell-level documents. For information on creating and viewing account annotations, supporting detail, cell text, and cell-level documents, see *Working with FreeForm*.

> **Note:**
>
> Because this function is for Service Administrators and Power Users, the application assumes that you have full access to details you delete.

To clear cell details:

1. Click the **Navigator** icon  , and then under **Actions**, click **Clear Cell Details**.
2. For **Cube**, select a cube.
3. To select all dimensions in the cube, click **Add All Dimensions**.

   Select members for the displayed dimensions.
4. Specify the type of information to delete by selecting at least one option from **Clear Options**.
5. Select members for the data intersections:

   a. In **Dimension**, select at least one dimension with details to delete.

   b. For displayed dimensions, click  . On the **Member Selection** page, make a selection that includes the details to delete.

> **Note:**
>
> For every dimension selected, you must select at least one member. If a dimension isn't selected, the application includes all its members when clearing cell details.

> **Caution:**
>
> To clear exactly the data you want, select at least one Account member and members from all other dimensions. If you don't select at least one Account member, no data is cleared, even if you select the **Yes, set value to #Missing** option. If you don't select members from the other dimensions, all data for the selected Account member is cleared across all other dimensions.

6. **Optional**: To select another dimension so you can select its members, click  .
7. Click **Clear**.

   A Confirmation page displays your selections.
8. Click **Finish** to proceed, or **Back** to change your selections.

   If Clear Cell Details is successful, data is deleted from the cube. If data isn't deleted successfully, a message is displayed. You can also check the log file.
9. **Optional**: To view the execution status of Clear Cell Details and review the information that was deleted, view the Job . See Managing Jobs.
10. To clear cell details for another cube, select another cube in step 2, and repeat the procedure.

Copying Data

You can copy plans from one dimensional intersection to another, including relational data and supporting detail. For example, you can copy *Budget, FY10, Final* to *Forecast, FY11, First Draft*.

> **Note:**
>
> - Selected Copy Data settings are preserved for the current session only.
> - Copied dimension members must be present in the selected cubes.
> - Data must be copied into cells that can accept data. For example, you can't copy data into read-only or dynamic cells.
> - You can copy account annotations, supporting detail, and cell text.
> - You can't use this feature with attributes, so don't select attributes to be copied.
> - Because this is an administrative function, the application assumes you have full access to data you copy.
> - This feature doesn't calculate data. To perform calculations, such as increasing the forecast by 5%, apply the business rule after copying data.
> - For Copy Data to be successful, you must select at least one member for Scenario, Account, Entity, Period, and Version dimensions.
> - Copy Data is enabled for block storage and aggregate storage cubes.

To copy data:

1. Click the **Navigator** icon  , and then under **Actions**, click **Copy Data**.
2. In **Cube**, select a cube.

   Dimensions are displayed for this cube. You can only copy from one cube at a time.
3. In **Copy Data Options**, select the type of information to copy.
4. Under **Static Dimensions**, select the members for the data intersections:
   a. For **Dimension**, select a dimension from which to copy.
   b. For **Members**, click  to make a selection from which to copy. You can select multiple members. You must select at least one member for Scenario, Account, Entity, Period, and Version dimensions.
5. **Optional**: To add another static dimension to the list, click  and select dimension members.

   > **Note:**
   >
   > To remove a dimension, select **Select Dimension**. The dimension moves to the **Dimensions with Source and Destination** area.

6. In **Dimensions with Source and Destination**, enter dimensions into which to copy data.

   In **Source** and **Destination** click  .
7. Click **Copy Data**.

   Data is copied from one intersection to the other. If data isn't copied successfully, a message is displayed. You can also check the log file.
8. To copy data for another cube, select another cube in step 2, and repeat the procedure.

Administering Application Diagnostics

Related Topics

- About Application Diagnostics
- Assumptions
- How Application Diagnostics Works
- Using Application Diagnostics Graphs
- Launching Application Diagnostics
- Modifying Artifacts for Optimal Performance

About Application Diagnostics

Application diagnostics enables Service 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. Service Administrators can use application diagnostics to evaluate the following:

- An entire application

- Types of artifacts such as forms
- Individual "cherry picked" artifacts

Application diagnostics identifies changes that Service 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. Application diagnostics evaluates your application and artifacts based on the user a Service Administrator selects. This enables you to determine the potential problems that may occur at runtime given the individual user's variables and security access as part of the analysis.

Assumptions

Oracle can only provide general guidelines about how to design your application and artifacts to ensure reasonable performance. Oracle can't specify precise value ranges for artifacts and operations.

How Application Diagnostics Works

Application diagnostics 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. Application diagnostics 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 application diagnostics to analyze the following:

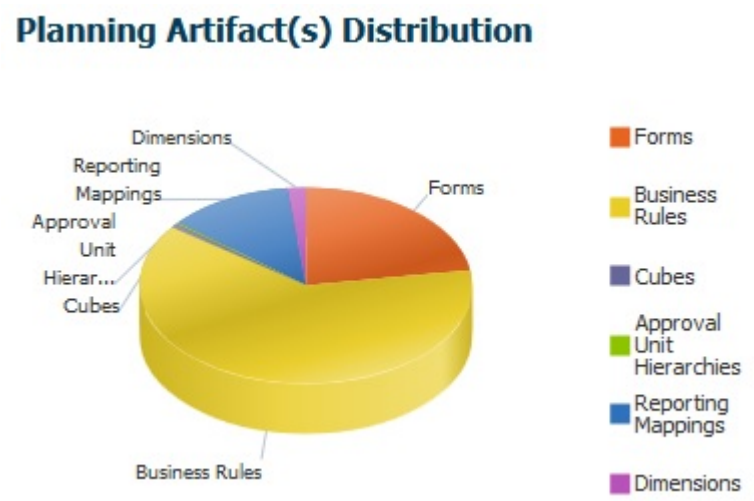
- Block storage cubes
- Dimensions
- Forms
- Standalone business rules
- Reporting mappings
- Export data functionality

Using Application Diagnostics 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:

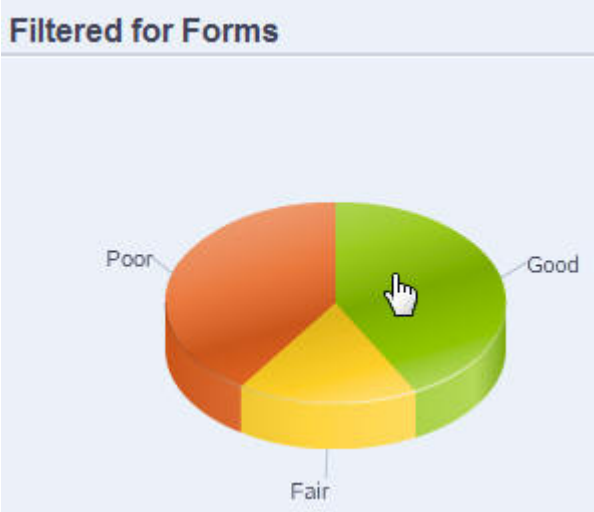
- **<Business Process> Artifacts Distribution** : Depicts how much of your application is comprised of various artifact types. Place your mouse over portions of the graph to view how many artifacts of each type were analyzed by application diagnostics. For example, if you mouse over forms and the pop-up 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.

  Example Artifact Distribution Graph:



- **Filtered For <Artifact>**: Click the graph portions that represent the artifact status, and to view and drill down on artifacts with a particular status.

  Filtered Graph:



Launching Application Diagnostics

To launch application diagnostics:

1. Click the **Navigator** icon, and then under **Monitor and Explore**, click **Application Diagnostics**.
2. In **Select User**, click to select the user whose settings you want to use to perform the analysis, and then click .
3. Perform a task:
   - A pie graph depicts the distribution of artifacts in the application. The lower grid represents the integrity of all application artifacts. A status of fair and poor indicate artifacts that you should or must modify for optimal performance.

- To analyze specific types of artifacts, such as forms, select the slice for the type within the pie graph. For example, to ensure that business rules are designed well and will not cause performance issues, click the Business Rules slice of the pie graph. If a large portion of the graph is fair or poor, 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, expand **Search** to find the artifact by specifying its name, start date, end date, and type, 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 Application Diagnostics Graphs.

4. Click  or select **Actions**, and then **Run Application Monitor**.
5. If the **View Details** column doesn't display in the grid, select **View**, then **Columns**, and then **View Details**. This also enables you to display or conceal different types of artifact data.
6. To display information about how to fix artifacts, click  in **View Details**. Then click artifacts to launch editors that enable you to modify them.

Modifying Artifacts for Optimal Performance

After reviewing the details provided by the application diagnostics that tell you how to modify your artifacts, edit them as follows:

- Forms: Clicking on the form artifact name launches the **Form Designer** in a new tab. Edit forms as described in Administering Forms.
- Business Rules: Clicking a business rule name launches the **System View** which displays all application artifacts that you can access. You can perform tasks such as:
  - Use the upper **Rule Designer** to view the individual components of the rule, such as conditions, commands, and scripts. Click each component to modify them in the lower portion of the page. You can also insert and remove components like formulas, system templates, and scripts by dragging and dropping them from the **Rule Pallet** and **Existing Object** panes to the left.
  - From **Designer**, select **Edit Script** to modify and format the script by performing tasks such as inserting functions, editing templates, inserting variables, inserting member ranges, use comments, and validate syntax.
  - Create, open, delete, and refresh artifacts, pages, and other documents.
  - Import and export business rules and other objects.
  - Determine how and where artifacts are used.

    See *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud* and Understanding Runtime Prompts.
- Cubes: Clicking the cube artifact name launches the **Performance Settings** tab of the dimension editor in a new tab. Edit the dimensional outline or dimensional sparsity as described in Editing Dimensions in the Simplified Dimension Editor.

- Reporting Mappings: Clicking the reporting mapping artifact name launches the **Report Mapping Designer** in a new tab. Edit reporting mappings as described in Defining Data Maps in *Administering Data Integration for Oracle Enterprise Performance Management Cloud*.

A

Naming Restrictions

Understand important restrictions for names, including names for applications, databases, dimensions, members, aliases, users, and groups.

Related Topics

- Naming Restrictions for Applications and Databases
- Naming Restrictions for Dimensions, Members, and Aliases
- Dimension and Member Names in Calculation Scripts, Report Scripts, Formulas, Filters, and Substitution Variables
- Restrictions for User and Group Names

Naming Restrictions for Applications and Databases

When naming an application, follow these rules:

- Use no more than eight characters for an application's name.
- Don't use spaces or tabs.
- Don't use an underscore (_) as either the first or last character.
- Don't use these special characters:

Table A-1 Restricted Characters in Application Names

Character	Meaning
*	asterisk
\	backslash
[ ]	brackets
:	colon
,	comma
"	double quotation mark
=	equal sign
>	greater than sign
<	less than sign
.	period
+	plus sign
?	question mark
;	semicolon
'	single quotation mark
/	slash
|	vertical bar

- For application names in relational database environments, don't use extended characters (except for underscores).
- For aggregate storage databases, don't use DEFAULT, LOG, METADATA, or TEMP as application names.

Enter names in the preferred case. Application names are created exactly as entered.

Naming Restrictions for Dimensions, Members, and Aliases

When naming dimensions, members, and aliases, follow these rules:

- Use no more than 80 characters for dimension, member, or alias names.
- Names are not case-sensitive for dimensions, non-shared members, and aliases. Do not use matching names with only case differences; for example, do not name two members "zzz" and "ZZZ," or a base member "zzz" and a shared member "ZZZ."

  In other words, shared member names must use the same name and case of the base member. When you rename a base member (even if all you do is change the case), the shared member names will be updated to match the base member names.
- Do not use dimension names that are the same name as a member property or a column header in the metadata load file; for example, Data Type.
- HTML tags cannot be used in member names, dimension names, aliases, and descriptions.
- The TAB key is restricted and cannot be used in dimension, member, and alias names.
- The following is a list of characters that are restricted and cannot be used in dimension, member, and alias names:

Table A-2 Restricted Characters in Dimension, Member, and Alias Names

Character	Meaning
"	quotation mark
[ ]	bracket
\	backslash
/	slash

> **Note:**
>
> – Brackets [ ] are permitted but not recommended in block storage outlines. They cause errors when converting to aggregate storage outlines.
>
> – Do not use the ASCII substitute character (zero-width space character) `0x1A` in member names and aliases.

- The following is a list of characters that cannot be used at the beginning of dimension, member, and alias names:

Table A-3 Restricted Characters at the Beginning of Dimension, Member, and Alias Names

Character	Meaning
@	at sign
&	ampersand
\	backslash
[ ]	brackets
,	comma
-	dash, hyphen, or minus sign
=	equal sign
<	less than sign
( )	parentheses
.	period
+	plus sign
"	quotation mark
/	slash
_	underscore
|	vertical bar

> **Note:**
>
> Member names with the preceding characters are not allowed in business process data export jobs.

- Spaces cannot be placed at the beginning or end of names.
- For time periods in custom calendars, don't use spaces in prefixes.
- The following cannot be used as dimension, member, or alias names:
  - Calculation script commands, operators, and keywords
  - Function names and function arguments
  - Names of other dimensions and members (unless the member is shared)
  - If Dynamic Time Series is enabled, don't use History, Year, Season, Period, Quarter, Month, Week, or Day.
  - In unique member outlines that contain an attribute dimension, you cannot use Sum, Count, Min, Max, and Avg as member or dimension names. Using these names causes duplicate name conflicts.

> **Note:**
>
> The members that are created in the attribute calculations dimension (Sum, Count, Min, Max, and Avg) are not considered reserved words because you can change these names in the attribute calculations dimension and then use the standard name in an attribute or standard dimension.
>
> If the outline is tagged as a duplicate member outline, and an attribute dimension (and, therefore, the attribute calculations dimension) exists before you use Sum, Count, Min, Max, and Avg as a base member, the duplicate name is allowed. However, if, you use Sum, Count, Min, Max, and Avg as a base member before creating an attribute dimension, the duplicate name isn't allowed.

- The following is a list of words that are restricted and cannot be used in dimension, member, and alias names:

Table A-4 Restricted Words in Dimension, Member, and Alias Names

Restricted Word	Restricted Word	Restricted Word
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	REQUEST	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 (&)
  - asterisk (*)
  - at sign (@)
  - backslash (\ )
  - braces ({ })
  - colon (:)
  - comma (,)
  - dash, hyphen, or minus sign (-)
  - exclamation point (!)
  - equal sign (=)
  - greater than sign (>)
  - less than sign (<)
  - parentheses ( )
  - percent sign (%)
  - period (.)
  - plus sign (+)
  - semicolon (;)
  - slash (/)
  - tilde (~)

In calculation scripts and formulas, enclose member names that are also 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 and Group Names

User and group names can have up to 80 characters.

