

ORACLE® HYPERION PLANNING, FUSION EDITION

RELEASE 11.1.1.3

ADMINISTRATOR'S GUIDE

ORACLE
ENTERPRISE PERFORMANCE
MANAGEMENT SYSTEM

Planning Administrator's Guide, 11.1.1.3

Copyright © 2001, 2009, Oracle and/or its affiliates. All rights reserved.

Authors: EPM Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable: U.S. GOVERNMENT RIGHTS: Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

Contents

Documentation Accessibility	13
Chapter 1. Using Planning	15
About Planning	15
Smart View	16
Planning Web Client	16
Performance Management Architect	16
EPM Workspace	16
EPM System Lifecycle Management	17
Essbase	17
Financial Reporting	18
Business Rules	18
Web Analysis	18
User Licensing for Third-Party Software	18
Planning Usage Scenario	18
Deployment	19
Setting Up Applications	19
Building Plans	24
Launching Business Rules	24
Starting the Budget Review Process	24
Post-Planning Activities	26
Chapter 2. Getting Started with Planning	27
Overview	27
Starting Essbase	28
Considerations for Working with Essbase	28
Connecting to Multiple Instances of Essbase	29
Starting the Relational Database	30
Optimizing SQL Relational Databases	30
Setting Planning Properties	30
Changing the JDBC Driver	31
Configuring JDBC Connection Pooling	32

Setting Offline Compression	32
Setting the Maximum Length for Cell Text	33
Controlling Smart View Messages	34
Setting Background Processing	34
Logging on to EPM Workspace and Accessing Planning	36
Suppressing Password Prompts in Planning Utilities	37
Saving Text Files for Planning Utilities	37
Using Application Servers	38
About Updating Instances and Clusters	38
Closing Planning and Essbase	38
Chapter 3. Setting Up Access Permissions	39
Levels of Access Permissions	39
Planning Elements That Can Be Assigned Access	40
Types of Access Permissions	40
Enabling Access Permissions for Dimensions	42
Assigning Access to Members and Business Rules	42
Adding Access	43
Editing Access	43
Deleting Access	44
Assigning Access Permissions to Shared Members	45
Managing Access to Data Forms and Folders	46
Assigning Access to Data Forms and Folders	46
Changing Access to Data Forms and Folders	47
Importing Access Permissions	48
Improving Performance When Importing Access Permissions	51
Exporting Access Permissions	51
Reporting on Access Permissions	54
Selecting Reporting Objects	55
Selecting Reporting Options	55
Working With Access Permissions Reports	56
Setting up Audit Trails	56
Managing Security Filters	58
Synchronizing Users and Groups With the UpdateUsers Utility	59
Synchronizing Users With the Provision Users Utility	60
Migrating User and Group Identities	61
Migrating Business Rule Security	62
Removing Stale User Records	63
Setting Up Access Permissions in Financial Reporting	63

Chapter 4. Managing Planning Databases	65
Opening Applications	65
Unlocking Applications	65
Using Broadcast Messaging	66
Viewing Usage Statistics	67
Creating and Refreshing Application Databases	67
Managing Exchange Rates	69
Creating Exchange Rate Tables	70
Editing Exchange Rate Tables	70
Deleting Exchange Rate Tables	71
Managing Currency Conversion	72
Working with Currency Conversion Calc Scripts	72
Optimizing Application Performance	73
Assigning Dense and Sparse Dimensions	73
Setting the Evaluation Order	73
About Reordering Dimensions	74
Allocating Memory for Supporting Detail Cache	74
Clearing Options when Creating or Refreshing an Application Database	74
Backing Up Applications and Application Databases	74
Chapter 5. Loading Data and Metadata	77
Overview of Loading Information	77
Working with the Outline Load Utility	78
Loading Data	79
Loading Metadata	82
Command Line Parameters for the Outline Load Utility	83
Dimension Properties	89
Load File Considerations	103
Loading Information with Data Integrator Adapter for Planning	104
Loading Information with DIM Adapter for Planning	105
Enabling Data Load of Essbase Data	105
Loading Information with Administration Services	105
File Example	106
Loading Information with Performance Management Architect	106
Loading Information with FDM and ERP Integrator	106
Chapter 6. Using Data Forms	107
About Data Forms	107
Data Forms and Plan Types	107
Data Forms and Access Permissions	108

Data Forms and Currencies	108
Designing Data Forms for Multiple Currencies	108
Designing Data Forms for Drill-Through Information	108
Data Forms and Versions	109
Data Forms and Attributes	109
Data Forms and Shared Members	109
Data Forms and Calculations	109
Understanding Data Forms	109
Point of View	110
Page Axis	110
Rows and Columns	110
Creating Data Forms	110
Selecting Data Forms and Folders	111
Setting Data Form Properties	111
Setting the Row and Column Layout	112
Setting Rows and Columns for Composite Data Forms	114
Setting Column and Row Properties	115
Adding Column Definitions	116
Adding Row Definitions	116
Changing Column Definitions	117
Changing Row Definitions	117
Selecting Members	117
Selecting Attribute Values as Members	119
Selecting Substitution Variables as Members	122
Selecting User Variables as Members	123
Creating Asymmetric Rows and Columns	124
Enabling Users to Add Rows	124
Defining the Page and POV	124
Setting Other Options for Data Forms	125
Using Business Rules	128
Selecting Business Rules	128
Setting Properties for Business Rules	129
About Runtime Prompts	130
Understanding Runtime Prompts	131
Launching Business Rules With a Utility	134
Previewing Data Forms	136
Managing User Variables	136
Selecting User Variables	137
Deleting User Variables	137

Printing Data Form Definitions	137
Searching for Data Forms	138
Modifying Data Forms	138
Moving Data Forms	139
Deleting Data Forms	139
Overview of Importing Data Form Definitions	139
Preparing Data Forms	140
Preparing Data Files	140
Importing Data Form Definitions	141
ImportFormDefinition Examples	142
Importing and Exporting Data Forms	143
Planning Offline Considerations	144
Offline Data Form Considerations	144
Offline User Considerations	144
Business Rule Considerations for Offline Calculations	145
Chapter 7. Managing the Budgeting Process	147
About the Budgeting Process	147
Planning Units	147
Starting the Review Process	147
Entity Hierarchy	148
Setting Up E-mail for Workflow Notification	148
Printing Planning Unit Annotations	149
Overview of Managing Task Lists	149
Managing Task Lists	149
Creating Task List Folders	150
Renaming Task Lists	150
Renaming Task List Folders	150
Moving Task List Folders	150
Deleting Task List Folders	151
Creating Task Lists	151
Adding Instructions to Task Lists	151
Adding and Defining Tasks	152
Editing Task Lists	154
Linking Tasks to Planning Web Pages	156
Assigning Access to Task Lists	157
Importing and Exporting Task Lists	159
Copying Data	160
Clearing Cell Details	161

Chapter 8. Working With Applications	163
Managing Data Forms and Folders	163
Creating Folders	163
Moving Folders	164
Deleting Folders	164
Renaming Folders	165
About Setting Preferences	165
Setting Personal Preferences	166
Setting Application Defaults	166
Specifying System Settings	166
Assigning Applications to Shared Services Projects	168
Limiting Use of Applications During Maintenance	168
Specifying Custom Tools	169
Setting Display Options	169
Setting Printing Options	169
Deleting Application Information Using SQL	169
Deleting Account Annotations	170
Deleting Supporting Detail Associated With a Scenario	170
 Chapter 9. Working with Menus	 173
Creating and Updating Menus	173
Working with Menu Items	173
Adding or Changing Menu Items	174
 Chapter 10. Working with Classic Application Administration	 177
About Creating Applications with Performance Management Architect and Classic Administration	177
Setting Up Classic Applications	177
Creating Applications	178
Managing Data Sources	178
Selecting Data Sources	181
Setting up the Calendar	182
Setting up Currencies	183
Specifying Plan Types	183
Reviewing Application Information	184
Registering Applications	184
Deleting Applications	185
Working with Alias Tables	185
About Alias Tables	185
Creating Alias Tables	186

Editing or Renaming Alias Tables	186
Deleting Alias Tables	186
Clearing Alias Tables	186
Copying Alias Tables	187
Setting a Default Alias Table	187
Working with Dimensions	187
Dimension Overview	187
Working with Dimension Hierarchies	188
Viewing a Member's Ancestors	192
Determining Where Members Are Used in Applications	192
About Custom Dimensions	192
About Entities	194
About Accounts	195
Accounts, Entities, and Plan Types	199
About User-Defined Custom Dimensions	199
Adding or Editing User-Defined Custom Dimensions	200
Setting Dimension Properties	200
Working with Members	201
Working with Shared Members	204
Working with Attributes	205
Working with Attribute Values	207
Customizing Calendars	209
Setting Up Currencies	213
Enabling Multiple Currencies	213
Specifying Exchange Rates	218
Setting Up Scenarios	220
Specifying Versions	223
Sorting Version and Scenario Members	225
Moving Scenario and Version Members in the Dimension Hierarchy	226
Setting up Dynamic Time Series Members	226
Additional Supported Essbase Features	227
Considerations for Alternate Hierarchies in Period Dimensions	228
Working with Classic Applications	228
Refreshing Application Databases Using a Utility	228
Refreshing Databases with Level 0 Members Set to Dynamic Calc	230
Working With Essbase Partitions	230
Sorting Members Using a Utility	230
Deleting Shared Descendants Using a Utility	231
Using Smart Lists, UDAs, and Member Formulas	232

Working with Smart Lists	232
Working with UDAs	235
Working with Member Formulas	237
Chapter 11. Customizing Planning Web Client	241
Customizing Reports	241
Customizing the Planning Tools Page	243
Customizing Cascading Style Sheets	243
Default Location of Cascading Style Sheet Files	245
Modifying Cascading Style Sheets	245
Customization Examples	245
Customizing the Style of Row and Column Members in Data Forms	247
About Customizing Text, Color, and Images	248
Customizing Text, Color, and Images	248
Localizing Text for Planning Applications	249
Restricted Characters in Customized Messages	250
Customizing Colors, States, and Actions for Process Status	251
Customizing Colors	251
Customizing States	252
Customizing Actions	252
Creating Spreading Patterns	253
Examples of Spreading Patterns	254
Chapter 12. Troubleshooting	257
Calculation Script is Too Long or Complex	257
Cannot Find Member	258
Cannot Process Request	258
Conflicting Change by Users	259
Copy Version Error	259
Currency Conversion Calc Script Failed	260
Data Form Error Message	260
Database Locked by Another Administrator	260
Exiting Planning	260
500 Error Message	261
Invalid Value When Entering Date Value	262
Manage Database Errors	262
Maximum Number of Applications Already Connected or Database is Locked	263
Registering a Restored Application with Planning	264
Session Timeout	265
Slow Performance When Opening Data Forms Using a Dial-Up Connection	265

Slow Performance When Opening Large Data Forms	266
Unable to Create Application Error	267
Unable to Refresh Application Error	267
Unsuccessful Log On	268
Appendix A. Customizing Data Forms with JavaScript	269
About Customizing Data Forms	269
Modifying JavaScript	269
customCellEnterPre	269
customCellEnterPost	270
customCellValidatePre	271
customCellValidatePost	272
customOnLoad	273
drawCustomButtons	273
validateForm	273
Deploying Custom JavaScript	274
Appendix B. Naming Restrictions for Essbase	275
Naming Restrictions for Applications	275
Naming Restrictions for Dimensions, Members, Aliases, and Data Forms	276
Dimension and Member Names in Calc Scripts, Report Scripts, Formulas, Filters, and Substitution Variables	277
Naming Restrictions for User Names	278
Appendix C. Sample Application	279
About the Application	279
Predefined Structure	280
Dimension Members	280
Data Forms	280
Business Rules	281
User Variables	281
Substitution Variables	282
Menus	283
Task Lists	283
Database	283
Users	283
Creating Sample Applications	283
Loading Data	284
Glossary	285

Index 309

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

1

Using Planning

In This Chapter

About Planning.....	15
Smart View.....	16
Planning Web Client	16
Performance Management Architect	16
EPM Workspace	16
EPM System Lifecycle Management	17
Essbase	17
Financial Reporting	18
Business Rules	18
Web Analysis	18
User Licensing for Third-Party Software	18
Planning Usage Scenario.....	18

About Planning

Oracle Hyperion Planning, Fusion Edition is a Web-based budgeting and planning solution that drives collaborative, event-based operational planning processes throughout the organization for a wide range of financial and operational needs. It enables a complete and closed-loop planning process that drives continuous business improvement. Decision makers and managers can communicate the course of action and collaborate with budget holders to optimize the planning process. Planners have the flexibility to adapt rapidly, ensuring plans are relevant and useful. Planning:

- Facilitates collaboration, communication, and control across multi-divisional global enterprises
- Provides a framework for perpetual planning, to manage volatility and frequent planning cycles
- Provides ease of use and deployment through the Web or Oracle Hyperion Smart View for Office, Fusion Edition
- Lowers the total cost of ownership through a shorter roll out and implementation phase, and easier maintenance for applications
- Enhances decision-making with reporting, analysis, and planning
- Promotes modeling with complex business rules and allocations

- Integrates with other systems to load data

Smart View

Planners can disconnect from the Planning server and work with data forms offline. Planners retrieve data forms from Planning, load them into Smart View, and take them offline. They can enter and save data to the Planning server. See the *Oracle Hyperion Smart View for Office User's Guide*.