B

Form Formula Functions

Learn how to use formulas and functions in forms.

Related Topics

- About Form Formula Functions
- Working with Formulas
- Formula Functions

About Form Formula Functions

After you create a formula row or column, define the formula using the **Segment Properties** pane. Formulas include grid references, mathematical operators, and, optionally, mathematical functions. For a complete list, see Formula Functions.

> **Note:**
>
> The application requires level 0 members that are Dynamic Calc to have a member formula. For Dynamic Calc members that don't have a formula, the application inserts a semicolon (;) when refreshed.

Working with Formulas

Related Topics

- Creating Formulas
- Editing Formulas
- Deleting Formulas

Creating Formulas

To create formulas:

1. Select the row or column for which to associate the formula.
2. If it doesn't automatically expand, select **Segment Properties**.
3. In Formula, enter the name for the formula and then click  .
4. In the **Formula** box that is displayed, select the operation or function that the formula will perform, such as COUNT(), MAX(), and IfThen(). See Formula Functions.
5. Click **Validate** to ensure that the formula doesn't contain any errors.

Editing Formulas

To edit formulas:

1. Open a form with a formula row or column.
2. In **Layout**, select a numbered formula row or column.
3. Use **Segment Properties** to modify the formula properties.
4. Click  to change the operation or function performed.
5. Save the form.

Deleting Formulas

To delete a formula:

1. Select the formula row or column.
2. Click **Delete**.
3. To verify formula deletion:
   a. Click the checkmark in the formula bar.
   b. Click another cell within the grid to reset the formula bar.
   c. Click the cell from which you deleted the formula to verify deletion.

Formula Functions

This section defines the mathematical functions available for creating formulas for form formula rows and columns. To insert formula rows and columns in forms, see Adding Formula Rows and Columns.

The syntax for mathematical functions:

```
FunctionName(arguments)
```

For more information about arguments, see Arguments.

Table B-1 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

Mathematical Functions

Table B-2 Mathematical Functions

Function	Description
Abs	Returns the absolute value of numeric values or references. See Abs.

Table B-2 (Cont.) Mathematical Functions

Function	Description
Average	Returns the average of a group of numeric values or references. See Average.
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. See AverageA.
Count	Returns the number of values in a group of numeric values or references. See Count.
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. See CountA.
Difference	Returns the absolute value of a numeric value or reference subtracted from another numeric value or reference. See Difference.
Eval	Evaluates an expression. Eval is useful for embedding expressions as function arguments. See Eval.
IfThen, If	Returns one value if a condition equals true, and another value if a specified condition equals false. See IfThen, If.
Max	Returns the maximum value from a group of numeric values or references. See Max.
Min	Returns the minimum value from a group of numeric values or references. See Min.
Mod	Returns the remainder, modulus, from a division formula. See Mod.
PercentOfTotal	Returns the result of a numeric value or reference divided by another numeric value or reference, multiplied by 100. See PercentOfTotal.
Pi	Returns the number 3.14159265358979, to 15 digits. See Pi.
Product	Multiplies all numbers or references and returns the product. See Product.
Random	Returns a random number between 0.0 and 1.0. See Random.
Round	Rounds a number up or down by specified digits. See Round.
Sqrt	Returns the square root of a numeric value, row, column, or cell. See Sqrt.
Sum	Returns the sum of a group of numeric values or references. See Sum.
Truncate / Trunc	Removes the specified number of digits from numeric values. See Truncate/Trunc.
Variance / Var	Evaluates the difference between the specified values based on the account type for the current account. See Variance/Var.
VariancePercent / VarPer	Evaluates the percentage difference between the specified values based on account type for the current account. See VariancePercent/VarPer.

Arguments

Mathematical functions accept numeric values, row, column, or cell references, or embedded functions as arguments. There are four argument types:

- Numeric

  See Numeric Arguments.

- Property

  See Property Arguments.
- Row, column, or cell reference

  See Row, Column, or Cell Reference Arguments.
- Embedded functions

  See Embedded Functions as Arguments.

Numeric Arguments

The syntax for a numeric argument is

```
(numeral1, numeral2,...numeraln)
```

where numerals *1* through *n* are any numbers including decimals and negative values. For example, the expression Average(10,20,30) returns the value 20.

Row, Column, or Cell Reference Arguments

The row, column, or cell argument identifies a row, column, or cell in a grid. The syntax:

```
FunctionName(GridName.GridElement[segment(range)].Property)
```

Table B-3 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 isn't 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 isn't specified, letters represent columns and numbers represent rows; for example: Max ([1, A], [2, A], [3, A]) refers to rows 1, 2 and 3 of column A.
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.

Table B-3 (Cont.) Argument Components

Argument	Description
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* isn't provided, all expanded cells are used. **Note:** If a segment expands to only one row or column, don't 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

  See Aggregate Property Argument.
- Reference

  See Reference Property Argument.

Aggregate Property Argument

An aggregate row, column, or cell contains multiple rows, columns, or cells, respectively.

The aggregate property argument is the last argument in the following mathematical function syntax:

```
FunctionName(GridName.Gridelement[segment(range)].property)
```

You apply the following aggregate properties to a row, column, or cell reference.

Table B-4 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
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
```

Table B-5 Reference Property Arguments

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

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 doesn't change. The function syntax:

```
Abs (argument)
```

where *argument* is one of the following:

Table B-6 Arguments for the Abs Function

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 doesn't include missing values regardless of whether they are suppressed or not.

The function syntax:

```
Average(arguments) or Avg(arguments)
```

where *arguments* is one or more of the following:

Table B-7 Arguments for the Average Function

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:

Table B-8 Arguments for the AverageA Function

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:

Table B-9 Arguments for the Count Function

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:

Table B-10 Arguments for the CountA Function

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

Examples:

- In the following example, if a grid has 4 rows with the values 10, 20, 30, and, #ERROR. The following formula in the fifth row returns the count of four rows:

```
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:

Table B-11 Arguments for the Difference Function

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:

Table B-12 Arguments for the Eval Function

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's supported by a conditional operator.

Table B-13 Conditional Operators

Conditional Operator	Syntax	Logic
Equal To	*expression* = *expression*	Tests whether the left expression is equal to the right expression. The routine that evaluates the condition doesn't consider any rounding. If rounding is required, use the Round function. Example: `1= 4` Returns false
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. The correct syntax is ">=". The syntax "=>" isn't 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
Less Than or Equal To	*expression* <= *expression*	Tests whether the left expression is less than or equal to the right expression. The correct syntax is "<=". The syntax "=<" isn't supported. Example: `1 <= 4` Returns true

Not Equal To	*expression* <> *expression*
	expression != *expression*
		Tests whether the left expression isn't equal to the right expression.
		The routine that evaluates the condition doesn't consider any rounding. If rounding is required, use the Round function.
		Examples:
		`1 <> 4`
		Returns true
		`1 != 4`
		Returns true

IsMissing	`IsMissing` (*reference*) `IsMiss` (*reference*)	Tests whether the reference contains a #MISSING result. 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.

Table B-13 (Cont.) Conditional Operators

Conditional Operator	Syntax	Logic
IsError	`IsError (`*reference*`)` `IsErr (`*reference*`)`	Tests whether the reference contains an #ERROR result. 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.
		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 B-14 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 doesn't apply when you use the IsMissing, IsError, or IsNonNumeric conditions.

Complex Conditions

Complex conditions And, Or, and Not are fully supported. However, they must be surrounded by parentheses.

Valid example:

```
If ( ([A] > [B] and [A] > 1000), [A], [B])
```

Invalid example:

```
If ( [A] > [B] and [A] > 1000, [A], [B])
```

Max

Max is a mathematical function that returns the maximum value in a group of numeric values, rows, columns, or cells. The function syntax:

```
Max (arguments)
```

where *arguments* is one or more of these arguments:

Table B-15 Arguments for the Max Function

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:

Table B-16 Arguments for the Min Function

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:

Table B-17 Arguments for the Mod Function

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`

Table B-17 (Cont.) Arguments for the Mod Function

Argument	Description
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:

Table B-18 Arguments for the PercentOfTotal Function

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

> **Note:**
>
> This function requires two arguments.

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]),
```

Results for the PercentOfTotal Function

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:

Table B-19 Arguments for the Product Function

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:

Table B-20 Arguments for the Round Function

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:

Table B-21 Arguments for the Sqrt Function

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:

Table B-22 Arguments for the Sum Function

Argument	Description
numeric	A numeric value. For example, Sum(10, 20, 30) returns the value 60. 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 30:

  ```
  sum(10,20)
  ```

- The following example returns the sum of three rows:

  ```
  sum(row[1],row[6],row[8])
  ```

- The following example calculates the sum of three aggregate columns:

  ```
  sum(column[E], column[G], column[I])
  ```

When a formula row includes `IDESC`, the sum includes all of the parents and their descendants.

For example, a form is created with the following rows (and each member has children):

```
IDESC("Mem1"), IDESC("Mem2"), IDESC("Mem3"), IDESC("Mem4")
```

If a formula row is added with the following formula:

```
SUM(ROW[1],ROW[2],ROW[3],ROW[4])
```

When the form is opened for data entry, the formula row will return the sum of all the parents and their children.

Truncate/Trunc

Truncate is a mathematical function that removes the specified number of digits from numeric values.

Syntax:

```
Trunc (arg1, integer)
```

- where *arg1* is one of these arguments:

Table B-23 Arguments for the Truncate/Trunc Function

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 B-24 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
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:

  Table B-25 Results for the Variance Function When Comparing Two Different Account Types

Sales	Expense	Result
-400	100	-500

- When the Variance function is applied to a dimension that isn't tagged as an Accounts type, an #ERROR results at runtime.
- #MISSING is treated as zero (0), unless specified differently using the ifnonnumber property.

Examples

The Variance function accepts cell, column, or row references only.

Table B-26 Syntax Examples for the Variance Function

Syntax	Example
Sample syntax referencing a column:	`Var ([A], [B])`

Table B-26 (Cont.) Syntax Examples for the Variance Function

Syntax	Example
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 B-27 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:

  Table B-28 Expected Results of the VariancePercent Function when Two Account Types are Compared

Sales	Expense	Result
-400	100	-5.

- When the VariancePercent function is applied to a dimension that isn't of type Accounts, an #ERROR results at runtime.
- #MISSING is treated as zero (0), unless specified differently using ifnonnumber property.

Examples

The VariancePercent function accepts, cell, column, or row references only.

Table B-29 Syntax Examples of the VariancePercent Function

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])
```

Example Result of the VariancePercent Function when Comparing Columns

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 Manage Applications

Understand how to manage applications in Smart View, including installation, controlling the display, downloading templates, and creating, updating, and deleting applications. Also learn about the Planning Admin Extension and Office AutoCorrect.

Related Topics

- About Managing Applications in Smart View
- Installing Smart View and the Admin Extension for Application Management
- Controlling the Display of Application Management Options in Smart View
- Downloading the Application Templates
- Creating an Application
- Working with Artifacts in the Application Template
- Updating an Application in Smart View
- Deleting an Application
- Planning Admin Extension and Office AutoCorrect

About Managing Applications in Smart View

Applies to: FreeForm, Planning

Service Administrators can use Excel to create and manage Planning, Planning Modules, and FreeForm applications in Oracle Smart View for Office and Oracle Smart View for Office (Mac and Browser).

> **Note:**
>
> You must have Service Administrator privileges to create and manage applications in Smart View and Smart View (Mac and Browser).

Using a downloadable Excel template workbook, you can quickly create, model, and manage applications in Smart View. By modifying the worksheets within the application template workbook, you can define application properties, dimension and member properties, security (access permissions), and substitution variables to fit your needs. You can also load data from the template workbook.

Once the application is created, it's available to users from Oracle Smart View for Office, Oracle Smart View for Office (Mac and Browser), and the web application. Service Administrators can further update and refresh the application in Smart View as needed. You can also delete the application from Smart View.

Application management in Smart View is a quick and easy way to get an application up and running in your business process environment.

To begin, see the topics in Installing Smart View and the Admin Extension for Application Management.

After installation, get started by downloading and then editing the application template workbook available in Smart View. Or, download the application template zip file from **Downloads** menu in your business process. The template zip file contains starter workbooks for Planning, Planning Modules, and FreeForm applications, and a sample Planning application workbook. See Downloading the Application Templates.

> **Note:**
>
> Starting in 22.03, a single Excel template workbook, `PlanningApplicationTemplate_FreeForm.xlsx`, allows you to create a multi-cube application (any combination of up to 12 ASO and BSO cubes) for your FreeForm business process.
>
> The multi-cube FreeForm template replaces the two previously available templates that allowed you to create a single-cube FreeForm ASO or BSO application.

Then continue with Creating an Application, using the information in Working with Artifacts in the Application Template to modify the template according to your requirements.

> **Note:**
>
> For information on creating applications in the web, see the documentation for your business process:
>
> - Creating a Planning Application in *Administering Planning*
> - Creating a FreeForm App in *Administering FreeForm*
> - Setting Up Your Application in *Administering Planning Modules*

Videos

Your Goal	Watch This Video
Learn about creating an application in Smart View. The video shows the process of creating a Planning application using Oracle Smart View for Office, but the concepts also apply to Oracle Smart View for Office (Mac and Browser), as well as the FreeForm and Planning Modules business processes.	 Overview: Creating and Managing Planning Cloud Applications Using Smart View

Installing Smart View and the Admin Extension for Application Management

Most procedures described in this appendix are performed within Smart View, with the exception of Controlling the Display of Application Management Options in Smart View and Downloading the Application Template Zip File from the Web Interface.

> **Note:**
>
> Application management is supported for these Oracle Enterprise Performance Management Cloud business processes: Planning, Planning Modules, and FreeForm.

Before you begin, ensure the following depending on whether you plan to use Windows-based Oracle Smart View for Office, Oracle Smart View for Office (Mac and Browser), or both:

- **Windows-based Smart View:** Install Smart View.

  The Admin Extension is not required to use the application management features in Windows-based Smart View.

  See Installing Windows-based Oracle Smart View for Office

- **Smart View (Mac and Browser):** Deploy Smart View (Mac and Browser) with the Dimension Editor options selected.

  You must select the Dimension Editor options in order to work with the application templates in Smart View (Mac and Browser).

  See Deploying Oracle Smart View for Office (Mac and Browser) with the Admin Extension

Installing Windows-based Oracle Smart View for Office

To use the Windows client-based version of Smart View, see Downloading and Installing Clients in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators* for instructions on installing Smart View.

> **Note:**
>
> The Admin Extension is not required with Windows-based Smart View when working with application management.

After installation, get started by downloading and then editing the application template workbook available in Smart View or from the template download zip file in the web.

- By default, you can download the Planning application workbook template directly from Smart View.
- Or, download the application template zip file from the Downloads page of the Planning, Planning Modules, or FreeForm business process. The template zip file contains starter application template workbooks for these business processes, along with a sample Planning application workbook.

Deploying Oracle Smart View for Office (Mac and Browser) with the Admin Extension

To use the Mac- or browser-based version of Smart View, you must deploy a Smart View (Mac and Browser) manifest file with the Dimension Editor options enabled.

See the following topics in *Deploying and Administering Oracle Smart View for Office (Mac and Browser)* for instructions on deploying Smart View (Mac and Browser) with the Dimension Editor options enabled:

- Creating and Saving the Manifest File

- Enabling the Admin Extension
- Sideloading and Logging In or Deploying the Manifest File to Office 365 Users

Controlling the Display of Application Management Options in Smart View

When a new application is created in the web, in Oracle Smart View for Office, or in Oracle Smart View for Office (Mac and Browser), all application management options are displayed by default to Service Administrators in either version of Smart View.

You can choose to hide or display two of the options, **Update Application** and **Delete Application**, using the **Suppress Application Management Options in Smart View** option in Planning, Planning Modules, or FreeForm.

This setting, and the display of the application management options, applies to Service Administrators only.

To control the display of the **Update Application** and **Delete Application** options in Smart View:

1. From the Home page, click **Application**, and then click **Settings**.
2. Choose a setting for **Suppress Application Management Options in Smart View**:
   - **Yes**: Application options are not displayed in Smart View.
   - **No**: Default. Application options are displayed in Smart View.
3. Click **Save**.

> **Note:**
>
> Any time a new application is created, whether in the web, in Oracle Smart View for Office, or in Oracle Smart View for Office (Mac and Browser), the **Suppress Application Management Options in Smart View** setting defaults to **No** and the **Update Application** and **Delete Application** options will automatically be displayed. To hide the options, perform the above procedure.

Downloading the Application Templates

Related Topics

- Downloading the Template in Smart View
- Downloading the Application Template Zip File from the Web Interface

Downloading the Template in Smart View

When you're logged into Smart View as a Service Administrator, the Download Template command is available in the Smart View Panel on Windows and in the Smart View Home panel on the Mac and in the browser.

The default template that you download can be used immediately to begin creating a Planning application. Or, you can modify the template to create your own custom application.

To download the default application template from Smart View:

1. Log in to the applicable data source in Smart View.
2. Perform an action:
   - In the Smart View Panel (Windows), at the bottom of the panel, click the **Download Template** command.
   - In the Smart View Home panel (Mac and Browser), click  , and select **Download Template** from the drop-down menu.

     If prompted, allow the file to be created in your cloud repository.

   The default application template, `PlanningApplicationTemplate.xlsx`, is opened. The default template contains the basic elements required to begin creating a standard Planning application.

   You can edit the default template to create a custom application, as described in Working with Artifacts in the Application Template.

   To download the Planning Modules, FreeForm, or sample Planning templates, download the application template zip file from the web, described in Downloading the Application Template Zip File from the Web Interface.

Downloading the Application Template Zip File from the Web Interface

When you're logged into Planning, Planning Modules, or FreeForm as Service Administrator, you can download a zip file containing these application template workbook files:

- `PlanningApplicationTemplate.xlsx`: Creates a Planning application.
- `PlanningApplicationTemplate_epbcs.xlsx`: Creates a Planning Modules application.
- `PlanningApplicationTemplate_FreeForm.xlsx`: Creates a FreeForm application.
- `PlanningApplicationTemplate_sample.xlsx`: Creates the Planning sample application, Sample.

> **Note:**
>
> Starting in 22.03, a single Excel template workbook, `PlanningApplicationTemplate_FreeForm.xlsx`, allows you to create a multi-cube application (any combination of up to 12 ASO and BSO cubes) for your FreeForm business process.
>
> The multi-cube FreeForm template replaces the two previously-available templates that allowed you to create a single-cube FreeForm ASO or BSO application.

To download the application template zip file:

1. From the computer where you want to install or deploy the client software, access a business process instance.
2. On the Planning, Planning Modules, or FreeForm Home page, access **Settings and Actions** by clicking your user name at the top right corner of the screen.
3. Select **Downloads**.

4. In the **Downloads** page, Planning Extensions section, click **Download for Application Templates**.
5. Save the zip file locally or to a server location that you can access, and unzip the contents.

   You should now have these files saved:

   - `PlanningApplicationTemplate.xlsx`
   - `PlanningApplicationTemplate_epbcs.xlsx`
   - `PlanningApplicationTemplate_FreeForm.xlsx`
   - `PlanningApplicationTemplate_sample.xlsx`
6. Open the template file to work with in Excel and proceed with Creating an Application, using the information in Working with Artifacts in the Application Template to modify the template according to your requirements.

Creating an Application

From Oracle Smart View for Office and Oracle Smart View for Office (Mac and Browser), Service Administrators can create a new application based on either the downloaded template with no modifications or a modified template. In either case, the template must adhere to template rules and guidelines.

When you create an application, you can:

- Define cubes
- Define dimensions
- Add a limited amount of data
- Define security and access permissions
- Define substitution variables

Only Service Administrators can create an application and only when no application exists in the cloud business process. Once an application is created, the option is no longer available in the Smart View Panel or Smart View Home panel.

Before you begin, you should launch Excel and, from Smart View, log in to the applicable data source.

To create a new application in Smart View:

1. If you have not already done so, perform one of the following actions:
   - Download the application template for Planning as described in Downloading the Template in Smart View.

     Downloading the application template simply opens the template in Smart View.
   - Download and unzip the application template zip file as described in Downloading the Application Template Zip File from the Web Interface, then open one of the template files.
   - Open a saved template file.

   An Excel application template workbook must be open when creating an application.
2. **Optional:** Edit and then save the application template as needed.

   See Working with Artifacts in the Application Template for information on modifying the worksheets in the application template workbook.

3. With the application template workbook open in Smart View, select **Create New Application**:

   - Oracle Smart View for Office (Windows):

     In the tree in the Smart View Panel, select the server name to access the **Create New Application** command at the bottom of the panel. You can also right-click the server name to access the **Create New Application** command.

   - Oracle Smart View for Office (Mac and Browser):

     In the Smart View Home panel, click the Actions button, , and then select **Create New Application** from the drop-down menu.

   The application creation status appears in the lower left corner of Excel.

4. When the application creation process is complete, verify that the application appears in the Smart View Panel (Windows) or Smart View Home panel (Mac and Browser).

   - Oracle Smart View for Office (Windows):

     The application name is displayed in the tree in the Smart View Panel. Expand the application name to view folders and cubes.

   - Oracle Smart View for Office (Mac and Browser):

     In the Smart View Home panel, the application folders and cubes are displayed. The application name is not displayed.

   Note that only one application per cloud business process is allowed. Once an application is created, the **Create New Application** command no longer appears in the Application Management popup window.

   The application is now available to the applicable users.

   Next steps include updating the application template workbook. You can also delete the application and create a new one. See:

   - Working with Artifacts in the Application Template
   - Updating an Application in Smart View
   - Deleting an Application

Working with Artifacts in the Application Template

Related Topics

- About Working with Artifacts in the Application Template
- Application Definition
- Dimension Definition
- Attribute Dimension Definition
- Data Definition
- Substitution Variable Definition
- Security Definition
- Advanced Settings Definition

About Working with Artifacts in the Application Template

Editing the application template involves editing Excel worksheets in the application workbook.

The application template workbook contains sheets for defining application properties, cubes, dimension and member properties, substitution variables, and security. You can also load data using the template.

Application Definition

In the application template workbook, the Definition worksheet defines your application and cubes. Using the application definition worksheet provided in the template zip file as your starting point, you can edit the worksheet provided to suit your requirements. You can also use the Sample application template file as your guideline for a standard application or adapt it to create a Planning Modules or FreeForm application.

Follow this workflow for setting up a data worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define Application Properties
4. Define Cubes

Define the Worksheet Name

The worksheet name is in the Excel worksheet tab. The first tab is in the workbook is for the application definition worksheet. Ensure that the application definition worksheet is named, `Definition`. For example:

Define the Worksheet Type

In cell A1 in the application worksheet, type `Application Definition` for the worksheet type. In Figure 1, cell A1 contains the worksheet type, `Application Definition`.

Figure C-1 Worksheet Type and Application Properties Shown in Application Definition Worksheet

	A	B
1	**Application Definition**	
2		
3	**Application Name**	PBCS
4	**Description**	PBCS Standard Application
5	**Application Type**	Standard
6	**Planning Frequency**	Monthly
7	**Start Year**	2010
8	**End Year**	2019
9	**First Month of Fiscal Year**	January
10	**Weekly Distribution**	Even
11	**Main Currency**	USD
12	**Multi Currency**	No
13	**Multi Currency Type**	Standard
14	**Sandboxes**	Yes
15	**Task Flow Type**	EPM Task Manager

Define Application Properties

Application properties are the basic information needed to create an application. In general, only the Application Name, Application Description, and Application Type are required properties. For all other properties, if not specified, the default values will be provided during application creation.

> **Note:**
>
> You can create the FreeForm app with only the application properties and no dimensions, attribute dimensions, security (access permissions), or substitution variables in the workbook.

Table 1 shows the supported properties, their default values, and valid values.

Table C-1 Application Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Application Name	Yes	Text			Name of Planning application
Description	Yes	Text			Application description
Application Type	Yes	Text	Standard	**Standard**: An advanced application with two cubes and the option to add more later **Enterprise**: Also referred to as EPBCS Advanced **FreeForm**: A FreeForm application	Similar to choices on web interface
Planning Frequency	No	Text	Monthly	Monthly Weekly Quarterly Custom	
Weekly Distribution	No	Text	445	Even 445 454 544	Applicable for Monthly Planning frequency
Periods Per Year	No	Numeric	13		Only for Custom Planning Frequency

Table C-1 (Cont.) Application Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Prefix	No	Text	TP		Only for Custom Planning Frequency
Start Year	No	Numeric	2010		Year in 4 digits
End Year	No	Numeric	2019		Year in 4 digits Calculated as (*Start Year + Number of Years + 1)*
First Month of Fiscal Year	No	Text	January	Jan through Dec	Only for Monthly Planning Frequency
First Fiscal Year Start Date	No	Date			Only for Quarterly or Custom Planning frequency
Rolling Forecast	No	Boolean	Yes	True | False Yes | No	Only for Monthly Calendar
Rolling Forecast Period Duration	No	Numeric	6	1 to 36	Only when Rolling Forecast is True
Task Flow Type	No	Text	EPM Task Manager	EPM Task Manager Task List	Defaults to EPM Task Manager if entry is missing or typed incorrectly. **Note:** This option does no apply to FreeForm applications.

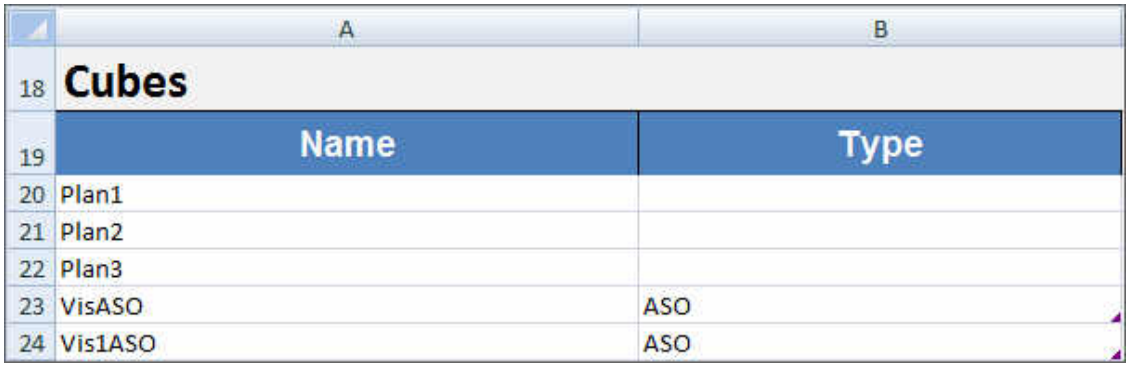
Define Cubes

In the application worksheet, the same sheet where you define the application properties, list all cubes.

To define cubes in the application worksheet, add a line with the label, `Cubes` after the application definition section. For example, in Table 2, a line with the word `Cubes` is added to line 18 of the worksheet.

In the next row, add the `Name` label to column A, and the `Type` label to column B.

Figure C-2 Cube Definition Section of Application Definition Worksheet



	A	B
18	**Cubes**	
19	**Name**	**Type**
20	Plan1	
21	Plan2	
22	Plan3	
23	VisASO	ASO
24	Vis1ASO	ASO

Table 2 shows the properties related to cube definition. Only the cube name is required. For the Type column, use BSO or ASO. Note that if the type is left blank, BSO is used by default.

All application types, Planning, Planning Modules, and FreeForm, support multiple cubes and any combination of BSO and ASO cubes.

Table C-2 Cube Properties

Header Name	Required	Comments
Name	Yes	Name of the cube
Type	BSO ASO	BSO is the default. If cell is left blank, BSO is used

Dimension Definition

In the template workbook, create a new worksheet for each dimension in your application. You can use the worksheets provided in the templates, including the Sample application template, as your guide.

Follow this workflow for setting up a dimension worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Dimension Name
4. Define Dimension Properties
5. Define Members

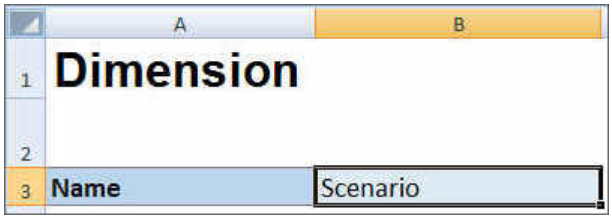
Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. For a dimension worksheet name, use the dimension name prefixed with "`Dim.`". For example, for the Scenario dimension, you would name the worksheet `Dim.Scenario`. For example:

Define the Worksheet Type

In cell A1 in the dimension worksheet, type `Dimension` for the worksheet type. In Figure 1, cell A1 contains the worksheet type, `Dimension`.

Figure C-3 Worksheet Type and Dimension Name Shown in a Dimension Worksheet of Application Template



Define the Dimension Name

In the dimension worksheet, type the dimension name. In Figure 1, cell B3 contains the dimension name, `Scenario`.

If the dimension name that you provide doesn't exist, then a new custom type dimension is automatically created when you create the application.