Planning Web Client

Planning provides complete functionality for Web users. Use the Web interface to roll out applications to large, distributed organizations without installing software on client computers. All software resides on the server. Many administrative features that were formerly in the Planning Desktop are now available through Planning Web. Other administrative features, such as creating and administering applications and dimensions, are available through Oracle Hyperion EPM Architect, Fusion Edition.

Performance Management Architect

Performance Management Architect is a component of Planning installation and configuration. You use it to create and work with Planning applications and dimensions, Smart Lists, UDAs, member formulas, and other features.

With Performance Management Architect, you can view, create, and validate Performance Management Architect applications, and deploy them to create Planning applications. Deploying applications from Performance Management Architect to Planning is a long-running operation. The initial deployment may take more time than subsequent re-deployments. For assistance on tasks performed in Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*. For information on installing and configuring Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

Performance Management Architect applications can use business rules created with Hyperion Calculation Manager (see the Calculation Manager part of the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*).

EPM Workspace

Planning is available within Oracle Enterprise Performance Management Workspace, Fusion Edition. For information on EPM Workspace features, such as EPM Workspace preferences, see the *Oracle Enterprise Performance Management Workspace User's Online Help* or *Oracle Enterprise Performance Management Workspace Administrator's Online Help*. To log on to EPM Workspace, see [“Logging on to EPM Workspace and Accessing Planning” on page 36](#).

EPM System Lifecycle Management

Lifecycle Management provides a consistent way for Oracle Hyperion Enterprise Performance Management System products to migrate an application, a repository, or individual artifacts across product environments and operating systems. Generally, the Lifecycle Management interface in Oracle's Hyperion® Shared Services Console is consistent for all EPM System products that support Lifecycle Management. However, Oracle Hyperion Enterprise Performance Management System products display different artifact listings and export and import options in the Lifecycle Management interface. Migration to and from Planning is a long-running operation.

Lifecycle Management features:

- Viewing applications and folders
- Searching for artifacts
- Comparing applications and folders
- Migrating directly from one application to another
- Migrating to and from the file system
- Saving and loading migration definition files
- Viewing selected artifacts
- Auditing migrations
- Viewing the status of migrations
- Importing and exporting individual artifacts for quick changes on the file system

In addition to providing the Lifecycle Management interface in Shared Services Console, there is a command-line utility called Lifecycle Management Utility that provides an alternate way to migrate artifacts from source to destination. The Lifecycle Management Utility can be used with a third-party scheduling service such as Windows Task Scheduler or Oracle Enterprise Manager.

Lastly, there is a Lifecycle Management Application Programming Interface (API) that enables users to customize and extend the Lifecycle Management functionality.

For detailed information about Lifecycle Management, see the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*.

Essbase

Planning leverages Oracle Essbase analytic and calculation capabilities, security filters, APIs, pre-built financial intelligence, calculation functions, and multi-database application support. Planning stores the application definition in a relational database, and creates Essbase databases and security privileges for applications.

Data sources are used to link the relational database and the Essbase server, and are associated with each Planning application. For information on creating data sources for Classic Planning applications, see [“Managing Data Sources” on page 178](#). For information on data sources for

Planning applications created in Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Financial Reporting

Oracle Hyperion Financial Reporting, Fusion Edition is a management reporting tool that transforms data into meaningful business information through highly formatted reports. Planning users can use Financial Reporting to manage reporting tasks and implement plan review and analysis. Users can create reports featuring text, grids of data, charts, graphs, and images. They can use real-time, ad hoc variance reporting, and produce a variety of sophisticated financial reports that can be viewed online or printed with production-quality formatting.

Business Rules

Business rules perform complicated calculations through predefined formulas, variables, calculation scripts, macros and sequences (for Oracle's Hyperion® Business Rules) and rulesets and templates (for Calculation Manager). See [“Using Business Rules” on page 128](#).

Web Analysis

Oracle's Hyperion® Web Analysis is an analysis, presentation, and reporting solution. It allows organizations to deliver information to large user communities at a low cost.

User Licensing for Third-Party Software

To use Planning, you must purchase licenses from third-party vendors, for example, for a relational database and Web application server.

Planning Usage Scenario

You create applications using Performance Management Architect application administration, and deploy them to Planning. Applications contain dimensions and dimension attributes designed to meet planning needs, such as accounts, entities, scenarios, and other dimension elements. You can create an unlimited number of applications. For information about Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

These topics outline how Planning is used:

- [“Deployment” on page 19](#)
- [“Setting Up Applications” on page 19](#)
- [“Building Plans” on page 24](#)

- “Launching Business Rules” on page 24
- “Starting the Budget Review Process” on page 24
- “Post-Planning Activities” on page 26

Deployment

An IT professional installs and configures Planning on server computers and client workstations. The IT professional should have experience administering relational databases, installing client/server and Web server software, and configuring user directories.

Table 1 Server Installation

User	Task
IT professional	Installs Planning system requirements, including operating systems, relational database, Web server, application server, and Web browser
IT professional	Installs Essbase
IT professional	Sets up the data source name (DSN) for the OLAP and relational databases
IT professional	Installs Planning and Financial Reporting.
IT professional	Creates relational databases and connections
IT professional	Configures the Web server
IT professional	Sets up users in an external user directory
IT professional	Optional: Creates a test application with users
IT professional	Tests installation and configuration

Table 2 Client Installation Users and Tasks

User	Task
IT professional	Installs the Planning remote Windows client, Smart View, Microsoft Excel, and Financial Reporting
IT professional	Installs a Web browser
IT professional	Tests connections to the Web and network

Setting Up Applications

Setting up applications includes the tasks listed in this section. Administrators can also set up applications using Performance Management Architect application administration, and deploy them to Planning. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide*.

Creating Metadata

Consulting partners and the Planning administrator design applications and create an application framework with Performance Management Architect application administration. There is only one application owner per application. The owner can grant ownership to another administrator. The application framework includes:

- Application name
- Number and names of plan types
- Calendar structure
- Whether the application is a multicurrency application
- Default currency

After the application framework is created, dimensions and members are added to the application and assigned to the plan type in Performance Management Architect. There are up to eight required dimensions in an application:

- Currency (for multicurrency applications)
- HSP_Rates (for multicurrency applications)
- Account
- Entity
- Scenario
- Version
- Period
- Year

Table 3 Creating Metadata

User	Task
Administrator	Designs and creates applications in Performance Management Architect.
Administrator	Defines metadata (currencies, scenarios, versions, calendar) in Performance Management Architect
Administrator	Enters exchange rates
Administrator	Defines attributes and attribute values in Performance Management Architect
Administrator	Loads custom dimensions using Performance Management Architect

Defining Users and Access Permissions

An IT professional defines users and groups and configures a user directory before users can access Planning applications. For detailed information, see the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*. Within Planning, you assign users or groups to secured objects.

Table 4 Setting Up Users and Access Permissions

User	Task
IT professional	Sets up Planning users with a supported user directory
Administrator	Synchronizes users in the user directory with a Planning application
Administrator	Assigns access permissions to users and groups
Administrator	Assigns access permissions to dimensions in Planning

Creating and Refreshing Applications

Administrators create and periodically refresh the Essbase outline and security structure for the Planning application. The outline and security structure are created based on metadata stored in the Planning application's relational database. Planning creates:

- An Essbase application
- Multiple Essbase databases (one per plan type)
- Essbase access permissions filters
- Essbase outlines (all metadata):
 - Members
 - Shared members
 - User-defined attributes and attribute values
 - Exchange rates

Table 5 Generating Databases

User	Task
Administrator	Creates and refreshes Planning applications based on metadata stored in the application's relational database

Designing Data Forms

Table 6 Designing Data Forms

User	Task
Administrator and interactive user	Creates and maintains data forms in Planning
Administrator and interactive user	Creates and manages folders in Planning for data form management
Administrator and interactive user	Assigns access permissions to data forms and folders

Designing Worksheets

Planning users can install Smart View to work with data forms in Excel to leverage Excel worksheet models, build custom formulas, format reports, and work disconnected from Planning.

Populating Applications with Data

Administrators and interactive users use Performance Management Architect application administration to pre-populate Planning applications with data. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Table 7 Populating Applications with Data

User	Task
Administrator	Use Copy Data to copy plans from one dimensional intersection to another, including relational data and supporting detail
Administrator and interactive user	Performs bulk loads
Any user	Enters values for global saved assumptions referenced in business rules
Any user	Enters data into the Planning application through data forms or Smart View
Any user	Calculates and creates data using business rules or Copy Version

Creating Business Rules

Table 8 Creating Business Rules

User	Task
Administrator	Assigns users access permissions to business rules
Administrator and interactive user	Creates business rules. See the <i>Oracle Hyperion Enterprise Performance Management System Security Administration Guide</i> for information on Calculation Manager roles and rights.
Administrator, interactive user, and planners who are assigned launch access permissions by an administrator	Launches business rules for Planning applications

For information on administering business rules, see:

- For business rules created with Business Rules: *Hyperion Business Rules Administrator's Guide*.
- For business rules created with Calculation Manager: Calculation Manager part of the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

See also: [“Using Business Rules” on page 128](#).

Setting Targets

Administrators set target type versions for Planning applications. User access to target data is determined by access permissions. Typically, target data is stored in the upper levels of metadata, such as Business Unit. Administrators configure the Essbase database so target data is not replaced by lower-level #MISSING values.

Target data is typically used as guidance for data entry and analysis. Create it, for example, though:

- Manual data entry
- Modeling with business rules
- Pushing targets down to lower-level members using business rules

Table 9 Setting Targets

User Type	Task
Administrator	Adds a standard target-type version
Administrator	Assigns users access permissions (typically, read-only) to the target version
Administrator	Configures the Essbase database so target data is not replaced by lower-level #MISSING values
Any user	Creates target data
Administrator	Publishes targets using Financial Reporting
Any user	Displays targets on data forms for guidance or input

Reporting

Use Financial Reporting to create reports for Windows or Web-enabled clients. Financial Reporting uses Essbase databases and adheres to Essbase access permissions filters generated by Planning.

Table 10 Reporting

User Type	Task
Administrator and interactive user	Creates and launches reports using Financial Reporting
Any user	Prints reports
Any user	Views reports throughout the planning cycle

Initializing Planning Cycles

Table 11 Initializing Planning Cycles

User Type	Task
Administrator	Selects planning units for iterative review, analysis, and approval
Any user	Specifies whether to receive e-mail after becoming planning unit owners

Building Plans

Administrators define task lists to guide users through the planning process. Planning users start the planning cycle by logging into an application and opening data forms. Users read data form instructions and review historical or target data for guidance when preparing plans. For additional user tasks, see the *Oracle Hyperion Planning User's Online Help*.

Users can save data in data forms, and can refresh currently stored data values in data forms before saving. Users can restore data to the last saved version, and save a personal version using Copy Version. When users save:

- Data is saved to the Essbase database.
- Subtotals on data forms are calculated and stored if the Calculate Data Form business rule is selected to run on save when data forms are created. (By default, this business rule is not selected to run on save.)
- Business rules are launched to calculate data.

Launching Business Rules

The business rules available to users depend on access permissions. Administrators can assign launch access permissions to Calculation Manager business rules from within Planning (see [Chapter 3, “Setting Up Access Permissions”](#)). For more information on business rule access permissions, also see:

- For Calculation Manager: Calculation Manager part of the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*
- For Business Rules: *Hyperion Business Rules Administrator's Guide*
- *Hyperion EPM System Security Administration Guide*

Users with launch permissions can launch business rules from Planning or Smart View.

Starting the Budget Review Process

The administrator initializes review processes by placing planning units in the First Pass state. Data is typically promoted for review after users enter data, calculate, and save. When users promote planning units, they enter annotations that record assumptions and select the next owners or reviewers. Promoting planning units:

- Assigns ownership to the selected user
- Overrides access permissions on the planning unit to read-only for non-owners
- Changes planning unit status from First Pass to Under Review
- Updates status for upper-level (or lower-level) planning units to Under Review as necessary
- Records an audit entry in the process history
- Optionally sends e-mail notifications to the application owner and new planning unit owner

Reviewers can make modifications and annotations. Read-only users can enter planning unit annotations and account annotations, but cannot enter data values. Users can keep a personal copy of the original data to track modified data. When planning units are promoted, the reviewer is the new owner and typically:

- Receives e-mail notifications and is directed to the Process Definition page
- Sorts the status page by entity, process state, or current owner
- Reviews annotations and planning unit history
- Opens data forms and reviews, analyzes, modifies, and calculates data
- Makes annotations (including account annotations)
- Promotes or rejects the planning unit, with iterations until the planning unit is approved

After planning units are approved, the application owner becomes the planning unit owner. Only administrators can reject planning units. To close the review cycle, the application owner or administrator changes the scenario or version to read-only for all users.

Table 12 Starting the Budget Review Process

Users	Tasks
Administrators	Starts the planning unit for the planning cycle
Users with access	Enters data
Users with access	Calculates data
Users with access	Optionally, creates a copy of submissions
Users with access	Promotes data for review
Users with access	Enters annotations to support the review process. Until an owner is established for a planning unit, multiple users can update it. Access permissions for non-owners are changed to read-only when the planning unit is promoted to the next reviewer.
Users with access	Notifies reviewers that the planning unit is ready for review
Users with access	Reviews data in the promoted planning unit
Users with access	Reads and modifies promoted data through data forms
Users with access	Runs business rules

Users	Tasks
Users with access	Reads promoted data using Financial Reporting, Oracle's Hyperion® Web Analysis, Smart View, or third-party reporting tools
Users with access	Modifies data values, enters annotations, signs off on planning units, and promotes, rejects, and approves planning units
Administrators	Checks the planning process status
Administrators	Reads supporting annotations
Administrators	Views the audit trail
Administrators	Completes the planning cycle
Administrators	Changes user access of scenarios and versions in Essbase to read-only
Administrators and interactive users	Publishes reports using Financial Reporting

Post-Planning Activities

Table 13 Post-Planning Activities

Users	Tasks
Administrators and interactive users	Uploads planning data by scenario to other applications, a general ledger, or an ERP
Users with access	Copies a version of the completed plan to a new version or copies data from one dimensional intersection to another
Administrator and interactive user	Performs bulk loads of the latest actuals data to report on actual performance versus plan
Users with access	Accesses data forms, Financial Reporting, Smart View, Oracle Hyperion Performance Scorecard, Fusion Edition, or third-party reporting tools to read updates to actuals and analyze performance to plan

2

Getting Started with Planning

In This Chapter

Overview	27
Starting Essbase	28
Considerations for Working with Essbase.....	28
Connecting to Multiple Instances of Essbase	29
Starting the Relational Database.....	30
Optimizing SQL Relational Databases	30
Setting Planning Properties	30
Logging on to EPM Workspace and Accessing Planning	36
Suppressing Password Prompts in Planning Utilities	37
Saving Text Files for Planning Utilities.....	37
Using Application Servers	38
About Updating Instances and Clusters	38
Closing Planning and Essbase.....	38

Overview

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

You can work with applications using Performance Management Architect and Classic application administration. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide* for these Performance Management Architect application administration tasks:

- Create an application in Performance Management Architect and deploy it to create a Planning application.
- Create entities to reflect your organization's budget structure.
- Create accounts and dimensions for gathering planning data.
- Create scenarios and versions.
- Modify the calendar with any custom summary time periods needed by your organization.
- For multicurrency applications, specify the base currency, currency code and symbol, triangulation currency, reporting currency, and exchange rate type.

See this guide for Classic application administration tasks, including:

- Create, delete, and register Classic applications.
- Create data forms, task lists, and menus.
- Assign access permissions for dimension members, task lists, data forms, and data form folders. See [Chapter 3, “Setting Up Access Permissions”](#) and the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide* and its appendix on Planning.
- Manage currency conversion and exchange rate tables.
- Identify the review and approval process, requirements, and participants, and manage the budgeting process.

For information on installing and configuring Planning and Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect System Installation and Configuration Guide*.

Starting Essbase

The data for Planning applications resides on the Essbase server. Essbase must be running before you can open Planning applications. After Essbase is started, it can be minimized and run in the background or as a service.

Considerations for Working with Essbase

Creating and refreshing on the Manage Database page affects data in the Essbase database. When you click Create, data is erased and Planning plan types are rebuilt. When you click Refresh, data might be replaced. For important information, see [“Creating and Refreshing Application Databases” on page 67](#).

If you are not using Performance Management Architect, you work with applications using Classic application administration. If you upgrade an application created in Classic application administration to Performance Management Architect, you cannot return to working with that application in Classic application administration. Modifying Essbase outlines directly through Oracle Essbase Administration Services is not supported.

When you refresh Planning applications, the Essbase database makes these changes:

- If a member does not exist in the Planning relational database, the member and its properties, attributes, and User-Defined Attributes (UDAs) are not retained in Essbase on Refresh.
- Planning first retrieves member properties from any existing member in Essbase, then sets and overrides any member property definable in Planning.

This behavior is different from prior releases, where the refresh process was incremental and retained modifications made directly to the outline in Essbase, even where those members, attributes, and UDAs did not exist in Planning.

The HSP_UDF UDA preserves member formulas defined outside of Planning. You can assign this UDA to members from within Planning. Unassigning them through Planning refresh does not unassign the UDAs from members in the Essbase outline. If a member with the HSP_UDF UDA is added directly in Essbase, refreshing the database preserves any formula on this member in Essbase, and does not generate a formula placeholder (;) on members that are level zero and dynamic calc. Other UDAs, such as custom UDAs and the HSP_NOLINK UDA, are retained only if defined in Performance Management Architect or in Planning (for Classic applications). To create and update UDAs, see [“Working with UDAs” on page 235](#).

Because member formula support is available in Performance Management Architect and Planning (for Classic applications) there is less need for the HSP_UDF UDA. Every effort should be made to define UDAs in Performance Management Architect or in Planning (for Classic applications).

The @XREF function looks up a data value from another database to calculate a value from the current database. You can add the HSP_NOLINK UDA to members to prevent the @XREF function from being created on all plan types that are not the source plan type selected for that member. For more information on UDAs and functions, see the *Oracle Essbase Database Administrator's Online Help*.

It is not necessary to modify Essbase outlines for the most commonly used Essbase member properties. Performance Management Architect and Planning support Dynamic Time Series in the Period dimension, alternate hierarchy in the Period dimension, hierarchies in Scenario and Version dimensions, and hierarchy and aliases for attribute dimensions.

Use these guidelines for working in Essbase while logged on as a supervisor or application administrator:

- Do not change dense/sparse dimension settings or the order of dimensions in Essbase.
- You can use any attribute dimension type, including text, Boolean, numeric, and date.
- Do not use Link Reporting Objects (LRO).
- Do not change the Time Balance default settings. The defaults are Ending for Balance and None for Flow.
- Do not change the Variance Reporting settings. These are preset for all account types except Saved Assumptions. For Saved Assumptions account types, Expense and Non-Expense are valid settings.
- Back up the database before refreshing. See [“Backing Up Applications and Application Databases” on page 74](#).

Connecting to Multiple Instances of Essbase

You can connect to multiple instances of Essbase simultaneously from the Planning application by specifying the port number:

- When creating a Planning application with the Classic Application wizard or in Performance Management Architect (repeat for every data source name).
- As the value for the property ESS_SERVER in the HSPSYS_DATASOURCE database table.

For instructions on installing multiple instances of Essbase on the same computer, see the *Oracle Essbase Database Administrator's Online Help*.

Starting the Relational Database

A relational database must be running before you can open Planning applications. After the relational database is started, it can be minimized and run in the background or as a service.

Optimizing SQL Relational Databases

You can configure the timeout value through the Timeout registry key setting. Depending on the size of your database records, some SQL queries issued by Planning could take longer to execute than the default ADO timeout of 30 seconds. This could lead to failure when refreshing the application database. You can increase the timeout value (for example, to 180 seconds) to decrease the likelihood of refresh failure.

► To optimize the timeout value:

- 1 **Open the Registry Editor.**
- 2 **Navigate to this location:**

`HKEY_LOCAL_MACHINE/SOFTWARE/HyperionSolutions/Planning`

If the Timeout registry key exists in the right pane, the timeout value is set to a corresponding value. If the key does not exist, the value is set to the default. Its type must be DWORD.

Setting Planning Properties

You can add or change application or system properties.

Table 14 Examples of Configurable Properties

PROPERTY_NAME Column	Description
RDB_DRIVER RDB_SERVER_URL	See “Changing the JDBC Driver” on page 31.
JDBC_MIN_CONNECTIONS JDCB_MAX_CONNECTIONS	See “Configuring JDBC Connection Pooling” on page 32.
OFFLINE_COMPRESSION_THRESHOLD	See “Setting Offline Compression” on page 32.
SUPPORTING_DETAIL_CACHE_SIZE	See “Allocating Memory for Supporting Detail Cache” on page 74.
SUBST_VAR_CACHE_LIFETIME	See “Selecting Substitution Variables as Members” on page 122.
DIRECT_DATA_LOAD DATA_LOAD_FILE_PATH	See “Loading Data” on page 79.

PROPERTY_NAME Column	Description
MAX_CELL_NOTE_SIZE	See “Setting the Maximum Length for Cell Text” on page 33.
RULE_MAX_WAIT	See “Setting Background Processing” on page 34.
CAPTURE_RTP_ON_JOB_CONSOLE	See “About Runtime Prompts” on page 130.

► To set Planning application or system properties:

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

2 Select:

- **Application Properties:** set properties for the current application.
- **System Properties:** set properties for all Planning applications.

3 To:

- Change a property, change its value under **Property Value**.
- Add a property, click **Add**. Enter a name (avoid using spaces) and value for the property in the blank row.

If you enter a property already on the tab, its name turns red to indicate it is a duplicate. You can duplicate properties between the application and system tabs.

4 Click **Save** and confirm your changes.

5 Stop and then restart the application server.

Properties are saved in the HSPSYS_PROPERTIES system database table, which by default is located in the relational database you create when selecting the Configure Database task under Planning in Oracle's Hyperion Enterprise Performance Management System Configurator.

Changing the JDBC Driver

By default, Planning uses the embedded Oracle JDBC driver. To change the JDBC driver, update the JDBC property using these values:

Table 15 JDBC Property Values

JDBC Driver	RDB_SERVER_URL	RDB_DRIVER
Oracle Thin	jdbc:oracle:thin:@%SERVER_NAME%:1521:%DB_NAME%	oracle.jdbc.driver.OracleDriver
DB2 native	jdbc:db2:%DB_NAME%	COM.ibm.db2.jdbc.app.DB2Driver
DB2 native for remote DB2 server	jdbc:db2://myhost.mydomain.com:6789/%DB_NAME%	COM.ibm.db2.jdbc.net.DB2Driver

Database driver names and URLs are stored in the HSPSYS_DATASOURCE database table.