Define Dimension Properties

You can define dimension properties directly on the dimension worksheet. Using the Type property, you can specify user-defined names for the Account, Year, and Entity dimensions and then map them to the Accounts, Year, or Entity dimension types. You can also define any other custom dimensions as you require. The templates in the template zip file includes example Accounts, Year, and custom dimensions and their properties to guide you.

> **Note:**
>
> You can create the FreeForm app with no dimensions or any other application properties (such as attribute dimensions, security (access permissions), and substitution variables) in the workbook.

Table C-3 Dimension Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Name	Yes	Text			Name of the dimension

Table C-3 (Cont.) Dimension Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Type	No	Text		Account Period Entity	If this property is omitted, or no value is supplied, then this dimension is interpreted as a custom dimension.
Description	No	Text			An optional dimension description.
Alias	No	Text	None	The alias for the dimension	Aliases are alternate name for dimension members.
Alias Table	No	Text	No	The alias table	The default alias table for the application.
Hierarchy Type	No	Text	Not Set	Not Set Stored Dynamic Multiple	Available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be Stored.

Table C-3 (Cont.) Dimension Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Density	No	Boolean	Sparse	Dense | Sparse	Sparse dimensions lack data values for the majority of member combinations. Dense dimensions have data values for the majority of member combinations.
Two Pass Calculation	No	Boolean	No	Yes | No	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.
Apply Security	No	Boolean	No	Yes | No	Allow security to be set on the dimension members, per the security laid out in the Security worksheet in the application template.

Table C-3 (Cont.) Dimension Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Partial Share	No	Boolean	No	Yes | No	Alternate hierarchies are supported in the Entity dimension. One entity can have multiple parents, and contribute differently to each parent. These members are referred to as partially shared entities, where only a part of input data is shared across all instances of the entities.
Data Storage	No	Text	Store	Dynamic Calc and Store Dynamic Calc Store Never Share Shared Label Only	Data storage properties define where and when consolidations are stored. For example, by default, members are tagged as Store.
Performance Order	No	Numeric	None	A positive or negative number; for example: 1 -1	Evaluation order of dimensions in the application.

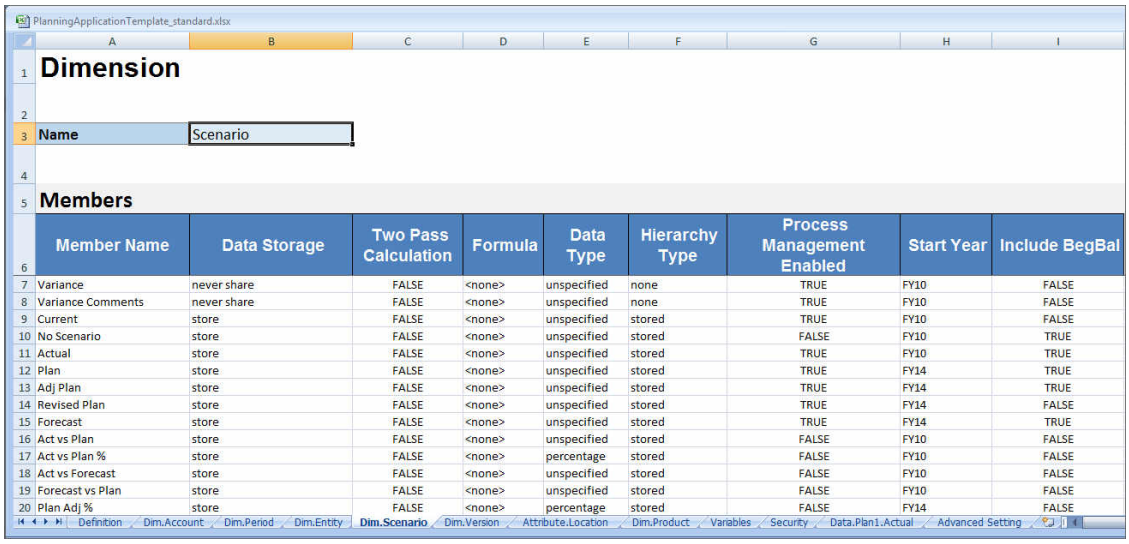
Table C-3 (Cont.) Dimension Properties

Property	Required	Value Type	Default Value	Valid Values	Comments
Evaluation Order	No	Numeric	None	A positive or negative number; for example: 1 -1	The order of your dimensions determines how your data calculations will perform. Evaluation Order enables you to specify which data type prevails when a data intersection has conflicting data types.
Display Option	No	Text	Member Name or Alias	**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.	Set application default display option.

Define Members

In the dimension worksheet, under the `Members` heading in (cell A5 in the Planning application template), add members and their properties. Your list of members should conform to the format described in Importing and Exporting Data and Metadata. You may add columns only for required properties. Any missing properties will be added based on application, cube, and dimension defaults.

Figure C-4 Example Dimension Worksheet for the Scenario Dimension in a Standard Planning Application



PlanningApplicationTemplate_standard.xlsx

Dimension

Name	Scenario

Members

Member Name	Data Storage	Two Pass Calculation	Formula	Data Type	Hierarchy Type	Process Management Enabled	Start Year	Include BegBal
Variance	never share	FALSE	<none>	unspecified	none	TRUE	FY10	FALSE
Variance Comments	never share	FALSE	<none>	unspecified	none	TRUE	FY10	FALSE
Current	store	FALSE	<none>	unspecified	stored	TRUE	FY10	FALSE
No Scenario	store	FALSE	<none>	unspecified	stored	FALSE	FY10	TRUE
Actual	store	FALSE	<none>	unspecified	stored	TRUE	FY10	TRUE
Plan	store	FALSE	<none>	unspecified	stored	TRUE	FY14	TRUE
Adj Plan	store	FALSE	<none>	unspecified	stored	TRUE	FY14	TRUE
Revised Plan	store	FALSE	<none>	unspecified	stored	TRUE	FY14	FALSE
Forecast	store	FALSE	<none>	unspecified	stored	TRUE	FY14	TRUE
Act vs Plan	store	FALSE	<none>	unspecified	stored	FALSE	FY10	FALSE
Act vs Plan %	store	FALSE	<none>	percentage	stored	FALSE	FY10	FALSE
Act vs Forecast	store	FALSE	<none>	unspecified	stored	FALSE	FY10	FALSE
Forecast vs Plan	store	FALSE	<none>	unspecified	stored	FALSE	FY10	FALSE
Plan Adj %	store	FALSE	<none>	percentage	stored	FALSE	FY14	FALSE

Definition | Dim.Account | Dim.Period | Dim.Entity | Dim.Scenario | Dim.Version | Attribute.Location | Dim.Product | Variables | Security | Data.Plan1.Actual | Advanced Setting

Attribute Dimension Definition

In the template workbook, create a new worksheet for each attribute dimension in your application. You can use the attribute dimension worksheet provided in the template for the Vision application as your guideline.

Follow this workflow for setting up an attribute dimension worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Attribute Dimension Name
4. Define the Attribute Dimension Type
5. Define the Base Dimension
6. Define Attribute Members

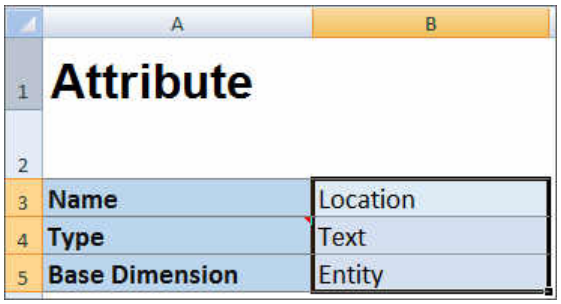
Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. For the worksheet name, use the dimension name prefixed with "`Attribute.`". For example, for the Location attribute dimension, you would name the worksheet `Attribute.Location`. For example:

Define the Worksheet Type

In cell A1 in the attribute dimension worksheet, type `Attribute` for the worksheet type. In Figure 1, cell A1 contains the worksheet type, `Attribute`.

Figure C-5 Worksheet Type and Attribute Dimension Properties Shown in an Attribute Dimension Worksheet of Application Template



Define the Attribute Dimension Name

In the attribute dimension worksheet, type the dimension name. In Figure 1, cell B3 contains the attribute dimension name, `Location`.

Define the Attribute Dimension Type

In the attribute dimension worksheet, type the attribute dimension type. In Figure 1, cell B4 contains the attribute dimension type, `Text`.

Valid attribute dimension types are:

- Text
- Numeric
- Boolean
- Date

Define the Base Dimension

In the attribute dimension worksheet, type the base dimension for the attribute dimension. In Figure 1, cell B5 contains the base dimension, `Entity`.

Define Attribute Members

In the dimension worksheet, under the `Members` heading in cell A5, add attribute members and their properties. Your list of attribute members should conform to the format described in Importing and Exporting Data and Metadata. You may add columns only for required properties. Any missing properties will be added based on application, cube, and dimension defaults.

Data Definition

With the application creation template, you can load up to 1,000 rows of data using the Essbase data format.

In the template workbook, create a worksheet for your application data, using the data worksheet provided in the template for the Sample application as your guideline. Or, edit the worksheet provided in the template.

Follow this workflow for setting up a data worksheet in your application template workbook:

1. Define the Worksheet Name

2. Define the Worksheet Type
3. Define the Cube Name
4. Define Data

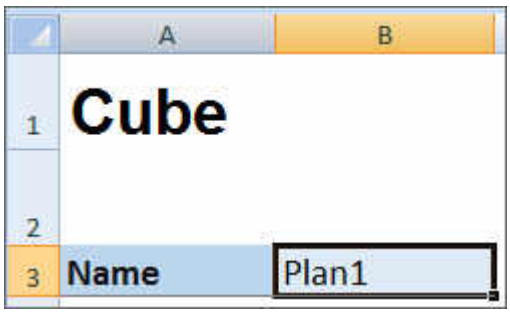
Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. For the worksheet name, use the cube name where data will be loaded, prefixed with `Data.`. For example, to load data to the Plan1 cube, you would name the worksheet `Data.Plan1` to indicate that you're loading data to Plan1. After the required `Data.` prefix, you may add any other indicators to the name to help you identify the data. For example, to show that you're loading actual data versus historical data to Plan 1, you would name the worksheet `Data.Plan1.Actual`.

Define the Worksheet Type

In cell A1 in the data worksheet, type `Cube` to signify that data in this sheet will be loaded to a cube. In Figure 1, cell A1 contains `Cube` to indicate that the worksheet will contain data for loading to a cube.

Figure C-6 Worksheet Type and Cube Name Shown in Data Worksheet of Application Template



Define the Cube Name

In the data worksheet, type the name of the cube to which to load data. In Figure 1, cell B3 contains the cube name, `Plan1`.

Define Data

Define data in the Essbase columnar format. Use "Dimension" as the header for each member intersection.

Substitution Variable Definition

In the application template workbook, include a worksheet for your substitution variables, using the substitution variable definition worksheet provided in the template for the Sample application as your guideline.

Substitution variables can be defined for Standard, Enterprise, and FreeForm applications.

Follow this workflow for setting up a substitution variable worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Substitution Variables

Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. Name the substitution variable definition worksheet, `Variables`. For example:

Define the Worksheet Type

In cell A1 in the data worksheet, type Substitution Variables to signify that data in this sheet will be loaded to a cube. In Figure 1, cell A1 contains `Substitution Variables` to indicate that the worksheet will contain substitution variable information for loading to the specified cubes in the application.

Figure C-7 Worksheet Type and Substitution Variables Shown in Variables Worksheet of Application Template

	A	B	C
1	**Substitution Variables**		
2			
3	**Cube**	**Name**	**Value**
4	All Cubes	CurrYr	FY12
5	Plan3	CurrYr	FY11
6	Plan2	NextYr	FY13
7	All Cubes	CurVersion	Working

Define the Substitution Variables

In the Variables worksheet, using Figure 1 for reference, add information for each substitution variable in the following format:

- In the **Cube** column, enter the name of the cube to which the substitution variable will apply. For example:
  - If a variable applies to all cubes, enter `All Cubes`
  - If a variable applies to a specific cube, enter the cube name; for example, enter `Plan3`

  - If a variable applies to more than one cube, but not all cubes, list each cube name on its own row individually; for example, if a variable applies to both Plan2 and Plan3 only, add an entry for `Plan2` on one row, and an entry for `Plan3` on another row.

- In the **Name** column, enter the substitution variable name. In Figure 1, the substitution variable names are `CurrYr`, `NextYr`, and `CurVersion`.
- In the **Value** column, enter the value to be used for the substitution variable.

Security Definition

In the application template workbook, include a worksheet for user permissions, using the security definition worksheet provided in the template for the Vision application as your guideline.

Follow this workflow for setting up a user permissions worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define Security Properties

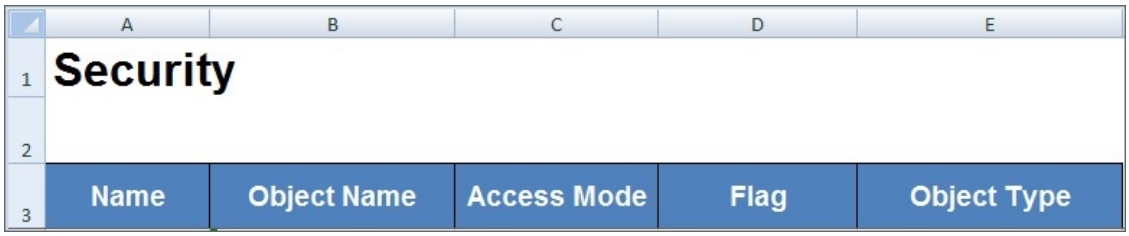
Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. Name the security worksheet where you'll define user permissions, `Security`. For example:

Define the Worksheet Type

In cell A1 in the data worksheet, type `Security` to signify that data in this sheet will be loaded to a cube. In Figure 1, cell A1 contains `Security` to indicate that the worksheet will list all user access permissions for the application.

Figure C-8 Worksheet Type Shown in Security Worksheet of Application Template



Define Security Properties

Security properties include the user name and other user access information.

Table 1 shows the supported security properties.

Table C-4 Security Properties

Property Name	Description
Name	Name of the user or group
Object Name	Artifact name
Access Mode	Permission granted. Default is NONE. Valid values: • READ • WRITE • READWRITE • LAUNCH (valid when Object Type is Rule) • NOLAUNCH (valid when Object Type is Rule) • NONE: Default
Flag	Member function to be used while applying the access. Default is MEMBER. Valid values: • MEMBER: Default • CHILDREN • @ICHILDREN • @DESCENDANTS • @IDESCENDANTS
Object Type	Artifact object type. Default is SL_DIMENSION (Dimension/Member). Valid values: • SL_FORM - Form • SL_COMPOSITE - Composite Form, Infolet, or Dashboard • SL_TASKLIST - Tasklist • SL_CALCRULE - Rule • SL_FORMFOLDER - Form Folder • SL_CALCFOLDER - Rule Folder • SL_DIMENSION - Dimension/Member: Default • SL_CALCTEMPLATE - Template • SL_REPORT - Management Report • SL_REPORTSSHOT - Management Report Snapshot

Advanced Settings Definition

In the application template workbook, include a worksheet for advanced settings, using the advanced settings worksheet provided in the template for the Sample application as your guideline.