- After changing JDBC properties, to make the necessary third-party files accessible to the Planning server:

1 Locate the .jar file (for Oracle) or .zip file (for DB2):

- For Oracle, find `classes12.jar` in the `OraHome/jdbc/lib` directory.
- For DB2, find `db2java.jar` or `db2java.zip` in the `/Program Files/sqlllib/java` directory.

2 Copy the .jar or .zip file from the previous step to the appropriate destination directory, depending on the Web application server you use.

Table 16 Destination Directory Examples for Web Application Servers

Web Application Server	Destination Directory
Tomcat	<code>drive letter:/Hyperion/Planning/AppServer/InstalledApps/Tomcat/5.0.28/Webapps/HyperionPlanning/WEB-INF/classes</code>
WebLogic	<ol style="list-style-type: none"> Extract the .ear file. Copy the .jar or .zip file to the <code>HPDomain/applications</code> directory. Add the file to the CLASSPATH in <code>startHPServer.cmd</code>.
WebSphere	<code>HYPERION_HOME/products/Planning/AppServer/InstallableApps/Common/HyperionPlanning.ear</code>

3 Stop and restart the Web application server.

Configuring JDBC Connection Pooling

You can set the minimum and maximum number of JDBC connection pools. How you configure these depends largely on the number of Planning users accessing the relational database. For example, you can specify the minimum connection number in JDBC connection pool as one, and the maximum as five. Doing this creates one connection when a user logs on to Planning. An additional connection is created for the next four users who log on to Planning, resulting in a total of five connections. Additional users who log on share the five connections.

The JDBC connection settings are set by default to a minimum of 2 and a maximum of 10. The Planning application does not function correctly if you lower the maximum `JDBC_MAX_CONNECTIONS` to less than 2.

To reconfigure JDBC connection pool parameters, change the minimum and maximum property values:

- `JDBC_MIN_CONNECTIONS`
- `JDBC_MAX_CONNECTIONS`

Setting Offline Compression

You can set up compression for applications taken offline by adding the `OFFLINE_COMPRESSION_THRESHOLD` property and setting the threshold, in bytes, for when to

start using compression. Compression is enabled when the server's response to offline client requests is greater than the Offline Compression Threshold number.

► To set offline compression:

- 1 Select **Administration**, then **Manage Properties**.
- 2 Select **System Properties** to set properties for all Planning applications.
- 3 Update the **Offline Compression Threshold** setting:
 - To add the property, click **Add**. In the blank row, enter `OFFLINE_COMPRESSION_THRESHOLD` (avoid using spaces). Under **Property Value**, enter the number of bytes to use for the threshold for when to compress offline applications.
 - To change the property, change its value under **Property Value**. To disable compression, enter 0.
 - To delete the property, select its name, and press **Delete**.
- 4 Click **Save** and confirm your changes.
- 5 Stop and then restart the application server.

Setting the Maximum Length for Cell Text

By default, the maximum number of single-byte characters allowed for cell text in each cell is 1500, and the column data type is set to `varchar(2000)`. To display additional characters in cell text, set the Planning application property `MAX_CELL_NOTE_SIZE` to the maximum length your application requires.

Note: Updating this setting to more than 2000 requires that you make a corresponding change to the database. If you increase the maximum number of characters allowed for cell text, you must alter the database column size or type to support the changed size. (Changing the column type to CLOB, NCLOB, TEXT, or NTEXT to accommodate large cell text size can affect performance. Do so only if your application requires large cell text entries.) For additional information, see the database documentation provided by your vendor.

► To set the maximum length for cell text:

- 1 Select **Administration**, then **Manage Properties**.
- 2 Select **Application Properties**.
- 3 Click **Add**, and enter this property in the blank row:
`MAX_CELL_NOTE_SIZE`
- 4 Enter a value in **Property Value** to represent the maximum number of single-byte characters allowed for cell text in each cell.
- 5 Click **Save** and confirm your changes.
- 6 Stop, and then restart the Planning server.

- 7 Back up the database, and then update the database column size or type to support the changed size specified in this property.

Controlling Smart View Messages

If you use different releases of Smart View and Planning, a message displays when users start Smart View. To prevent the message from displaying, you can add the system property `SMART_VIEW_DISPLAY_WARNING`.

- To control Smart View messages:
 - 1 Select **Administration**, then **Manage Properties**.
 - 2 Select **System Properties** to set properties for all Planning applications.
 - 3 Click **Add**, and enter this property in the blank row:
`SMART_VIEW_DISPLAY_WARNING`
 - 4 Enter a value in **Property Value**:
 - Yes: Display messages
 - No: Do not display messages
 - 5 Click **Save** and confirm your changes.

Setting Background Processing

You can set jobs—for business rules, Clear Cell Details, and Copy Data—to switch to background processing after a threshold that you configure. After the specified period, jobs execute in the background. You can also set how often Planning checks job status, displayed on the Job Console (see the *Oracle Hyperion Planning User's Online Help*).

- To set background processing and the frequency of status checks:
 - 1 Select **Administration**, then **Manage Properties**.
 - 2 Select **Application Properties** to set properties and values for a Planning application.
 - 3 Add the property by clicking **Add**, entering the property in the blank row, and entering a value in **Property Value**:

Table 17 Job Property Names and Values

PROPERTY_NAME Column	Property Value	Description
HBR_MAX_WAIT_FOR_RULE	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is	For business rules created with Business Rules or those migrated from Business Rules to this release. The interval in milliseconds to wait before running business rules in the background. If this property is not set, the feature does not work, and

PROPERTY_NAME Column	Property Value	Description
	ignored, and the property is set to the default value.	business rules do not run in the background. Note: Business rules that are set to automatically run when a data form is loaded or saved never run in the background.
HBR_MONITOR_DELAY	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	The interval in milliseconds for checking the status of business rules created with Business Rules.
RULE_MAX_WAIT	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	For business rules created with Calculation Manager. The interval in milliseconds to wait before running business rules in the background. If this property is not set, the feature does not work, and business rules do not run in the background. Note: Business rules that are set to automatically run when a data form is loaded or saved never run in the background.
RULE_MONITOR_DELAY	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	The interval in milliseconds for checking the status of business rules created with Calculation Manager.
CLR_CELL_MAX_WAIT	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	The interval in milliseconds to wait before running Clear Cell Details jobs in the background. If this property is not set, Clear Cell Details jobs do not run in the background.
CLR_CELL_MONITOR_DELAY	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	The interval in milliseconds for checking the status of Clear Cell Detail operations.
COPY_DATA_MAX_WAIT	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	The interval in milliseconds to wait before running Copy Data jobs in the background. If this property is not set, Copy Data jobs do not run in the background.
COPY_DATA_MONITOR_DELAY	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is	The interval in milliseconds for checking the status of Copy Data jobs.

PROPERTY_NAME Column	Property Value	Description
	ignored, and the property is set to the default value.	
COPY_DATA_MONITOR_DELAY	The default and minimum value is 180000 milliseconds (3 minutes). If you set a lower value, that value is ignored, and the property is set to the default value.	The interval in milliseconds for checking the status of Copy Data jobs.
JOB_STATUS_MAX_AGE	Specify the value in milliseconds. There is no default or minimum value.	<p>The maximum age of a completed job record before Planning deletes it from the database table, HSP_JOB_STATUS. Planning checks the job records every 30 minutes.</p> <p>For example, if you set the property value to 60,000 (1 minute), a job completes at 3:00, and Planning checks the job records at 3:01, then Planning would delete the completed job record.</p> <p>Removing completed job records can improve performance.</p>

- 4 Click **Save** and confirm your changes.

Logging on to EPM Workspace and Accessing Planning

You work with Planning in the EPM Workspace environment. The default EPM Workspace URL is `http://web_server:port/workspace/`, where `web_server` is the Web server machine hostname and `port` is the Web server listen port, for example, 19000, if using the Apache instance configured with Oracle's Hyperion Reporting and Analysis. Communicate the URL to all Planning users to enable them to log on to EPM Workspace and access Planning.

For information on installing and configuring EPM Workspace, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* and *Oracle Enterprise Performance Management Workspace Administrator's Guide*.

- To log on to EPM Workspace and access Planning:
 - 1 Ensure that the Web server is started and the Web application server is running in the Services panel.
 - 2 In your browser, enter the URL for the EPM Workspace **Log On** page.
 - 3 Enter your user name.
 - 4 Enter your password.
 - 5 Click **Log On**.
 - 6 In EPM Workspace, select **Navigate**, then **Applications**, then **Planning**. Select a Planning application. If prompted, enter your logon information. Planning does not support non-ASCII characters for passwords.

You can log on to several Planning applications simultaneously, and navigate among them in the EPM Workspace tabs. The application names display as tabs at the bottom of the window, and you can click the tabs to move between applications. You can also have two views of the same application in the EPM Workspace tabs.

For information on Performance Management Architect application administration tasks, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Planning supports users on a variety of network bandwidths. The performance on a 56K dialup connection can be enhanced by using HTTP compression on the Planning server.

Suppressing Password Prompts in Planning Utilities

When running Planning utilities that prompt for passwords, administrators can use an option to suppress password prompts, for example, when running utilities in batch mode. To enable suppressing password prompts, use the `PasswordEncryption` utility to create a file that stores a password in encrypted form. After the file is set up, you can run Planning utilities with the `[-f:passwordFile]` option as the first parameter in the command line to skip the password prompt and use the password from the encrypted file. Each password file contains one password, and is stored in the location that you specify when running the utility.

The `PasswordEncryption` utility uses a command line interface. By default, it is installed in: `HYPERION_HOME/products/Planning/bin`.

► To enable suppressing password prompts in Planning utilities:

1 Enter this command from the `bin` directory, where `passwordFile` is the full file path and file name for the password file:

- **Windows:** `PasswordEncryption.cmd passwordFile`
- **UNIX:** `PasswordEncryption.sh passwordFile`

2 When prompted, enter your password.

The masked password is encrypted and stored in the file and location specified in `passwordFile`. For other Planning utilities with password prompts, you can use `[-f:passwordFile]` as the first parameter in the command line to skip the prompt and use the encrypted password from the password file specified in `passwordFile`.

Saving Text Files for Planning Utilities

These Planning utilities use text files that must be saved in UTF-8 format: `FormDefUtil`, `SampleApp_Data`, and `TaskListDefUtil`. Other utilities do not use text files or do not require specific encoding.

For example, if you update the text file for the `FormDefUtil` utility in Notepad, the file is saved in the correct encoding format by default. If you change the Encoding option or create a text file with a different encoding format, such as ANSI or Unicode, the utility does not work correctly. When saving the text file, ensure that the UTF-8 encoding option is selected.

Using Application Servers

You can access Planning applications through application servers. You must register application servers on your workstation.

About Updating Instances and Clusters

When creating applications, you select a data source that is associated with an instance (also called a cluster). If necessary, you can update clusters using the EPM System Configurator. See “Planning Cluster Management” in the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* or Online Help.

Closing Planning and Essbase

- To close Essbase Server:
 - 1 **Maximize the Essbase window.**
 - 2 **Enter Quit.**

- To log off Planning Web pages select **File**, then **Log Off**. You return to the Logon page.

- To close Planning on the Web, select **File**, then **Exit**.

3

Setting Up Access Permissions

In This Chapter

Levels of Access Permissions	39
Planning Elements That Can Be Assigned Access	40
Types of Access Permissions.....	40
Enabling Access Permissions for Dimensions.....	42
Assigning Access to Members and Business Rules.....	42
Managing Access to Data Forms and Folders.....	46
Importing Access Permissions	48
Improving Performance When Importing Access Permissions	51
Exporting Access Permissions	51
Reporting on Access Permissions	54
Managing Security Filters.....	58
Synchronizing Users and Groups With the UpdateUsers Utility	59
Synchronizing Users With the Provision Users Utility	60
Migrating User and Group Identities	61
Migrating Business Rule Security.....	62
Removing Stale User Records	63
Setting Up Access Permissions in Financial Reporting.....	63

Levels of Access Permissions

Setting access permissions to Planning application elements prevents unauthorized users from viewing or changing data. You can set access permissions at these levels:

- Authentication of provisioned users and groups by an external user directory. See the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide* and its appendix on Planning.
- Oracle's Hyperion® Shared Services roles that set access permissions for managing projects, applications, dimensions, users, and groups. For example, users must have these Shared Services roles to perform the specified tasks:
 - Project Manager: Creates and manages projects in Shared Services.
 - Provisioning Manager: Provisions users and groups to applications.
 - Dimension Editor: Required for Performance Management Architect and Classic applications. For Performance Management Architect, allows access to application

administration options for Planning. For Classic, allows access to the Classic Application Administration options for Planning (in combination with the Planning Application Creator role).

- Planning Application Creator: Required for Performance Management Architect and Classic applications. For Performance Management Architect, allows users to create Planning applications and Performance Management Architect Generic applications. For Classic, allows access to the Classic Application Administration options for Planning (in combination with the Dimension Editor role). For detailed information on these roles, see the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.
- User-defined dimensions. Assign access permissions to members by selecting the dimension property Apply Security. If you omit setting Apply Security, all users can access the dimension's members. By default, the Account, Entity, Scenario, and Version dimensions are enabled for access permissions.
- Users and groups, which can vary among applications. Assign access to Planning application elements by using Assign Access.

After updating access permissions, refresh the application to update Essbase security filters.

Planning Elements That Can Be Assigned Access

You can assign access permissions to:

- Scenario members
- Version members
- Account members
- Entity members
- User-defined custom dimension members
- Launch privileges to Calculation Manager business rules
- Data forms
- Data form folders and Calculation Manager business rule folders
- Task lists

When you change the user type for a user, the user has full read/write access to the application through Smart View and Essbase until you refresh the Essbase database. After the database is refreshed, appropriate access permissions are applied to the user.

Types of Access Permissions

Access permissions for the specified user or group to the dimension member, data form, or task list include:

- Read: Allow view access

- Write: Allow view and modify access
- None: Prohibit access; the default access is None

You can also set who can launch which Calculation Manager business rules as:

- **Launch:** Allow launch privileges

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

- **No Launch:** Disallow launch privileges. Note that if a user inherits Launch access permission to a business rule by belonging to a group, and is also assigned No Launch permissions by belonging to another group, the more restrictive No Launch assignment takes precedence.

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

Inheriting Access Permissions

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

Table 18 Options for Inheriting Access Permissions

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

How access permissions are evaluated

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

1. Role-level security. Users with the Administrator role have access to all application elements (except the Mass Allocate role, which must be assigned to use the Mass Allocate feature).
2. For interactive users and planner user types, access permissions that are specifically assigned to users.
3. Access assignments that are acquired by belonging to a group.
4. Parent-level assignments (for example, to parent members or folders).

Enabling Access Permissions for Dimensions

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

► To enable access permissions for dimensions:

- 1 Select **Administration**, then **Dimensions**.
- 2 For **Dimension**, select the dimension to change.
- 3 Click **Edit**.
- 4 In **Dimension Properties**, select **Apply Security** to allow access permissions to be set on its members.

If you do not select this option, there is no security on the dimension, and users can access its members without restriction.

- 5 Click **Save**.

Click Refresh to revert to the previous values.

Assigning Access to Members and Business Rules

Before you can assign access to members of user-defined custom dimensions, you must select the Apply Security check box on the dimension's Property tab. See [“Enabling Access Permissions for Dimensions” on page 42](#).

► To assign access to members or Calculation Manager business rules:

- 1 Select the member or business rule:
 - For members: Select **Administration**, then **Dimensions**, then select the dimension and member.
 - For Calculation Manager business rules: Select **Administration**, then **Business Rule Security**. Select the folder containing the business rules, then select the business rules.
- 2 Click **Assign Access**.
- 3 **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
- 4 **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.
- 5 Add, change, or remove access.

See [“Adding Access” on page 43](#), [“Editing Access” on page 43](#), and [“Deleting Access” on page 44](#).

Adding Access

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

➤ To assign access permissions to members or business rules:

1 Select the member or business rule:

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

2 Click **Assign Access.**

- **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
- **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.

3 Click **Add Access.**

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

- Click **Users** to display all user names; click **Groups** to display all groups.
- If there are multiple pages of users or groups, type the page number to go to in **Page**, and click **Go**.
- Click **Start** or **End** to navigate to the first or last page.
- Click **Prev** or **Next** to move to the previous or next page.

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

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

6 **For Calculation Manager business rules only:**

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

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

8 Click **Close.**

Editing Access

You can access the Edit Access page for a member by clicking **View** in the **Security** column on the **Dimensions** tab.

➤ To modify access permissions for members or Calculation Manager business rules:

1 Select the member or business rule:

- For members: Select **Administration**, then **Dimensions**, then select the dimension and member.
 - For Calculation Manager business rules: Select **Administration**, then **Business Rule Security**. Select the folder containing the business rules, then select the business rules.
- 2 Click **Assign Access**.
 - **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
 - **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.
 - 3 Click **Edit Access**.
 - 4 For the selected member or business rule, select the access type for the displayed users or groups.

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

For business rules only:

 - Click **Launch** to allow the selected users and groups to launch the selected business rules.
 - Click **No Launch** to prevent the selected users and groups from launching the selected business rules.
 - 5 **Optional for members:** Select a relationship.

For example, select Children to assign access to children of the selected member.
 - 6 Click **Set**.
 - 7 Click **Close**.

Deleting Access

- To remove access permissions for members or Calculation Manager business rules:
 - 1 **Select the member or business rule:**
 - For members: Select **Administration**, then **Dimensions**, then select the dimension and member.
 - For Calculation Manager business rules: Select **Administration**, then **Business Rule Security**. Select the folder containing the business rules, then select the business rules.
 - 2 Click **Assign Access**.
 - **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
 - **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.
 - 3 **Select the users and groups for whom to remove access to the selected member or business rule.**

Click Users to display all user names; click Groups to display all groups.
 - 4 Click **Remove Access**.

- 5 Click **OK**.
- 6 Click **Close**.

Assigning Access Permissions to Shared Members

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

Existing inheritance access permissions apply to shared members. The precedence order for access permissions is checked at each level, first by user then by group, based on the member's access permissions inheritance relationship. If there are multiple access permissions, inheritance rules are additive, so the highest access permission is applied.

This example assigns access rights to shared members by assigning access to the parent of the shared member. The security filters with the effective access described in the table below is generated only if the database is refreshed or created with the Security and Shared Members options in the Manage Database Page.

Example: Access permissions

Entity		
	United States	
		CA (base)
		NY
	West	
		CA (shared)
		NV
	Sales Region 1	
		CA (shared)

Table 19 Example of Access with Shared Members

Case	Access Permission	Effective Access for CA
Case 1	CA (base) = None iDescendants (West) = Read	Read
Case 2	iDescendants (United States) = None iDescendants (West) = Read iDescendants (Sales Region 1) = Write	Write
Case 3	iDescendants (United States) = Write iDescendants (West) = None	Write

Case	Access Permission	Effective Access for CA
	iDescendants (Sales Region 1) = Read	

Managing Access to Data Forms and Folders

Assigning Access to Data Forms and Folders

You can assign access to data forms, data form folders, and Calculation Manager business rule folders. (For information on assigning access to business rules and members, see [“Types of Access Permissions” on page 40](#) and [“Assigning Access to Members and Business Rules” on page 42](#).)

Principles:

- **Data forms:** Planners can view or enter data only into data forms to which they have access (and can work only with members to which they have access). Administrators and interactive users have write access to all data forms for design modifications.
- **Calculation Manager business rules:** Planners can see and launch only business rules to which they are assigned Launch access.
- **Data form folders and Calculation Manager folders:**
 - Planners who are assigned access to a data form folder can access the data forms in that folder, unless they are assigned more specific access. Likewise, planners have Launch access to the Calculation Manager business rules in folders to which they are assigned access, unless they are assigned more specific access.
 - When you assign access to a folder, all folders under it inherit that access.
 - If you assign specific access (for example, None or Write) to a data form folder, that access permission takes precedence over its parent folder's access permissions. For example, if a user has Write access to Folder1 that contains Folder2 to which the user has None access, the user can open Folder1, but does not see Folder2.
 - If you assign specific access (for example, Launch) to a Calculation Manager folder, that access permission takes precedence over its parent folder's access permissions. For example, if a user has Launch access to RulesFolder1 that contains RulesFolder2 to which the user has No Launch access, the user can open RulesFolder1, but does not see RulesFolder2.
 - If a user has None access to a data form folder called Folder1 that contains a data form called Form1 to which the user has Write access, the user can see Folder1 and Form1.
 - If a user has No Launch access to a Calculation Manager folder called RulesFolder1 that contains a business rule called Rule1 to which the user has Launch access, the user can see RulesFolder1 and Rule1.

For procedures, see [“Adding Access to Data Forms and Folders” on page 47](#).

Adding Access to Data Forms and Folders

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

- 1 **Select the data form or folder.**

- For data forms and folders, see [“Selecting Data Forms and Folders” on page 111](#).
- For business rule folders, select **Administration**, then **Business Rule Security**.
- For business rules, select **Administration**, then **Business Rule Security**. Open the business rule folder containing the business rules and select the rules.

You can assign access to one data form, business rule, or folder simultaneously.

- 2 **Click **Assign Access**.**

- **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
- **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.

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

- Click **Users** to display all user names; click **Groups** to display all groups.
- If there are multiple pages of users and groups, type the page number to go to in **Page**, and click **Go**.
- Click **Start** or **End** to navigate to the first or last page.
- Click **Prev** or **Next** to move to the previous or next page.

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

For business rules or their folders only:

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

- 5 **Click **Add**.**

- 6 **Click **Close**.**

Changing Access to Data Forms and Folders

- To change which users can use or change data forms or folders:

- 1 **Select the data form or folder.**

- For data forms and folders, see [“Selecting Data Forms and Folders” on page 111](#).
- For Calculation Manager business rule folders, select **Administration**, then **Business Rule Security**.

- 2 **Click **Assign Access**.**

- **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
 - **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.
- 3 **Select the users or groups for which to change access, and click **Edit Access**.**
Click Users to display all user names; click Groups to display all groups.
 - 4 **For **Type of Access**, select the kind of access users or groups have to the data form or folder.**
 - 5 **Click **Set**.**
 - 6 **Click **Close**.**
- To remove access from data forms or folders:
- 1 **Select the data form or folder.**
 - For data forms and folders, see [“Selecting Data Forms and Folders” on page 111](#).
 - For Calculation Manager business rule folders, select **Administration**, then **Business Rule Security**.
 - For Calculation Manager business rules, select **Administration**, then **Business Rule Security**. Open the business rule folder containing the business rules and select the rules.
 - 2 **Click **Assign Access**.**
 - **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
 - **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.
 - 3 **Select the users or groups for which to remove access, and click **Remove Access**.**
Click Users to display all user names; click Groups to display all groups.
 - 4 **Click **OK**.**

Importing Access Permissions

The `ImportSecurity` utility loads access permissions for users or groups from a text file into Planning. (To add users or groups, see the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.) Importing access permissions overwrites existing access assignments only for imported members, data forms, data form folders, task lists, Calculation Manager business rules, and Calculation Manager business rule folders. All other existing access permissions remain intact. The `SL_CLEARALL` parameter clears all existing access permissions; you can use it with other parameters to replace existing access permissions. See also [“Exporting Access Permissions” on page 51](#).

The `ExportSecurity` utility automatically creates the `SecFile.txt` file, from which you can import access permissions. If you prefer, you can also manually create the `SecFile.txt` file using these guidelines:

- You must name the text file `SecFile.txt` and save it in the `bin` directory. If you installed Planning to the default location, it is in this path: `HYPERION_HOME/products/Planning/bin`.
- All users, groups, and artifacts must be defined in the application.
- Before importing access permissions on a user-defined custom dimension, you must allow access permissions to be set on it by selecting Apply Security (see [“Enabling Access Permissions for Dimensions” on page 42](#)).
- Each line in the `SecFile.txt` file must specify access permissions information.

Each line must contain these items, separated by one of these delimiters: comma (,) Tab, semi-colon (;), pipe (|), colon (:), space (). Comma is the default.