Advanced settings are:

- Evaluation order
- Performance settings
- Dimension properties such as hierarchy type, two-pass calculations, and data storage options
- Density (for aggregate storage applications only)
- Dimension assignments to cubes

Follow this workflow for setting up an advanced settings worksheet in your application template workbook:

1. Define the Worksheet Name

2. Define the Worksheet Name
3. Define Evaluation Order
4. Define Performance Settings
5. Define Dimension Settings
6. Define Density Settings
7. Assign Dimensions to Cubes

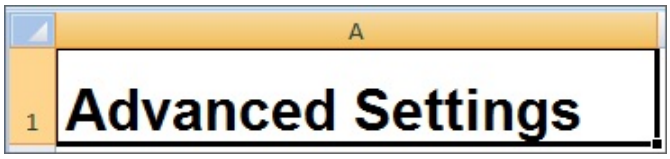
Define the Worksheet Name

Name the advanced settings definition worksheet, `Advanced Settings`. For example:

Define the Worksheet Type

In cell A1 in the data worksheet, type `Advanced Settings` to signify that this sheet contains the advanced, or miscellaneous settings for the application. In Figure 1, cell A1 contains the worksheet type, **Advanced Settings**, to indicate that the worksheet will contain advanced settings for the application.

Figure C-9 Worksheet Type Shown in Advanced Settings Worksheet of Application Template



Define Evaluation Order

In the Advanced Settings worksheet, define the evaluation order of dimensions in the "Evaluation Order" section. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns. The dimensions in each cube can have a different evaluation order.

3	Evaluation Order	Cubes	
4	Dimensions	Plan1	PBCS
5	Account	3	
6	Period		
7	Entity	1	
8	Year		
9	Scenario	2	
10	Version	1	
11	HSP_View		

Define Performance Settings

For Planning applications only.

In the Advanced Settings worksheet, define dimension storage property of each dimension for each cube in the "Performance Settings" section. Valid values are Dense or Sparse. Each cube requires at least one dense dimension. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns. The sparse or dense value of each dimension can differ between cubes.

14	Performance Setting	Cubes				
15	Dimensions	Plan1	Plan2	Plan3	VisASO	Vis1ASO
16	Account	Dense	Dense	Dense	Dense	Dense
17	Period	Dense	Dense	Dense	Dense	Dense
18	Entity	Sparse	Sparse	Sparse	Sparse	Sparse
19	Year	Sparse	Sparse	Sparse	Sparse	Sparse
20	Scenario	Sparse	Sparse	Sparse	Sparse	Sparse
21	Version	Sparse	Sparse	Sparse	Sparse	Sparse
22	HSP_View	Sparse	Sparse	Sparse	Sparse	Sparse

Define Dimension Settings

In the Advanced Settings worksheet, define dimensions properties in the "Dimension Settings" section. Dimensions are listed in the first column under the heading "Dimensions". Dimension properties are listed in subsequent columns. See Table 1 for the list of dimension properties

Table C-5 Dimension Properties Used in Dimension Settings Section of the Advanced Settings Worksheet

Dimension Property	Description or Valid Values
Description	Optional descriptive text
Alias Table	Optional alias table
Hierarchy Type	• Dynamic • Store
Two Pass Calculation	Yes | No
Apply Security	Yes | No
Data Storage	• Store • Dynamic Calc and Store • Dynamic Calc • Never Share • Label Only • Shared
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. • Member Name • Alias • Member Name:Alias • Alias:Member Name

You can also use Figure 2 for reference in adding information to the Dimension Settings section.

Figure C-10 Dimension Settings Section of the Advanced Settings Worksheet

25	Dimension Settings							
26	Properties	Description	Alias Table	Hierarchy Type	Two Pass Calculation	Apply Security	Data Storage	Display Option
27	Account							
28	Entity				Yes		Never Share	
29	Period			Dynamic				
30	Year							
31	Scenario							
32	Version							

Define Density Settings

For Planning Modules applications only.

In the Advanced Settings worksheet, define the dimension storage property of each dimension for each cube in the "Density" section. Each cube requires at least one dense dimension. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns. The sparse or dense value of each dimension can differ between cubes.

25	Density	Cubes	
26	Dimensions	Plan1	EPBCS
27	Account	Dense	Dense
28	Currency	Sparse	Sparse
29	Entity	Sparse	Sparse
30	HSP_View	Sparse	Sparse
31	Period	Dense	Dense
32	Scenario	Sparse	Sparse
33	Version	Sparse	Sparse
34	Years	Sparse	Sparse

Assign Dimensions to Cubes

In the Advanced Settings worksheet, assign dimensions to cubes in the "Dimension Valid for" section. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns.

For each dimension and cube intersection:

- Type `Yes` to include the dimension in the cube.
- Type `No` or leave the cell blank to exclude the dimension from the cube.

35	Dimension Valid For	Cubes				
36	Dimensions	Plan1	Plan2	Plan3	VisASO	Vis1ASO
37	Account	Yes	Yes	Yes	Yes	Yes
38	Entity	Yes	Yes	Yes	Yes	Yes
39	Period	Yes	Yes	Yes	Yes	
40	Year	Yes	Yes	Yes	Yes	
41	Scenario	Yes	Yes	Yes		
42	Version	Yes	Yes	Yes		

Updating an Application in Smart View

From Oracle Smart View for Office and Oracle Smart View for Office (Mac and Browser), Service Administrators can update Planning, Planning Modules, or FreeForm applications based on either the downloaded template or a modified copy of one of the templates. The modified template must always adhere to template guidelines (described in Working with Artifacts in the Application Template).

When you update an application, you can:

- Define new or modify existing dimensions
- Add or modify a limited amount of data
- Define new or modify existing access permissions
- Define new or modify existing substitution variables
- Modify advanced application setting

Only Service Administrators can update an application.

Before you begin, you should launch Excel and, from Smart View, log in to the applicable data source.

To update an application in Smart View:

1. If you have not already done so, open the modified application template file.
2. In the Smart View Panel (Windows) or Smart View Home panel (Mac and Browser), perform a task:
   - Oracle Smart View for Office: Select the application name and, in the Action Panel, select **Application Management**.

     The application name is displayed in the tree in the Smart View Panel.
   - Oracle Smart View for Office (Mac and Browser): In the Smart View Home panel, click the Actions button, , and select the **Application Management** command in the drop-down list.

     You can click the Actions button, , from any folder in the library tree.
3. In the popup dialog, select **Update Application**.

   The application update status appears in the lower left corner of Excel.
4. When the application update process is complete, access the application, either in the web application or in Smart View, and verify your changes.

Deleting an Application

Service Administrators may delete applications from Oracle Smart View for Office and Oracle Smart View for Office (Mac and Browser).

Before you begin, you should launch Excel and, from Smart View, log in to the applicable data source.

To delete an application from Smart View:

1. In the Smart View Panel (Windows) or Smart View Home panel (Mac and Browser), perform a task:
   - Oracle Smart View for Office: Select the application name and, in the Action Panel, select **Application Management**.

     The application name is displayed in the tree in the Smart View Panel.
   - Oracle Smart View for Office (Mac and Browser): In the Smart View Home panel, click the Actions button, , and select the **Application Management** command in the drop-down list.

     You can click the Actions button, , from any folder in the library tree.
2. In the popup dialog, select **Delete Application**, and confirm in the subsequent dialog.
3. In the Smart View Panel (Windows) or Smart View Home panel (Mac and Browser), verify that the application is removed.

Planning Admin Extension and Office AutoCorrect

> **Note:**
>
> The information in this topic applies to the Planning Admin Extension on the Windows-based version of Oracle Smart View for Office; this information does not apply to the Planning Admin Extension in Oracle Smart View for Office (Mac and Browser).

In the Planning Admin Extension on Windows-based versions of Excel, Office AutoCorrect can affect member editing in Smart View.

When editing member properties, such as member names, member descriptions, aliases, or member formulas of the data type Text, you may have names that begin with two initial capital letters; for example

```
DGreen
```

Excel will autocorrect this name to one initial capital; for example:

```
Dgreen
```

In Excel, you can set an AutoCorrect option to retain two initial capital letters in a name.

To retain two initial capital letters in a name:

1. In Excel, access the **Excel Options** dialog.

2. Select **Proofing**, and then click the **AutoCorrect Options** button.
3. In the **AutoCorrect** dialog, clear the **Correct TWo INitial CApitals** check box.
4. Click **OK** to close the **AutoCorrect** dialog, and then click **OK** to close the **Excel Options** dialog.

The change takes effect immediately. There is no need to restart Excel.

D

Using Smart View to Import and Edit Application Metadata

Understand how to work with application metadata, install the Admin Extension, and use Smart View grids. Also learn about dimensions, members, application members, and shared members in Smart View as well as how to refresh databases.

Related Topics

- About Using Smart View to Work with Application Metadata
- Installing Smart View and the Admin Extension for Editing Dimensions
- Using Smart View Grids to Import and Edit Application Metadata
- Importing Dimensions in Smart View
- Editing Members in Smart View
- Adding Application Members in Smart View
- Moving Members in Smart View
- Working with Attribute Dimensions
- Designating Shared Members in Smart View
- Refreshing Databases
- Planning Admin Extension and Office AutoCorrect

About Using Smart View to Work with Application Metadata

Applies to: Enterprise Profitability and Cost Management, Financial Consolidation and Close, FreeForm, Planning, Tax Reporting

Service Administrators can use the Planning Admin Extension with Oracle Smart View for Office and Oracle Smart View for Office (Mac and Browser) to quickly import and edit dimension and member metadata.

Note:

- You must have Service Administrator privileges to import and edit dimensions in Smart View and Smart View (Mac and Browser).
- All procedures described in this appendix are performed within Smart View using the Planning Admin Extension.
  - To use the Windows client-based version of Smart View, see *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators* for instructions on installing the extension.
  - To use the Mac- or browser-based version of Smart View, see the *Deploying and Administering Oracle Smart View for Office (Mac and Browser)* for instructions on deploying the extension.

Installing Smart View and the Admin Extension for Editing Dimensions

Before you begin editing dimensions in Oracle Smart View for Office, ensure the following depending on whether you plan to use Windows-based Smart View, Oracle Smart View for Office (Mac and Browser), or both:

- **Windows-based Smart View**Install Smart View and the Planning Admin Extension.

  See Installing Oracle Smart View for Office and the Planning Admin Extension
- **Smart View (Mac and Browser):** Deploy Smart View (Mac and Browser) with the Dimension Editor options enabled.

  See Deploying Oracle Smart View for Office (Mac and Browser) with the Admin Extension

Installing Oracle Smart View for Office and the Planning Admin Extension

To use the Windows client-based version of Smart View, see *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*, Downloading and Installing Clients for instructions on installing Smart View.

Then install the Admin Extension.

> **Note:**
>
> Prior to 22.07, the Admin Extension file was in MSI format. Starting in 22.07, the Admin Extension file is in SVEXT format.
>
> If you installed the Admin Extension prior to 22.07, then you must first uninstall the MSI version of the extension from the Windows Control Panel, Programs and Features. If you keep the MSI version on your machine and also install the SVEXT version, then the Smart View extension update process will not recognize updates to the SVEXT version of the Admin Extension. You may continue to use the MSI version of the Admin Extension that you previously installed. However, the MSI version of the extension will not be updated in the future so will not contain any future bug fixes or enhancements. For this reason, Oracle recommends moving to the Admin Extension in SVEXT format.

> **Tip:**
>
> - Before proceeding with installing the Admin Extension, Oracle recommends that you first clear your browser cache.
> - Enterprise Profitability and Cost Management: The Admin Extension is supported starting with Smart View 22.100.

To install the Admin Extension:

- Download the extension from the Downloads page of your business process. See *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*, Downloading and Installing Clients for instructions on installing the Planning Admin Extension.
- Download and install using the extension installation and update process in Smart View. See Installing Extensions in *Working with Oracle Smart View for Office*.

Either method will download and install the Admin Extension in SVEXT format.

Deploying Oracle Smart View for Office (Mac and Browser) with the Admin Extension

To use the Mac- or browser-based version of Smart View, you must deploy a Smart View manifest file with the Admin Extension options enabled.

See the following topics in the *Deploying and Administering Oracle Smart View for Office (Mac and Browser)* for instructions on deploying Smart View with the Admin Extension features.

- Creating and Saving the Manifest File
- Enabling the Admin Extension
- Sideloading and Logging In or Deploying the Manifest File to Office 365 Users

Using Smart View Grids to Import and Edit Application Metadata

The Smart View grid allows you to rapidly add, edit, and move the members of a dimension.

- About the Smart View Grid

- The Smart View Grid and Ribbon Display
- Guidelines for Using the Smart View Grid
- Default Metadata Dimension Member Properties

About the Smart View Grid

In Oracle Smart View for Office and in Oracle Smart View for Office (Mac and Browser), the grid consists of two dimensions:

- An Oracle Enterprise Performance Management Cloud dimension on one axis (Planning, Financial Consolidation and Close, Tax Reporting)
- A metadata dimension on another axis

A metadata dimension represents a flat list of metadata members. For each dimension, a predefined set of metadata members is presented by default. Each metadata member corresponds to a specific member property valid for the dimension that was imported onto the grid. Instead of containing numeric values, the metadata grid data cells hold the specific property values. To add metadata dimension member properties that are not on the grid by default, use the **Member Selection** dialog in Smart View.

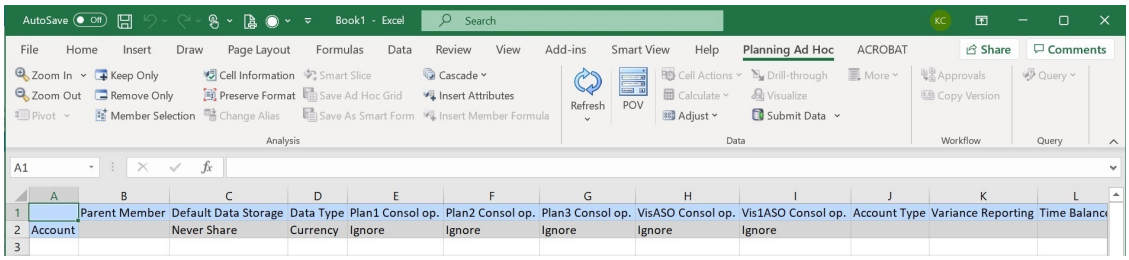
The Smart View Grid and Ribbon Display

The Smart View grid displays the dimension with member names in rows and member properties in columns. When you first import a dimension for editing, a predefined set of member properties is displayed by default in the grid.