Item	Description
<i>username or group name</i>	<p>The name of a user or group defined in Shared Services Console.</p> <p>To import access permissions information into a group with the same name as a user, append this information to the line in the <code>SecFile.txt</code> file that pertains to the group: <code>sl_group</code></p> <p>For example:</p> <pre>admin,member1,read,member admin,member1,read,member,sl_group</pre>
<i>artifact name</i>	<p>The named artifact for the imported access permissions (for example the member, data form, task list, folder, or Calculation Manager business rule). Example: <code>Account1</code>.</p> <p>If an artifact name contains a character that you are using as the delimiter, enclose the name in double quotes. For example, if you are using a space as the delimiter, enclose the name South America in double quotes: <code>"South America"</code>.</p>
<i>access permissions</i>	<p>Read, ReadWrite, or None. If there are duplicate lines for a user/member combination, the line with ReadWrite access takes precedence. For example, for these lines:</p> <pre>User1,Member1,Read,@ICILDREN User1,Member1,ReadWrite,@ICILDREN</pre> <p>Access permissions for User1 to Member1 are applied as ReadWrite.</p> <p>For Calculation Manager business rules and folders only: specify launch access permissions as either None or Launch.</p>
<i>Essbase access flags</i>	<p>@CHILDREN, @ICILDREN, @DESCENDANTS, @IDESCENDANTS and MEMBER.</p> <p>Security implementation for these functions is identical to Essbase.</p>
<i>artifact type</i>	<p>For artifacts other than members, distinguish which artifact you are importing security for with <i>artifact type</i> identifier:</p> <ul style="list-style-type: none"> ● <code>SL_FORM</code>—for data forms ● <code>SL_COMPOSITE</code>—for composite data forms ● <code>SL_TASKLIST</code>—for task lists ● <code>SL_CALCRULE</code>—for Calculation Manager business rules (not business rules in Business Rules) ● <code>SL_FORMFOLDER</code>—for data form folders ● <code>SL_CALCFCOLDER</code>—for folders containing Calculation Manager business rules

Item	Description
	<p>Note: The <code>ExportSecurity</code> utility automatically adds the required <code>artifact type</code> identifiers in the <code>SecFile.txt</code> file. If you manually create the <code>SecFile.txt</code> file, you must add the <code>artifact type</code> identifiers.</p> <p>Note: The <code>ExportSecurity</code> utility does not support exporting access permissions to task lists for administrators, so you must manually add such records to the <code>SecFile.txt</code> file before you can import them.</p>

Sample lines from a file:

```
User1,Account1,read,@CHILDREN
Group2,DataForm08,readwrite,MEMBER,SL_FORM
User3,TaskList09,readwrite,MEMBER,SL_TASKLIST
NorthAmericaGroup,Sales,write,MEMBER,SL_FORMFOLDER
```

➤ To import access permissions into Planning:

- 1 **Locate the `ImportSecurity` utility by navigating to the `bin` directory.**
- 2 **From the Command Prompt, enter this case-sensitive command, one space, and the parameters, separating each with a comma. Enclose the parameters with double quotation marks:**

```
ImportSecurity.cmd [-f:passwordFile] "appname,username,[delimiter],
[RUN_SILENT],[SL_CLEARALL]"
```

where:

Parameter	Description
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37 .
<code>appname</code>	Name of the Planning application to which you are importing access permissions.
<code>username</code>	Planning administrator user name.
<code>delimiter</code>	Optional: <code>SL_TAB</code> , <code>SL_COMMA</code> , <code>SL_PIPE</code> , <code>SL_SPACE</code> , <code>SL_COLON</code> , <code>SL_SEMI-COLON</code> . If no delimiter is specified, comma is the default.
<code>RUN_SILENT</code>	Optional: Execute the utility silently (the default) or with progress messages. Specify 0 for messages, or 1 for no messages.
<code>[SL_CLEARALL]</code>	Optional: Clear existing access permissions when importing new access permissions. Must be in uppercase.

For example:

```
ImportSecurity.cmd "app1,admin,SL_TAB,1"
```

To clear all access permissions, enter:

```
ImportSecurity.cmd "app1,admin,,,SL_CLEARALL"
```

- 3 If prompted, enter your password.
- 4 After you execute the utility, check the log file `importsecurity.log` in the `bin` directory to verify the results.

Improving Performance When Importing Access Permissions

- To import access for many users, improve performance by not using full names:
 - 1 On Planning Web, select **Preferences**, then **Advanced Settings**, and then **System Settings**.
 - 2 Clear **Display Users' Full Names**.

Exporting Access Permissions

The `ExportSecurity` utility exports Planning access permissions to the `SecFile.txt` file, enabling you to export and import access permissions across applications (see “[Importing Access Permissions](#)” on page 48). For the specified user or group (or for all users and groups if you use only the mandatory parameters), the `ExportSecurity` utility exports access permissions to these artifacts: members, data forms, data form folders, task lists, Calculation Manager business rules, and Calculation Manager business rule folders. `ExportSecurity` appends an artifact type flag that specifies whether the exported artifact security is for a data form, composite data form, data form folder, task list, Calculation Manager business rule, or Calculation Manager folder.

Notes:

- If you specify only mandatory (not optional) parameters, all access permissions to all artifacts for all users and groups are exported. You can limit the export by specifying a member parameter (but only one member-based parameter).
 - You can specify the optional parameters in any order.
 - You can use only `/S_USER` or `/S_GROUP`, not both.
 - Use the `/S=searchCriteria` parameter to specify users and groups with the same name.
 - Running the utility creates a file named `SecFile.txt`, which contains the exported access permissions.
- To export access permissions from Planning to a text file:
 - 1 Navigate to the `bin` directory in this path: `HYPERION_HOME/products/Planning/bin`.
 - 2 From the Command Prompt, enter this case-sensitive command, one space, and the parameters. Separate each parameter with a comma:

```
ExportSecurity.cmd [-f:passwordFile] /A=appName, /U=username, [/S=searchCriteria|/S_USER=user|/S_GROUP=group], [/S_MEMBER=memberName|/S_MEMBER_ID=memberName|/S_MEMBER_D=memberName|/S_MEMBER_IC=memberName|/S_MEMBER_C=memberName], [/DELIM=delim], [/DEBUG=true|false], [/TO_FILE=fileName], [/HELP=Y]
```

where:

Parameter	Description	Mandatory?
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in <i>passwordFile</i> . See “Suppressing Password Prompts in Planning Utilities” on page 37.	No
<code>/A=appName</code>	The name of the Planning application from which you are exporting access permissions.	Yes
<code>/U=username</code>	The administrator's ID for logging into the application.	Yes
<code>/S=searchCriteria</code>	The user or group name. You cannot use this option with <code>/S_USER</code> or <code>/S_GROUP</code> .	No
<code>/S_USER=user</code>	A specified user name. You cannot specify multiple users or use this option with <code>/S_GROUP</code> or <code>/S=searchCriteria</code> .	No
<code>/S_GROUP=group</code>	A specified group. Only matching groups, not matching user names, are exported. You cannot specify multiple groups or use this option with <code>/S_USER</code> or <code>/S=searchCriteria</code> .	No
<code>/S_MEMBER=MemberName</code>	A specified member. You can specify only one member-based parameter.	No
<code>/S_MEMBER_ID=MemberName</code>	A specified member and its descendants.	No
<code>/S_MEMBER_D=MemberName</code>	A specified member's descendants.	No
<code>/S_MEMBER_IC=MemberName</code>	A specified member and its children.	No
<code>/S_MEMBER_C=MemberName</code>	A specified member's children.	No
<code>/DELIM=delim</code>	Specify one of these delimiters: , Tab ; : space. Comma is the default. Note: To specify a space, press the Space Bar; to specify a tab, press the Tab key.	No

Parameter	Description	Mandatory?
/DEBUG=	Specify <code>true</code> to display the utility's performed steps. <code>false</code> is the default.	No
/TO_FILE=	Specify the path to the <code>SecFile.txt</code> file. By default, the file is in this path: <code>HYPERION_HOME/products/Planning/bin</code> . If you specify another path, use double backslashes, for example: <code>C:\\Hyperion\\SecFile.txt</code> .	No
/HELP=Y	Specify as the only parameter to display the syntax and options for <code>ExportSecurity</code> .	No

For example, to export access permissions for a user and group named `Sales`, enter:

```
ExportSecurity.cmd /A=app1,/U=admin,/S=Sales
```

To export for a member named `Account100` and its descendants, with the colon delimiter to a file named `Account100.txt` in a specific path:

```
ExportSecurity.cmd /A=planapp1,/U=admin,/TO_FILE=D:\\Hyperion\\Product\\Planning\\bin\\Account100,/S_MEMBER_ID=Account100,/DELIM=:
```

3 If prompted, enter your password.

Also note:

- If a member, user, or group name contains a character used as the delimiter, the name is enclosed in double quotes. For example, if a space is the delimiter, the name `South America` is enclosed in double quotes: `"South America"`.
- Because commas are used to separate parameters, if a parameter contains commas (for example, `plan,app1`), precede it with a backslash. Also use backslash to escape the backslash from the command prompt. In this example, use two backslashes: `/A=plan\\,app1`
- The `ExportSecurity` utility does not support exporting access permissions to task lists for administrators, so you must manually add such records to the `SecFile.txt` file before you can import them.

Understanding the export file:

Item	Description
<i>user or group</i>	The name of a user or group defined in Shared Services Console.
<i>memName</i>	A member in the application.
<i>access permissions</i>	Read, Write or None. If there are duplicate lines for a user name/member name combination, the line with Write access takes precedence.





Item	Description
<i>Essbase access flags</i>	@CHILDREN, @ICHILDREN, @DESCENDANTS, @IDESCENDANTS, and MEMBER. Security implementation for these functions is identical to Essbase.
<i>artifact type</i>	After each line, the utility appends the artifact type: <ul style="list-style-type: none"> ● SL_FORM—for data forms ● SL_COMPOSITE—for composite data forms ● SL_TASKLIST—for task lists ● SL_CALCRULE—for Calculation Manager business rules (not business rules in Business Rules) ● SL_FORMFOLDER—for data form folders ● SL_CALCFCOLDER—for folders containing Calculation Manager business rules <p>Note: If you manually create the <code>SecFile.txt</code> file, you must add the artifact type identifiers.</p>

For example, an exported file might contain these lines:

```
User1,DataForm2,read,MEMBER,SL_COMPOSITE
User2,Folder3,write,MEMBER,SL_FORMFOLDER
User3,DataForm4,write,MEMBER,SL_FORM
"North America",Account101,write,MEMBER,SL_CALCFCOLDER
```

Reporting on Access Permissions

You can view current access permissions and print reports.

- To report on current access permissions for users and groups in Planning:
 - 1 In Shared Services Console, select a Planning application under **Projects**. Select **Administration**, then **Access Control Report**.
 - 2 On **Select User or Group**, select options:
 - Available Users
 - Available Groups
 - Available Users and Groups
 - 3 From the left **Available** panel, select and move users or groups to report on to the **Selected** panel:
 - To move selections, click .
 - To remove selections, click .
 - To move all users or groups, click .
 - To remove all users and groups, click .

If you enter a user or group name instead of browsing to it, you must enter the full name. For names with commas, enclose the name in quotation marks.

4 Click **Next**.





Selecting Reporting Objects

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

► To select reporting objects:

1 Start the Access Control Report.

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

- To move selections to **Selected Objects**, click .
- To remove selections, click .
- To move all objects, click .
- To remove all objects, click .

3 Click **Next**.

Selecting Reporting Options

► To specify options for access reports:

1 Start the Access Control Report.

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

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

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

Table 20 Access Report Types

Report Type	Description	Options
Assigned Access	Summarizes access permissions that administrators assign	Specify whether access is assigned by member selection relation or group membership: <ul style="list-style-type: none"> ● Show Matching Access of Relation: Member, Children, Children (inclusive), Descendants, or Descendants (inclusive). ● Show Inherited From Group: Show access permissions inherited by users in a group.