Dimension Grid in Smart View

In Figure 1, you can see an example of the default member properties for the Vision Account dimension in Oracle Smart View for Office.

Figure D-1 Dimension Imported into Smart View for Office Grid with the Planning Ad Hoc Ribbon Displayed



Note that in Figure 1, the Planning Ad Hoc ribbon is displayed. Depending on the data source to which you are connected, the provider ad hoc ribbon for that data source is displayed. For example, if you are connected to an Enterprise Profitability and Cost Management data source, then the EPCM Ad Hoc ribbon is displayed.

The ad hoc ribbon that is displayed is the same as the provider ad hoc ribbon displayed for a conventional Smart View ad hoc grid, only with fewer options enabled.

For all dimensions, including attribute dimensions, these are the valid options in the provider ad hoc ribbon:

- Zoom In
- Zoom Out
- Keep Only
- Remove Only
- Member Selection
- Refresh
- Submit Data

The following options may appear enabled, but are not valid for metadata grids: Preserve Format, Cascade, Insert Attributes, POV, and Adjust.

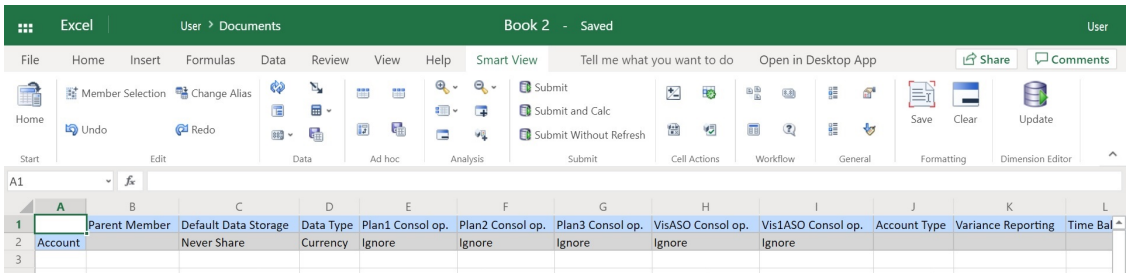
You can add metadata dimension member properties using the **Member Selection** dialog box, accessed from the ribbon, or accessed by right-clicking a property in the grid, then selecting **Smart View**, and then **Member Selection**.

For information on using the **Member Selection** dialog box in Smart View, see Selecting Members from the Member Selector in *Working with Oracle Smart View for Office 22.100*.

Dimension Grid in Smart View (Mac and Browser)

In Figure 2, you can see an example of the default member properties for the Vision Account dimension in Oracle Smart View for Office (Mac and Browser), running in the Chrome browser.

Figure D-2 Dimension Imported into Smart View (Mac and Browser) Grid with the Smart View Ribbon Displayed



Note that in Figure 2, the Smart View ribbon is displayed. This is the default ribbon name for an Oracle Smart View for Office (Mac and Browser) deployment. You may specify a different name for the ribbon. When importing a dimension on to a grid, all ribbon options appear enabled, but only a subset are applicable to a dimension grid.

For all dimensions, including attribute dimensions, these are the valid options in the ribbon:

- Zoom In
- Zoom Out
- Keep Only
- Remove Only
- Member Selection
- Refresh
- Update

You can add metadata dimension member properties using the **Member Selection** dialog box, accessed from the ribbon.

For information on using the **Member Selection** dialog box in Oracle Smart View for Office (Mac and Browser) to add metadata member properties to the grid, see Selecting Members from the Member Selector in *Working with Oracle Smart View for Office (Mac and Browser)*.

Guidelines for Using the Smart View Grid

The following guidelines will assist you in using the Smart View grid to edit dimension metadata:

- Use the Planning Admin Extension to add dimension members, and to edit and update dimension and member properties.
- The Planning Admin Extension cannot be used to:
  - Delete or remove dimensions (that is, the dimension root member) or members.
  - Edit dimension or member names.
- Data cell values can be textual or enumeration or numerical.
- The Parent Member is used to specify or modify the parent/child relationship.
- The position of a member in a grid doesn't necessarily represent the actual position of siblings in the outline.
- Each metadata grid must be linked to a corresponding dimension.
- Columns for each dimension are based on the corresponding set of member properties available in the dimension editor.

  When you first import a dimension, a predefined set of metadata dimension member properties is displayed. See Default Metadata Dimension Member Properties for a list of those member properties.
- After a metadata grid is opened, it can't be relinked to a different dimension.
- The corresponding valid set of metadata members is specific to each dimension.
- Dimension members are valid for corresponding dimensions only.
- Plan type-specific properties will only display in the grid for Plan types that a dimension is enabled for. This includes properties such as consolidation operator, member formula, data storage, and others. When a dimension is enabled for Plan types, that means that the **Valid for Cubes** setting for a dimension has one or more Plan types selected. If a dimension contains Plan type-specific properties, but a Plan type is not selected, then the data cell for the related Plan type-specific properties will read, "Not used for Cube."
- For the Time Period dimension, Data Storage is the only editable property.
- The following functionality isn't available in Smart View grids with dimension metadata:
  - Pivot
  - Pivot to POV
  - Cell Text
  - Cell Notes
  - Supporting Details

Default Metadata Dimension Member Properties

When you first import a dimension in the Planning Admin Extension, a predefined set of dimension metadata member properties is displayed by default. Use the Member Selection dialog box to add other metadata member properties to the grid.

The sections below list the default member properties for the Planning dimensions.

For information on using the Member Selection dialog box in Oracle Smart View for Office to add other member properties to the grid, see "Selecting Members from the Member Selector" in *Oracle Smart View for Office User's Guide*.

Account Dimension

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
- Account Type
- Variance Reporting
- Time Balance
- Default Alias Table
- Default Formula

Entity Dimension

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
- Base Currency
- Default Alias Table

Custom Dimensions

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
- Default Alias Table

Scenario Dimension

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes

- Start Yr.
- End Yr.
- Start Period
- End Period
- Default Alias Table
- Default Formula

Version Dimension

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
- Type
- Default Alias Table
- Default Formula

Period and Years Dimensions

- Leave as is or limit to Parent Member
- Default Data Storage – Note that for the Time Period dimension, Data Storage is the only editable property
- Data Type
- Default Alias Table

Currency Dimension

- Parent Member
- Default Data Storage
- Data Type
- Symbol
- Predefined Symbol
- Scale
- Precision
- Triangulation Currency
- Reporting Currency
- Thousands Separator
- Decimal Separator
- Negative Sign
- Negative Color
- Default Alias Table
- Default Formula

HSP_View Dimensions (Sandbox dimensions)

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
- Default Alias Table

Importing Dimensions in Smart View

Importing dimensions into the Smart View grid refers to placing the dimension and its members, and their respective properties, on the grid in preparation for editing.

By importing a dimension into Oracle Smart View for Office or Oracle Smart View for Office (Mac and Browser), you can rapidly add, edit, and move the members of the dimension.

> **Note:**
>
> You must have Service Administrator privileges to import dimensions in Smart View.

See:

- Importing Dimensions in Smart View for Office
- Importing Dimensions in Smart View (Mac and Browser)

Importing Dimensions in Smart View for Office

> **Note:**
>
> You must have Service Administrator privileges to import dimensions in Oracle Smart View for Office.

To import a dimension into a Smart View grid:

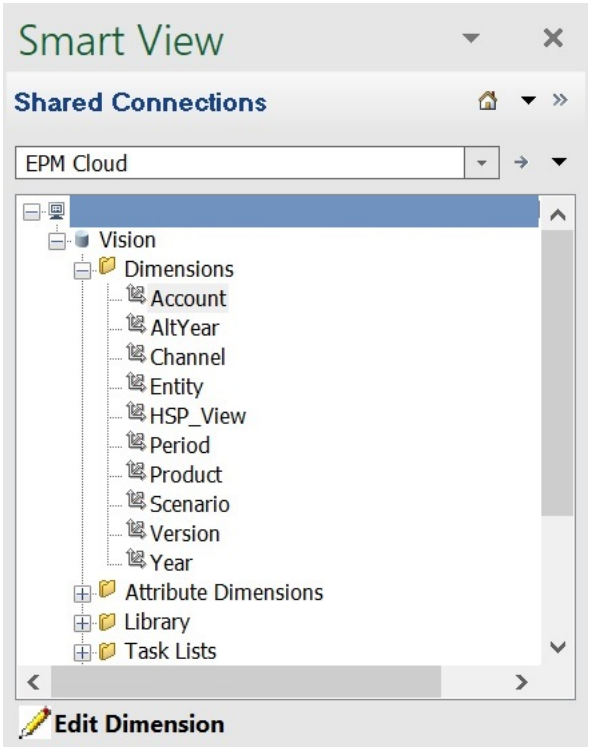
1. From the Smart View ribbon, click Panel.
2. In the Smart View Panel, click Shared Connections or Private Connections, and provide your user name, and password in the login windows.

   If you chose Private Connections, select the private connection in the drop-down list of connections, then provide your login credentials.
3. From the drop-down list in the Smart View Panel, select your business process (for example, **Planning**) or **EPM Cloud**.
4. In the tree in the Smart View Panel, expand the **Dimensions** folder to view the application dimensions in the folder.

   If attribute dimensions are defined, they are displayed in the **Attribute Dimensions** folder.

If a Time Period dimension is defined, it's displayed in the Dimensions folder. Note that for the Time Period dimension, Data Storage is the only editable property.

In the following example, the Dimensions folder is expanded and contains 10 dimensions.



5. Right-click a dimension name and select **Edit Dimension**.

   Alternatively, select a dimension and then, in the Action Panel, select **Edit Dimension**.

   See Dimension Grid in Smart View for an illustration of the initial grid for the Account dimension.

Importing Dimensions in Smart View (Mac and Browser)

Before you begin to work with dimensions in Oracle Smart View for Office (Mac and Browser), you should either sideload or deploy Smart View in your environment. Instructions are in *Deploying and Administering Oracle Smart View for Office (Mac and Browser)*:

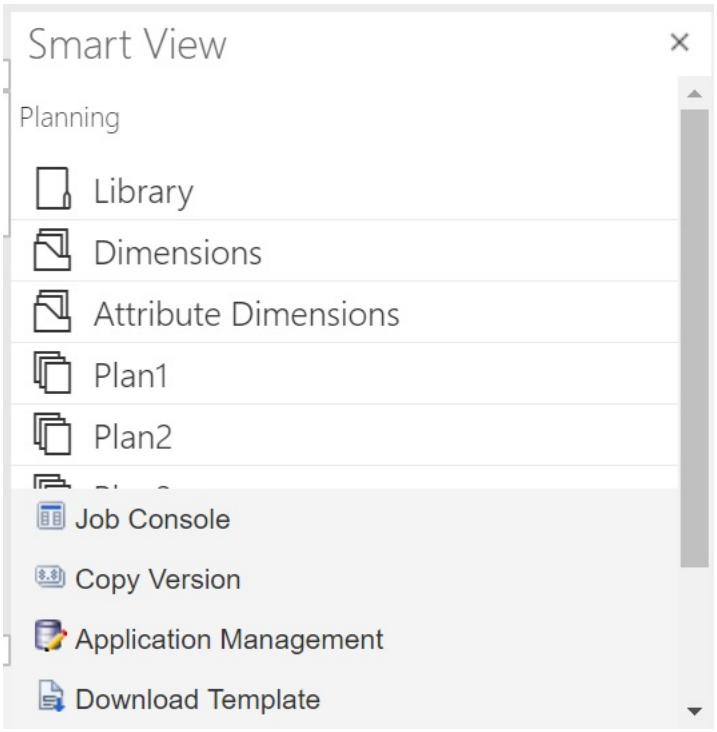
- Sideloading and Logging In
- Deploying the Manifest File to Office 365 Users

To import a dimension into an Oracle Smart View for Office (Mac and Browser) grid:

1. From the Smart View ribbon, click **Home**.

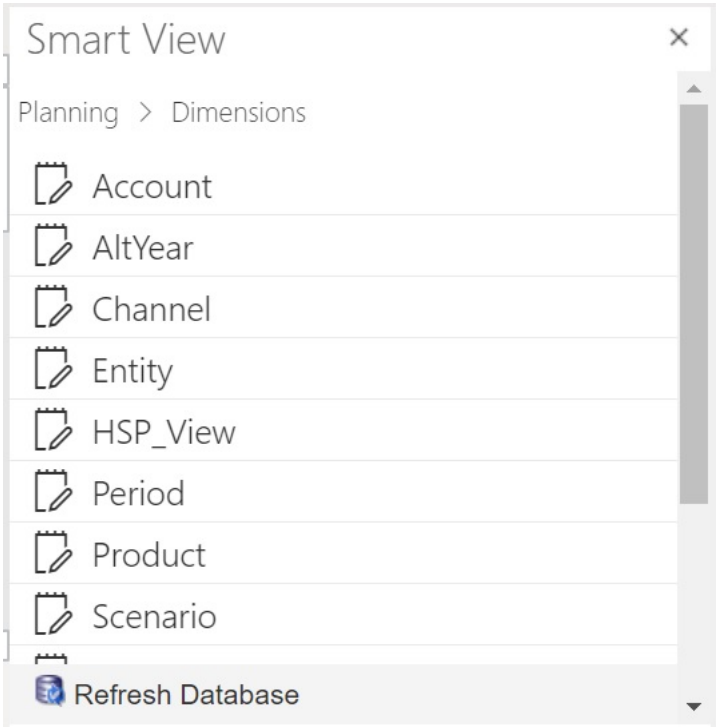
   In the **Home** panel, dimensions are displayed in the **Dimensions** folder. If attribute dimensions are defined, they are displayed in the **Attribute Dimensions** folder.

   If a Time Period dimension is defined, it's displayed in the Dimensions folder. Note that for the Time Period dimension, Data Storage is the only editable property.



2. In the **Smart View Home** panel, click the **Dimensions** folder to expand it and view the application dimensions in the folder.

   In the following example, the Dimensions folder is expanded and shows eight of the 10 dimensions in the Vision application.



3. Click a dimension name to import the dimension on to the grid.

See Dimension Grid in Smart View (Mac and Browser) for an illustration of the initial grid for the Account dimension.