Report Type	Description	Options
Effective Access	Summarizes access assignments as Planning evaluates them (for example, by member selection relation, such as children, or group membership). This is useful if there are conflicts in access permissions.	Describe the origin of the effective access by selecting Show Effective Access Origin. For example, a user named JSomebody may be assigned Write access to Entity1 and belong to a group named Sales that is assigned Read access to Entity1. This setting would show that JSomebody has Write access to Entity1 because individual assigned access supersedes access inherited by group membership.

5 Click Finish.

Adobe Acrobat launches, displaying the report online.

Working With Access Permissions Reports

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

Setting up Audit Trails

Administrators can select aspects of the application for change tracking. For example, you can track changes to metadata, such as when users change a member property or add a currency. You can also track changes in data forms, business rules, workflow, users, access permissions, and so on. Access assignments imported with the `ImportSecurity` utility are not reflected in audit reports. To view audits, administrators create and run reports using RDBMS report writers.

Table 21 Actions That Can be Audited

Audit Options	Tracked Changes
Dimension Administration	<ul style="list-style-type: none"> ● Dimension hierarchy: adding a member or dimension, moving, deleting, changing properties, renaming a member or dimension ● Performance settings: resetting a dimension's dense or sparse setting, changing the order of dimensions ● Currencies: adding or deleting currencies, setting a triangulation or reporting currency ● Updates by utilities that affect Planning (such as importing data form designs with the <code>ImportFormDefinition</code> utility)
Alias Table Administration	Changes to alias tables: creating, copying, renaming, deleting, and clearing
Data	<ul style="list-style-type: none"> ● Cell values ● Supporting detail ● Account annotations ● Cell-level documents
Launch Business Rules	Updates from calc scripts and business rules (including runtime prompts)

Audit Options	Tracked Changes
Data Form Definition	Data forms: creating, modifying, adding rows. (The audit record does not record how the design changed.)
Data Form Folder Administration	Folders: created, moved, or deleted
Workflow	Process management: planning unit owners, status, and status (started or excluded)
Copy Version	Versions copied, including supporting detail and annotations. The audit record does not record details (such as data, supporting detail, and annotations) of the copied version.
Security	Access permissions to dimension members, data forms, data form folders, business rules, and task lists
Users Administration	Users added, changed, or deleted
Groups Administration	Groups added, changed, or deleted; users added or removed
Offline	Data forms taken offline or synchronized back to the server
Task List	Task lists: created, copied, saved, moved, and deleted
Copy Data	Users' selections for Static Dimensions, Source Dimension, and Destination Dimension
Clear Cell Details	Users' selections for clearing supporting detail, cell text, and account annotations

► To specify aspects of the application for which Planning records changes:

- 1 **Select Administration, then Reporting.**
- 2 **Select Auditing.**
- 3 **Select the actions Planning tracks.**

To avoid affecting performance, be selective in which application elements you audit.

- 4 **Click Save Selections.**

Depending on selected audit options, application changes are recorded in a HSP_AUDIT_RECORDS table, stored in the relational database.

- 5 **Restart the Web application server.**
- 6 **View results in the HSP_AUDIT_RECORDS table using a RDBMS report writer.**

If anyone resets audit options, those changes are recorded.

Examples of Tracked Actions

For each recorded action, Planning tracks:

Tracked Changes	Examples
The type of change	Metadata, data, data form, access permissions, planning units

Tracked Changes	Examples
The affected object (The columns ID_1 and ID_2 in the audit report help define the object that changed.)	Data Form: Expenses 04 Group: Marketing
User	VHennings
Time Posted	12/22/2008 8:17
Action	Add
Property	Currency
Old value	Default
New value	USD

Viewing and Clearing Audit Reports

Audit results are recorded in the HSP_AUDIT_RECORDS table, stored in the relational database. To clear the audit report, use the SQL DELETE command on the HSP_AUDIT_RECORDS table. To clear entries that are a certain number of days old, compare them against the time_posted field. For example, to delete all entries from the table:

```
DELETE FROM HSP_AUDIT_RECORDS
```

To view audit records, sorted by the time they were posted:

```
SELECT * FROM HSP_AUDIT_RECORDS ORDER BY TIME_POSTED
```

Managing Security Filters

Access permissions in Planning are stored in the relational database. If you use other products outside of Planning, such as Financial Reporting or third-party tools, to access Planning data directly in Essbase, you must push Planning access permissions to Essbase by generating security filters.

To update security filters in Essbase for selected users, select Administration, then Manage Security Filters. To update security filters simultaneously for all users, select Administration, then Manage Database, and then Security Filters (see [“Creating and Refreshing Application Databases” on page 67](#)). First validate that the size of security filters does not exceed the Essbase limit of 64 KB per row.

For read and write security filters to be generated in Essbase, users must have read or write access permissions to at least to one member from each secured Planning dimension, including user-defined dimensions. If access is not assigned in these dimensions, the security filter for the user in Essbase is set to None.

- To create or update individual security filters:
 - 1 From Planning, select **Administration**, then **Manage Security Filters**.

- 2 Select the users whose security filters you want updated.
- 3 Click **Create**.

Essbase creates an encrypted file (`Essbase.sec`) to store access permissions information.

Note: If you want planners and interactive user types to have write access directly to Planning data in Essbase, assign them the role “Essbase Write Access” in Shared Services.

Synchronizing Users and Groups With the UpdateUsers Utility

If you move users and groups, their identities (SIDs) change, and unless you synchronize their identities between Planning and Shared Services, their assigned access permissions are lost. Situations requiring synchronization:

- You change authentication providers.
- Users, groups, or organizational units (OUs) are moved in an external provider.
- You migrate your application from one environment to another (for example, from Dev to Prod) and plan to change Shared Services repositories.

You can use the Security Filters option in Manage Database to synchronize such changes (see “Creating and Refreshing Application Databases” on page 67). Alternatively, you can use two utilities in concert to synchronize changes:

1. Run the `UpdateNativeDir` utility to update user and group identities in Shared Services.
`UpdateNativeDir` and its documentation are in the `HYPERION_HOME/Common/Utilities/SyncOpenLDAPUtility` directory.
2. Run the `UpdateUsers.cmd` utility to update the SIDs in Planning with the changes in Shared Services.

Note: It is important to run the utilities in this order.

► To launch the `UpdateUsers.cmd` utility:

- 1 At the Command Prompt, from the `bin` directory, enter the command, using this syntax:

```
updateusers.cmd [-f:passwordFile] serverName adminName applicationName
```

If you installed Planning in the default location, the `bin` directory is in this path:
`HYPERION_HOME/products/Planning/bin`.

Table 22 UpdateUsers Syntax

Parameter	Description	Required?
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in	No

Parameter	Description	Required?
	<i>passwordFile</i> . See “ Suppressing Password Prompts in Planning Utilities ” on page 37.	
<i>serverName</i>	The server on which the Planning application resides	Yes
<i>adminName</i>	The administrator's name for logging on to the Planning application.	Yes
<i>applicationName</i>	The Planning application for which to synchronize users and groups (must be on the server on which the utility is run).	Yes
/?	Specified by itself, prints the syntax and options for <code>UpdateUsers</code> .	No

For example:

```
updateusers.cmd ABCserver Planningapp
```

- 2 If prompted, enter your password.
- 3 **Optional:** To view information on the results, open the `UpdateUsers.log` file in the `bin` directory.

Synchronizing Users With the Provision Users Utility

The `ProvisionUsers` utility—run by administrators through a command line interface—synchronizes Planning users, groups, and roles in Shared Services Console with a Planning application and with Essbase. You can use this utility as an alternative to using the Security Filters option in Manage Database (see “[Creating and Refreshing Application Databases](#)” on page 67).

► To use the utility:

- 1 **Launch the `ProvisionUsers.cmd` file from the `bin` directory, using the following syntax:**

```
ProvisionUsers [-f:passwordFile] /ADMIN:adminName /A:appName [/U:user1[;user2;user3]] [/R:n]
```

If you installed Planning in the default location, the `bin` directory is in this path:

```
HYPERION_HOME/products/Planning/bin.
```

Table 23 `ProvisionUsers` Syntax

Parameter	Description	Required?
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in <i>passwordFile</i> . See “ Suppressing	No

Parameter	Description	Required?
	Password Prompts in Planning Utilities on page 37.	
<code>/ADMIN:adminName</code>	The administrator's name for logging on to the Planning application.	Yes
<code>/A:appName</code>	The Planning application to synchronize (must be on the server on which the utility is run).	Yes
<code>[/U:user1[;user2;user3]]</code>	Specifies users to synchronize. For example, to synchronize users Planner1 and Planner2, use <code>/U:Planner1;Planner2</code> . Omitting this argument synchronizes all users.	No
<code>[/R:n]</code>	Specifies an interval, in minutes, in which synchronization is run. For example, to synchronize every 30 minutes, use <code>/R:30</code> . Omitting this argument performs the synchronization once.	No
<code>/?</code>	Specified by itself, prints the syntax and options for <code>ProvisionUsers</code> .	No

2 If prompted, enter your password.

Example 1

Entering:

```
ProvisionUsers /ADMIN:admin /A:App1
```

Synchronizes all users in the App1 application.

Example 2

Entering:

```
ProvisionUsers /ADMIN:admin /A:App2 /U:Planner1 /R:60
```

Synchronizes user Planner1 in the App2 application every 60 minutes.

Migrating User and Group Identities

When you change a user or group's identity or their position in the user directory hierarchy, you must update—or migrate—this information to Planning.

- ▶ To migrate changed user and group identities from Shared Services Console to Planning:

1 Take an action:

- Select **Administration**, then **Dimensions**, and select a dimension member.
- Select **Administration**, then **Manage Data Forms**, and select a data form.

- If using Calculation Manager: Select **Administration**, then **Business Rule Security**, and select a business rule folder or business rule.
 - Select **Administration**, then **Manage Task Lists**, and select a task list.
- 2 Click **Assign Access**.
 - 3 Click **Migrate Identities**.

Migrating Business Rule Security

Using the `HBRMigrateSecurity.cmd` utility, administrators can migrate launch access permissions on business rules and their projects from Business Rules to Planning.

The `HBRMigrateSecurity.cmd` utility:

- Overwrites any launch access permissions that are already assigned to business rules in the specified Planning application.
 - Migrates access permissions only for users and groups that are provisioned for the specified Planning application in Shared Services Console.
 - Automatically deploys access permissions for business rules in the specified Performance Management Architect Planning application.
- To migrate access permissions on business rules and their folders:
- 1 **Before running `HBRMigrateSecurity.cmd`:**
 - If you are using Performance Management Architect, upgrade the Planning application to Performance Management Architect.
 - Migrate business rules from Business Rules to Calculation Manager.
 - Deploy the business rules to Planning.
 - 2 **At the Command Prompt, from the `bin` directory, enter this command and its parameters, separating each by a space:**

```
HBRMigrateSecurity.cmd [-f:passwordFile] /A:appname /U:admin /F:output file
```

If you installed Planning to the default directory, `HBRMigrateSecurity.cmd` is installed in `HYPERION_HOME\Products\Planning\bin`.

Table 24 HBRMigrateSecurity Parameters

Parameter	Purpose	Required?
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37.	No

Parameter	Purpose	Required?
<i>/A: appname</i>	Specify the Planning application to which to migrate launch access permissions for business rules	Yes
<i>/U: admin</i>	Specify the administrator's user name	Yes
<i>/F: output file</i>	Specify the name of the XML output file, including its full path if it is not in the <code>bin</code> directory (this file contains a log of the transaction, and helps with troubleshooting)	Yes
<i>/?</i>	Print the syntax and options for <code>HBRMigrateSecurity.cmd</code>	No

3 If prompted, enter your password.

For example:

```
HBRMigrateSecurity.cmd /A:appname /U:admin /F:C:\temp
\HBRExportedSecurity.xml
```

Removing Stale User Records

When you deprovision or delete users or groups in Shared Services, you can conserve disk space by updating the user and group tables in the Planning relational database by removing stale records.

► To remove deprovisioned users and groups from the Planning database tables:

1 Take an action:

- Select **Administration**, then **Dimensions** and select a dimension member.
- Select **Administration**, then **Manage Data Forms** and select a data form folder or data form.
- If using **Calculation Manager**: Select **Administration**, then **Business Rule Security** and select a business rule folder or business rule.
- Select **Administration**, then **Manage Task Lists** and select a task list.