4. **Optional:** Click the provider link in the path at the top of the panel to return to the main page of the Home panel

   For example, in the figure in step 2, you would click **Planning** to return to the main page of the Home panel.

Editing Members in Smart View

The Smart View grid allows you to rapidly edit the properties of members of a dimension.

To edit member properties in Smart View:

1. Within Oracle Smart View for Office or Oracle Smart View for Office (Mac and Browser), import a dimension into a Smart View grid (see Importing Dimensions in Smart View).
2. Highlight a member property in the grid.

   Note that for the Time Period dimension, Data Storage is the only editable property.
3. In the drop-down menu, select a value.

> **Note:**
>
> Modified cells are displayed in a different color.

4. To save the grid, perform a task:
   - Oracle Smart View for Office: In the Planning Ad Hoc ribbon, click **Submit Data**.
   - In Oracle Smart View for Office (Mac and Browser), in the Smart View ribbon, click **Update**.

> **Note:**
>
> You can save changes to multiple members and properties in the same **Submit Data** or **Update** operation. If the operation fails for one member, the application server will stop the operation and not save any changes.

Adding Application Members in Smart View

The Smart View grid allows you to rapidly add members to a dimension.

- Adding Members in Smart View
- Guidelines for Adding Members in Smart View

Videos

Your Goal	Watch This Video
Learn about adding Entity dimension members in Oracle Smart View for Office.	Managing Entity dimension members in Oracle® Smart View for Office
Learn about adding Account dimension members in Smart View.	Managing Account dimension members in Oracle® Smart View for Office
Learn about adding Scenario dimension members in Smart View.	Managing Scenario dimension members in Oracle® Smart View for Office

Adding Members in Smart View

The Smart View grid allows you to rapidly add members to a dimension.

To add members in Smart View:

1. Within Oracle Smart View for Office or Oracle Smart View for Office (Mac and Browser), import a dimension into a Smart View grid (see Importing Dimensions in Smart View).

   Note that Oracle Smart View for Office does not support adding members to the Time Period dimension.

2. Enter the name of a new member in the name column.

> **Note:**
>
> To add a member with a numeric name, place a single quote (') in front of the numeric member name to inform the system that the number is a member name and not data. For example, to add a member named `123`, enter `'123`.

3. From the ribbon, click **Refresh**.

   New members are marked by an asterisk (*) in the grid after you perform the Refresh. A default set of properties is automatically applied to the new member. The default Parent Member is the root member of the dimension.

4. **Optional:** To change any property from the default value (the root member), highlight the appropriate cell in the grid, and then from the drop-down menu, select another value. (See Editing Members in Smart View.)

> **Note:**
>
> Perform a Refresh *before* you modify any properties in your new member. The Refresh operation will replace any changed values with default values from the server.

5. To save the grid, perform a task:

   - Oracle Smart View for Office: In the Planning Ad Hoc ribbon, click **Submit Data**.
   - In Oracle Smart View for Office (Mac and Browser), in the Smart View ribbon, click **Update**.

> **Note:**
>
> - To modify the properties of the new member, see Editing Members in Smart View.
> - In Oracle Smart View for Office, member properties, including member names, can be localized based on the particular Locale set by in the Smart View add-in.
> - You can save changes to multiple members and properties in the same **Submit Data** or **Update** operation. If the operation fails for one member, the application server will stop the operation and not save any changes.

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** or **Update** 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** or **Update** operation.
- Default properties are automatically applied by the application to a new member. The specific default properties are based upon those of the dimension.
- When adding new currency members to a Currency dimension in a simplified application, the parent member for the new currency defaults to the "`Currency`" dimension member. It should default to the "`Input Currencies`" member. You can correct this by doing the following:
  1. Replace "`Currency`" with "`Input Currencies`" in the `Parent Member` column for each new currency member that you added.
  2. Perform a **Submit Data** or **Update**.
  3. Click **Refresh**.
  4. Verify that for each new currency you added, the entry for the `Parent Member` column is "`Input Currencies`."

Moving Members in Smart View

The Smart View grid allows you to rapidly move members from one parent to another within a dimension.

To move a member in Oracle Smart View for Office:

1. Within Smart View or Oracle Smart View for Office (Mac and Browser), import a dimension into a Smart View grid (see Importing Dimensions in Smart View for Office.)
2. Highlight the member in the **Parent Member** column in the grid.

> **Note:**
>
> An empty Parent Member property causes the value of the Parent Member to default to the root member.

3. Enter a parent name at the intersection of the **member name** and **Parent** column.
4. Click **Submit Data** to save the grid.

> **Note:**
>
> You can save moves of multiple members or subtrees in the same **Submit Data** operation. If the **Submit Data** operation fails for one member, the application server stops the operation and doesn't save changes.

Guidelines for Moving Members in Smart View

- An empty Parent Member in the grid denotes a root member in the dimension.
- The value of a Parent Member follows the rules applicable to the corresponding value in the metadata load file used by Smart View.

Working with Attribute Dimensions

Related Topics

- Adding Attribute Dimension Members in Smart View
- Associating Attribute Dimension Members with Dimension Members

Adding Attribute Dimension Members in Smart View

You create attribute dimensions in the web interface. You can then add members to attribute dimensions using the Admin Extension in Oracle Smart View for Office or Oracle Smart View for Office (Mac and Browser).

- Adding Attribute Dimension Members in Smart View for Office
- Adding Attribute Dimension Members in Smart View (Mac and Browser)

Adding Attribute Dimension Members in Smart View for Office

You create attribute dimensions in the web interface. You can then add members to attribute dimensions using the Admin Extension in Oracle Smart View for Office.

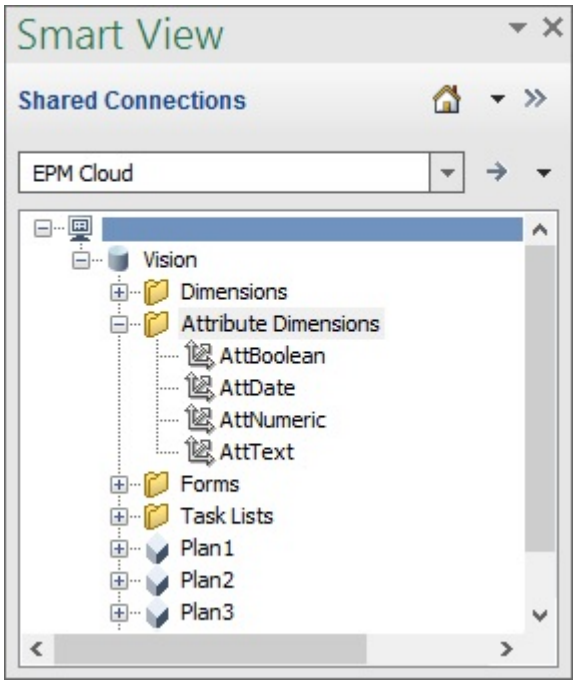
To add members to attribute dimension members:

1. From the Smart View ribbon, click Panel.
2. In the Smart View Panel, click Shared Connections or Private Connections, and provide your user name, and password in the login windows.

   If you chose Private Connections, select the private connection in the drop-down list of connections, then provide your login credentials.

3. From the Smart View Panel, select EPM Cloud.
4. In the tree in the Smart View Panel, expand the Attribute Dimensions node.

   In the following example, the Attribute Dimensions node contains four attribute dimensions, one for each attribute type: text, date, Boolean, and numeric.



5. Select an attribute dimension, and then select **Edit Dimension**.

   To illustrate, we selected the AttText attribute dimension to edit. An example of the initial grid for the attribute dimension is shown. This dimension has no members yet, and just two properties, Parent member and Default Alias Table.

	A	B	C
1		Parent Member	Default Alias Table
2	AttText		

6. Type the attribute member names to add; for example:

	A	B	C
1		Parent Member	Default Alias Table
2	AttText		
3	Gold		
4	Silver		
5	Platinum		

7. From the Smart View ribbon, click **Submit**.

   The attribute members are submitted to the application.

	A	B	C
1		Parent Member	Default Alias Table
2	AttText		
3	Gold	AttText	
4	Silver	AttText	
5	Platinum	AttText	

8. **Optional:** Continue editing the attribute members, entering data for other properties by typing directly in the grid, and then submit your changes.

Adding Attribute Dimension Members in Smart View (Mac and Browser)

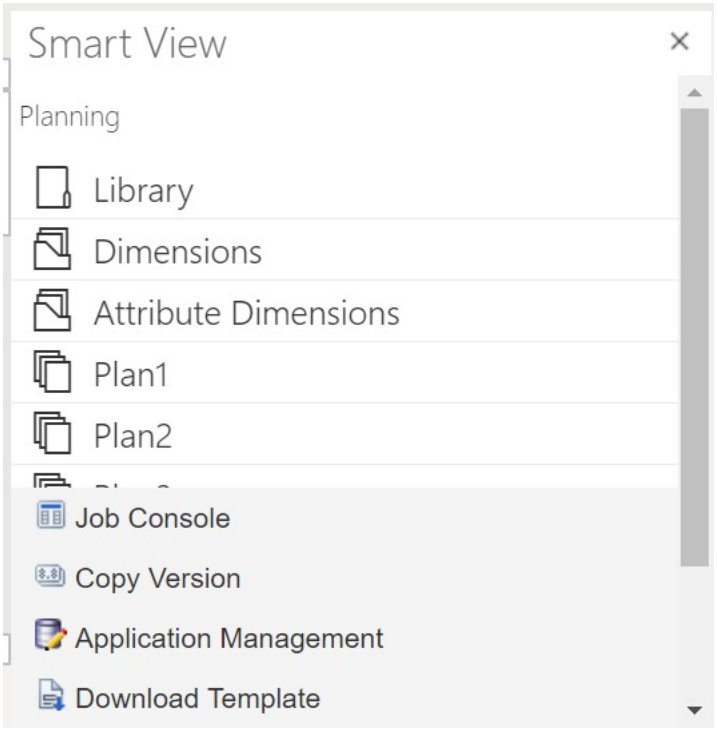
Before you begin adding attribute dimension members in Oracle Smart View for Office (Mac and Browser), you should either sideload or deploy Smart View in your environment. Instructions are in *Deploying and Administering Oracle Smart View for Office (Mac and Browser)*:

- Sideloading and Logging In
- Deploying the Manifest File to Office 365 Users

To add a member to an attribute dimension in a Oracle Smart View for Office (Mac and Browser) grid:

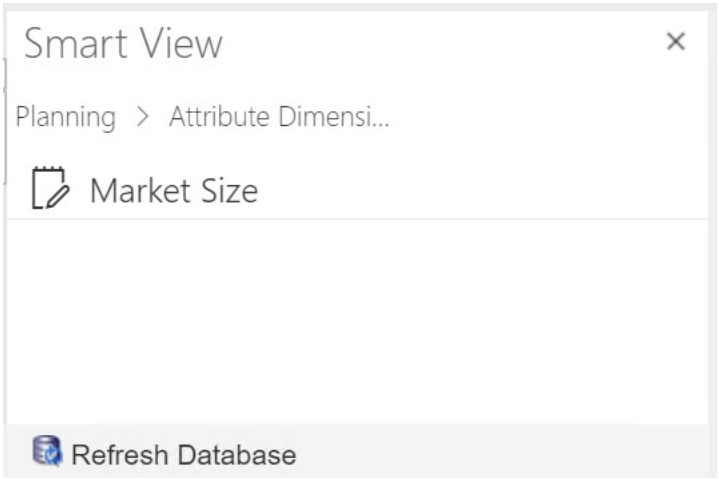
1. From the Smart View ribbon, click **Home**.

   In the **Home** panel, dimensions are displayed in the **Dimensions** folder. If attribute dimensions are defined, they are displayed in the **Attribute Dimensions** folder.



2. In the **Smart View Home** panel, click the **Attribute Dimensions** folder to expand it and view the attribute dimensions in the folder.

   In the following example, the Attribute Dimensions folder is expanded and shows one attribute dimension, Market Size, in the Vision application.



3. Select an attribute dimension to import it onto the grid.

   To illustrate, we selected the Market Size attribute dimension to edit. An example of the initial grid for the attribute dimension is shown. This dimension has no members yet, and just two properties, Parent member and Default Alias Table.

	A	B	C
1		Parent Member	Default Alias Table
2	Market Size		

4. In the attribute name column, under the attribute name, type the attribute member names to add.

	A	B	C
1		Parent Member	Default Alias Table
2	Market Size		
3	Large		
4	Medium		
5	Small		

5. From the Smart View ribbon, click **Submit**.

   The attribute members are submitted to the application.

	A	B	C
1		Parent Member	Default Alias Table
2	Market Size		
3	Large	Market Size	
4	Medium	Market Size	
5	Small	Market Size	

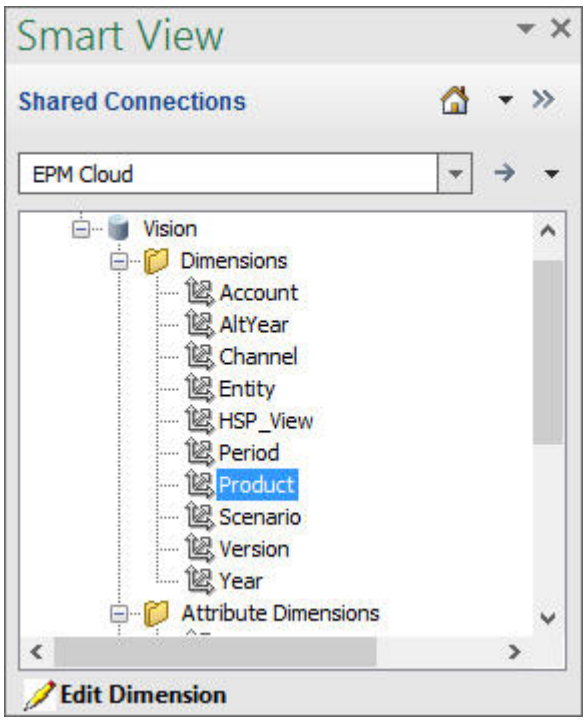
6. **Optional:** Continue editing the attribute members, entering data for other properties by typing directly in the grid, and then submit your changes.

Associating Attribute Dimension Members with Dimension Members

The illustrations in this topic are from Oracle Smart View for Office, but the concepts also apply to Oracle Smart View for Office (Mac and Browser).

1. In the tree in the Smart View Panel (Windows) or Smart View Home (Mac and Browser), expand the **Dimensions** node, and select the dimension to work with.

   In the following example, the Dimensions node shows the standard dimensions in the Vision application along with some custom dimensions such as AltYear and Channel. The Product dimension is selected.



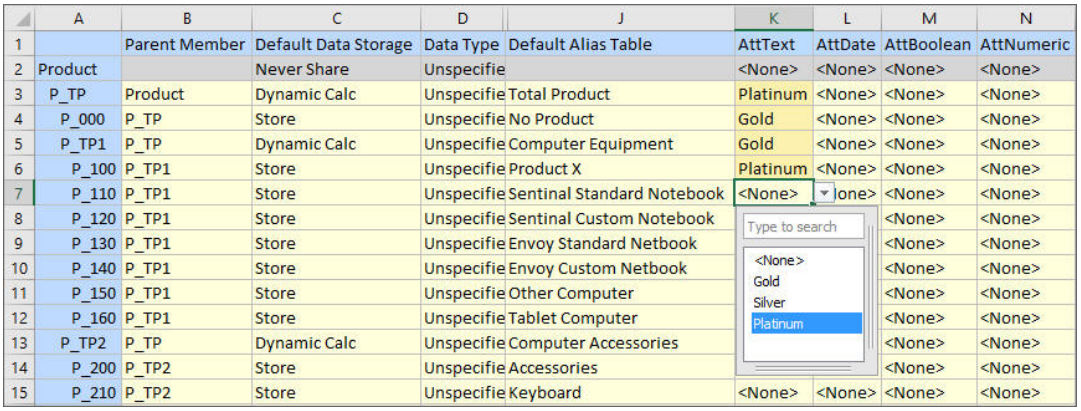
2. Select **Edit Dimension** and note the initial layout of the grid on the sheet.

   In Oracle Smart View for Office (Mac and Browser), just click the dimension to import it onto the grid.

   If the attribute dimensions are not displayed on the sheet, then use the Member Selector to add them. In the Planning Ad Hoc ribbon (Windows) or Smart View ribbon (Mac and Browser), select **Member Selection**, and then add the attribute dimensions.

3. In the grid, expand the dimension to view its members.
4. Associate the attribute member values to the dimension members using the cell-based member selector, as shown in Figure 1.

Figure D-3 Associating Attribute Values to Dimension Members



	A	B	C	D	J	K	L	M	N
1		Parent Member	Default Data Storage	Data Type	Default Alias Table	AttText	AttDate	AttBoolean	AttNumeric
2	Product		Never Share	Unspecifie		<None>	<None>	<None>	<None>
3	P_TP	Product	Dynamic Calc	Unspecifie	Total Product	Platinum	<None>	<None>	<None>
4	P_000	P_TP	Store	Unspecifie	No Product	Gold	<None>	<None>	<None>
5	P_TP1	P_TP	Dynamic Calc	Unspecifie	Computer Equipment	Gold	<None>	<None>	<None>
6	P_100	P_TP1	Store	Unspecifie	Product X	Platinum	<None>	<None>	<None>
7	P_110	P_TP1	Store	Unspecifie	Sentinal Standard Notebook	<None>	one>	<None>	<None>
8	P_120	P_TP1	Store	Unspecifie	Sentinal Custom Notebook			<None>	<None>
9	P_130	P_TP1	Store	Unspecifie	Envoy Standard Netbook			<None>	<None>
10	P_140	P_TP1	Store	Unspecifie	Envoy Custom Netbook			<None>	<None>
11	P_150	P_TP1	Store	Unspecifie	Other Computer			<None>	<None>
12	P_160	P_TP1	Store	Unspecifie	Tablet Computer			<None>	<None>
13	P_TP2	P_TP	Dynamic Calc	Unspecifie	Computer Accessories			<None>	<None>
14	P_200	P_TP2	Store	Unspecifie	Accessories			<None>	<None>
15	P_210	P_TP2	Store	Unspecifie	Keyboard	<None>	<None>	<None>	<None>

5. Repeat the previous step for any other dimension member and attribute value associations you want to define.
6. At any time, click **Submit Data** in the Planning Ad Hoc ribbon (Windows) or **Submit** in the Smart View ribbon (Mac and Browser) to submit your changes to the database.

   You can perform interim submits, or submit all changes at once.

Designating Shared Members in Smart View

The Smart View grid allows you to rapidly designate members for sharing within a dimension.

> **Note:**
>
> In Smart View, the Shared data storage option isn't available for members of the Currency dimension or any attribute dimensions.

To designate a shared member in Oracle Smart View for Office:

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** (Windows) or **Submit** (Mac and Browser) to save the grid.

The submit operation will refresh the base member with its original Parent Member and Data Storage properties. The shared member will be added under the specified parent on the server. You must zoom in on the parent to see the newly added shared member.

> **Note:**
>
> The new shared member will *not* be automatically added to the Smart View grid. The list of members displayed in the grid remains unchanged.

Refreshing Databases

Related Topics

- Refreshing Databases in Smart View for Office
- Refreshing Databases in Smart View (Mac and Browser)

Refreshing Databases in Smart View for Office

The Oracle Smart View for Office grid allows you to quickly refresh a database.

To refresh a database in Smart View:

1. From the Smart View ribbon, click **Panel**.
2. From the Smart View Panel, select **Planning**.
3. Expand the **Dimensions** folder to view the application dimensions in the folder.

   If attribute dimensions are defined, they are displayed in the **Attributes** folder.
4. Right-click the *root* **Dimension** or **Attributes** folder and select **Refresh Database**.

   The Smart View **Refresh Database** dialog box is displayed.

   > **Tip:**
   >
   > The database refresh cannot be completed while a rule type job is running or in progress. Check the Job Console, either in the web or in Oracle Smart View for Office (Mac and Browser), for any jobs running against the database. When all jobs are complete, then perform the database refresh.
5. Click **Refresh**.

   A progress bar is displayed, indicating the percentage of steps completed for the refresh or create operation.

Refreshing Databases in Smart View (Mac and Browser)

The Smart View grid allows you to quickly refresh a database or create a new one.

To refresh a database in Oracle Smart View for Office (Mac and Browser):

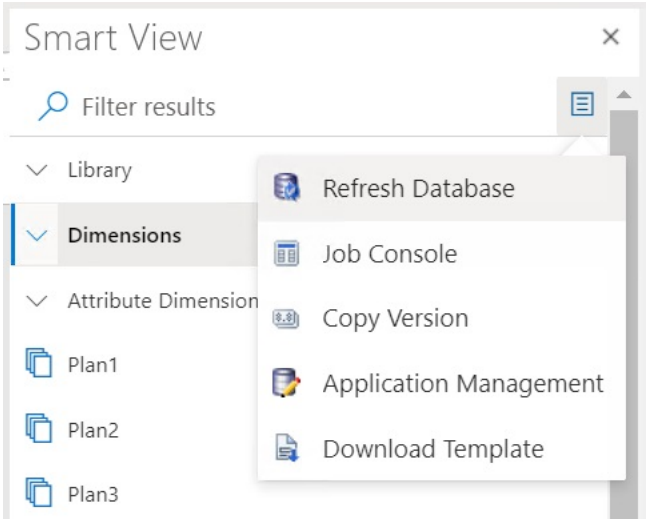
1. From the Smart View ribbon, click **Home**.
2. From the Smart View Home panel, navigate to the root application **Dimensions** folder.

   If attributes are defined in the application, to refresh attribute dimensions, navigate to the **Attribute Dimensions** folder.

3. Select the *root* **Dimension** or **Attribute Dimensions** folder, click the **Actions** button, , and then select **Refresh Database** from the drop-down menu.

   The **Refresh Database** dialog box is displayed.

   In the following example, the **Dimensions** folder is selected, and in the **Actions** drop-down menu, **Refresh Database** is selected.

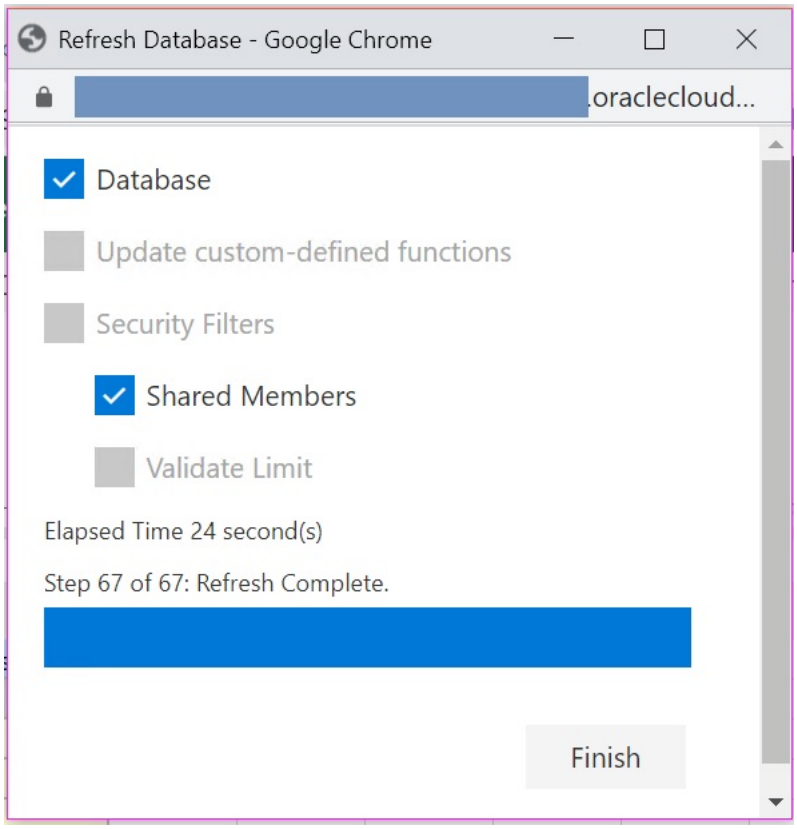


> **Tip:**
>
> The database refresh cannot be completed while a rule type job is running or in progress. Check the Job Console, either in the web or in Oracle Smart View for Office (Mac and Browser), for any jobs running against the database. When all jobs are complete, then perform the database refresh.

4. Select the **Database** check box, and then click **Refresh**.

   In the following illustration, the **Database** and **Shared Members** check boxes are selected. Other options that can be selected for refresh are:

   - Update custom-defined functions
   - Security filters
   - Validate limits



A progress bar is displayed, indicating the percentage of steps completed for the refresh or create operation.

5. Click **Finish** to close the dialog.

Planning Admin Extension and Office AutoCorrect

> **Note:**
>
> The information in this topic applies to the Planning Admin Extension on the Windows-based version of Oracle Smart View for Office; this information does not apply to the Planning Admin Extension in Oracle Smart View for Office (Mac and Browser).

In the Planning Admin Extension on Windows-based versions of Excel, Office AutoCorrect can affect member editing in Smart View.

When editing member properties, such as member names, member descriptions, aliases, or member formulas of the data type Text, you may have names that begin with two initial capital letters; for example

```
DGreen
```

Excel will autocorrect this name to one initial capital; for example:

```
Dgreen
```

In Excel, you can set an AutoCorrect option to retain two initial capital letters in a name.

To retain two initial capital letters in a name:

1. In Excel, access the **Excel Options** dialog.
2. Select **Proofing**, and then click the **AutoCorrect Options** button.
3. In the **AutoCorrect** dialog, clear the **Correct TWo INitial CApitals** check box.
4. Click **OK** to close the **AutoCorrect** dialog, and then click **OK** to close the **Excel Options** dialog.

The change takes effect immediately. There is no need to restart Excel.

E

FreeForm Best Practices

Use these best practices for FreeForm.

This table provides links to the best practices mentioned in this guide.

Category	Best Practice For	See this Section
EPM Center of Excellence	Creating and running an EPM Center of Excellence	Creating and Running an EPM Center of Excellence
Application Launch	Resolving login issues	Resolving Login Issues in *Oracle Enterprise Performance Management Cloud Operations Guide*
Down Environments	Resolving issues with environments that are down	Dealing with Down Environments in *Oracle Enterprise Performance Management Cloud Operations Guide*
Dynamic Tabs	Using dynamic tabs	Using Dynamic Tabs in Redwood Experience in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*
FreeForm App Creation	Creating FreeForm apps	Important Considerations for FreeForm Apps
Cube Optimization	Improving cube performance	• Improving Cube Performance • Optimize BSO Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide* • Optimizing Aggregate Storage Option Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide*
Data Export	Resolving issues if you get an Oracle Essbase query limit error when exporting a large number of data cells from ASO cubes	Handling Issues Related to Large Data Export from ASO Cubes in *Oracle Enterprise Performance Management Cloud Operations Guide*
Application Database Refresh	Resolving database refresh issues	Troubleshooting Database Refresh Issues in *Oracle Enterprise Performance Management Cloud Operations Guide*
Daily Maintenance	Deciding whether to perform a complete export during the daily maintenance process or to create an application backup	Exporting Smart List Textual Data During Daily Maintenance for Incremental Data Import in *Oracle Enterprise Performance Management Cloud Operations Guide*
Connecting Environments	Connecting EPM Cloud environments and migrating EPM Cloud connections	• Considerations • Considerations for Migrating EPM Cloud Connections

Category	Best Practice For	See this Section
Navigation Flows	Designing navigation flows and handling issues with navigation flows in connected environments	• Navigation Flow Design Best Practices and Naming Considerations • Handling Issues with Navigation Flows in *Oracle Enterprise Performance Management Cloud Operations Guide*
Dashboards Containing Master Forms	Designing dashboards with master forms and target objects	Creating Dashboards Containing Master Forms and Details
Dashboards 2.0 POV Bar	Using POV bars in Dashboards 2.0	Dashboard 2.0 POV Bar Considerations
Dynamic Calc	Adding children under a dynamic calc parent	About Dynamic Calc
Aliases	Assigning aliases to dimension members	About Aliases
Dynamic Members	Removing dynamic members	Considerations
Period Dimensions	Creating an alternate hierarchy in the Period dimension	Considerations for Alternate Hierarchies in Period Dimensions
Form Design	Designing forms and resolving form functional and performance issues	• Form Design Considerations • Resolving Form Functional and Performance Issues in *Oracle Enterprise Performance Management Cloud Operations Guide*
Smart Push	Resolving issues with Smart Push	Resolving Issues with Smart Push in *Oracle Enterprise Performance Management Cloud Operations Guide*
Rules Design	Designing rules to avoid common execution errors and optimizing slow rules	Troubleshooting Business Rule Errors and Performance in *Oracle Enterprise Performance Management Cloud Operations Guide*
Groovy Rules	Implementing and using Groovy rules and using Groovy rules to calculate data on forms	• Groovy Business Rule Tutorial Videos • Learning Groovy in Oracle EPM Cloud • Calculating Modified Data Using Groovy Rules
Groovy Rules and Smart Push	Using Groovy rules and Smart Push to move modified data into a reporting cube	Moving Modified Data Using Groovy Rules and Smart Push