2 Click **Assign Access**.

3 Click **Remove Non-provisioned Users/Groups**.

Setting Up Access Permissions in Financial Reporting

Financial Reporting supports these access permissions:

- User authentication
 - Logon access permissions

- Access to Financial Reporting and data source
- Application permissions
 - Access to tasks within Financial Reporting
 - Permissions to design or view reports
- Data Rights
 - Access to data source data such as members and values
 - Access to Financial Reporting objects such as reports

4

Managing Planning Databases

In This Chapter

Opening Applications	65
Unlocking Applications	65
Using Broadcast Messaging	66
Viewing Usage Statistics	67
Creating and Refreshing Application Databases	67
Managing Exchange Rates	69
Managing Currency Conversion	72
Working with Currency Conversion Calc Scripts	72
Optimizing Application Performance	73
Backing Up Applications and Application Databases	74

For Classic Planning applications, databases are created and maintained within Planning. For Performance Management Architect applications, applications are created in Performance Management Architect and deployed to Planning. For information on tasks performed in Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Opening Applications

You can open applications to work with data or run reports. Multiple applications can reside on the same server and be open at the same time. This can slow performance, depending on the server's configuration, the processor speed of your computer, and available memory. To determine server configuration, contact your Oracle consultant.

Unlocking Applications

Occasionally Planning applications can become locked, for example, if users abnormally exit the application and Planning. The Unlock Application utility clears all records in the `HSP_LOCK` table. You must run the utility from the Planning application server.

Ensure there are no users connected to the Planning application before running the utility. To confirm this, launch the task manager on the Planning server and ensure there are no processes called `hsxser~1` (`hsxserver`) or `hspds`.

- To unlock Planning applications:
 - 1 **Locate the `HspUnlockApp.cmd` utility by navigating to the `bin` folder using the command line.**
For example, if you installed Planning in the default location, the file is in `HYPERION_HOME/products/Planning/bin`.
 - 2 **Enter `HspUnlockApp.cmd [-f:passwordFile] SERVER_NAME USER_NAME PASSWORD APPLICATION_NAME`, where *application name* is the application to unlock.**
Optional: If an encrypted password file is set up, use `[-f:passwordFile]` as the first parameter in the command line to read the password from the full file path and name specified in *passwordFile*. See [“Suppressing Password Prompts in Planning Utilities” on page 37](#).
 - 3 **If prompted, enter your password.**
 - 4 **Check the application event logs using the Event Viewer in the console application log to determine whether a success or failure event is reported. For information about logs, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.**

Using Broadcast Messaging

Use broadcast messaging to communicate a text message to all Planning users currently logged on to an application. For example, you can send messages about system availability or periodic maintenance. You should also send broadcast messages to request that users log out before upgrading or migrating applications.

You can send broadcast messages using the Web client or a command line utility. If you send them using the Web, they are sent to users of your current application. If you send them using the command line, you can specify any application, without being logged on to it. You can also schedule messages using standard operating system mechanisms.

For users logged on to the application, the broadcast message displays in their browser the next time they refresh the page or go to a different page, whether you send the broadcast message through the Web or the command line. Only users who are currently logged on to the Planning application see broadcast messages. Users who are logged on to the application through other products or third-party reporting tools do not receive broadcast messages.

- To send broadcast messages using Planning Web:
 - 1 **Select **Tools**, then **Broadcast Message**.**
 - 2 **In **Create Message**, enter the message to broadcast.**
 - 3 **Click **Send**.**
- To send broadcast messages using a command line:
 - 1 **Locate the `BroadcastMessage.cmd` utility by navigating to the `bin` folder with a command prompt.**
For example, if you installed Planning to the default location, the file is in `HYPERION_HOME/products/Planning/bin`.
 - 2 **Launch the utility, using this syntax at the command line:**

```
broadcastmessage.cmd ([SERVER_NAME], APPLICATION_NAME, USER_NAME, MESSAGE)
```

Optional: If an encrypted password file is set up, use [-f:passwordFile] as the first parameter in the command line to read the password from the full file path and name specified in passwordFile. See [“Suppressing Password Prompts in Planning Utilities” on page 37](#).

SERVER_NAME: the localhost name.

APPLICATION_NAME: the name of the application to whose users you send messages.

USER_NAME: the administrator who has rights to send broadcast messages.

MESSAGE: a text message of up to 127 characters to send to application users.

- 3 If prompted, enter your password.**
- 4 You can view status, errors, or information for broadcasting messaging in the console.**

For example:

```
Broadcastmessage.cmd ABCserver, testapp, VHennings001, Please log off the application for routine maintenance.
```

Viewing Usage Statistics

You can determine which Planning users are logged on to the current application, and how long since they accessed the application. Users are not listed if they log on through other applications such as Financial Reporting, Smart View, or third-party reporting tools.

You can view the percentage of supporting detail detection cache being used, to determine whether an appropriate amount of RAM is allocated. If the number is very low or high, consider allocating less or more RAM. A default value of 20 is stored as a Planning property (see [“Allocating Memory for Supporting Detail Cache” on page 74](#)).

- To view usage statistics:
 - 1 From Planning Web, log on to an application.**
 - 2 Select Administration, then Statistics.**Usage statistics display for each server.

Creating and Refreshing Application Databases

On the Manage Database page, you can create and refresh application databases, which are used to store data in Essbase for each plan type in the application. Databases are structured according to dimensions, hierarchical members, attributes, and other data specified in applications. Essbase creates an encrypted data file (Essbase.sec) to store access permission information.

When you create an application, select Create to update the Essbase multidimensional databases that store application data. While building your outline, you can transfer database changes and access permissions separately to improve performance and make changes quickly available to

users. When the outline is complete, Oracle recommends that you include access permissions when refreshing database information.

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

During refresh:

- Essbase security filters are updated
- Currency conversion calc scripts are updated
- Accounts or associated properties are propagated from the relational database to the Essbase database
- Custom attributes are added, modified, or deleted in the Essbase database
- Exchange rate values are repopulated in the Essbase outline
- Member formulas for certain accounts are generated or updated
- Additions or changes to alias tables and their association to dimensions or members are updated
- The Essbase database is restructured
- UDAs are added to the Essbase database

Caution! Oracle recommends backing up the application before creating or refreshing. See [“Backing Up Applications and Application Databases” on page 74](#). Following these steps affects data in the database. When you click Create, data is erased and Planning plan types are rebuilt. When you click Refresh, data might be replaced. For important information, see [“Considerations for Working with Essbase” on page 28](#).

When an administrator uses Manage Database, all other tasks are unavailable to other users, including the application owner. All users must be logged off from the Planning application before Essbase databases can be refreshed. Oracle recommends that administrators send a broadcast message to all users, asking them to save their work and close the application before the Essbase application is refreshed. Planning does not log users out during refresh. See [“Limiting Use of Applications During Maintenance” on page 168](#).

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

- To create or refresh the application database:
 - 1 **Back up the application.** See [“Backing Up Applications and Application Databases” on page 74](#).
 - 2 **Select Administration, then Manage Database.**
 - 3 **Select options:**

- **Database:** Creates or refreshes an Essbase database for the application.
 - **Security Filters:** Generates security filters in Essbase for use by third-party applications. Stores access permissions in an encrypted data file (`Essbase.sec`). To generate security filters for all users in the application, select **Security Filters** but do not select **Validate Limit**. To generate security filters for selected users, see [“Managing Security Filters” on page 58](#).
 - **Shared Members:** Applies access permissions based on access to shared members and base members. Clear this option if you apply access permissions based only on access to base members.
 - **Validate Limit:** Identifies security filters that exceed the Essbase security filter limit of 64 KB per row. This validates filter size to ensure it does not exceed the size limit before building Essbase security filters.
- 4 **To create or refresh data in the Essbase database, select an option:**
 - Click **Create**.
 - Click **Refresh**.
 - 5 **Review the confirmation message. To continue, click **Create** or **Refresh**. After the update completes, click **Finish**.**
 - 6 **Optional:** If the create or refresh process takes some time, you can click **Run in Background** to run the process in the background without displaying the status. To view the results of running create or refresh in the background, view the Planning logs. For information about logs, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.

Managing Exchange Rates

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

To use exchange rates, you must select Multiple Currencies when creating an application. (Use Performance Management Architect to specify the base currency, currency code and symbol, triangulation currency, reporting currency, and exchange rate type. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.) You can set up exchange rates using the procedure in [“Creating Exchange Rate Tables” on page 70](#) and [“Editing Exchange Rate Tables” on page 70](#).

Multiple-currency applications store exchange rates with the HSP_Rates dimension, which includes these members and others that store currency rates:

- `HSP_InputValue`: Stores data values
- `HSP_InputCurrency`: Stores currency types for data values

When generating reports or loading data, you must refer to the HSP_InputValue member. When loading data, you must load data against the local currency. You need not refer to the HSP_InputCurrency member. By default, the HSP_Rates dimension is set to Dense. You can change this in Performance Management Architect. Planning supports currency conversion by triangulation through the triangulation currency set in Performance Management Architect.

Creating Exchange Rate Tables

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

- To create exchange rate tables:
 - 1 Select **Administration**, then **Manage Exchange Rates**.
 - 2 Click **Create**.
 - 3 Specify information for the Exchange Rate table:
 - Name
 - Description
 - 4 Click **Save**, then define settings for the table as specified in [“Editing Exchange Rate Tables” on page 70](#).

Editing Exchange Rate Tables

The default currency and triangulation currencies are available as destination currencies. You can enter exchange rates from source currencies to the default or triangulation currencies. You enter conversion values between the default currency and all the currencies defined in the Exchange Rates page. Exchange rate tables span all application time periods, so you can apply exchange rates to all scenarios. When creating or modifying exchange rate tables, you must refresh the application to store them in the plan types.

If you modify a currency’s triangulation currency, you must re-enter exchange rates for the triangulation currency property and refresh the application to transfer and store the exchange rates. You cannot select the application’s default currency as a triangulation currency.

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

- To edit exchange rate tables:
 - 1 Select **Administration**, then **Manage Exchange Rates**, select the table to edit, then click **Edit**.
 - 2 In the **Rate Table** tab, select options:
 - Display Options
 - Average
 - Ending
 - BegBalance

- Historical
 - Method
- 3 Click **Next**.
 - 4 In the **Exchange Rate** tab, set options:

Table 25 Exchange Rate Table Options

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

Tip: After entering values, fill in the value throughout the current year or all years in the table. For example, if you enter a value for Avg in the Jan07 cell and select Fill Year, the value is spread to all the months in 2007. If you select Fill Table, the value is spread to the months for all the years included in this exchange rate table. To fill in values, enter a value for Avg or End, right-click the cell, and select Fill Year or Fill Table.

- 5 Click **Save**.


Deleting Exchange Rate Tables

- To delete exchange rate tables:
 - 1 Select **Administration**, then **Manage Currency Conversion**.
 - 2 Select the exchange rate table to delete.
 - 3 Click **Delete**.

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

Managing Currency Conversion

- ▶ To manage currency conversion:
 - 1 Select **Administration**, then **Manage Currency Conversion**.
 - 2 Click **Create**.
 - 3 In the **Create File** tab, specify information for the Currency Conversion Script file, then click **Next**.
 - Name
 - Description
 - 4 In the **Details** tab, select information for the Currency Conversion Script details:
 - Currency
 - Scenario
 - Version Type: **Bottom-up** or **Target**
 - Version

Click  to select members for fields.

Working with Currency Conversion Calc Scripts

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

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

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

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

Optimizing Application Performance

Use these methods to optimize application performance:

- Assign dimensions as dense or sparse. (Select Administration, then Dimensions, and click Performance Density. Update the Density setting.)
- Change the order of dimensions from most to least dense. (Select Administration, then Dimensions, and click Evaluation Order. Select the plan type, and move available dimensions to selected dimensions. Select dimensions, and click the arrows to set the order.)
- Allocate memory for supporting detail cache.
- Clear options when creating or refreshing application databases.





Assigning Dense and Sparse Dimensions

You can assign dimensions as dense or sparse in Performance Management Architect. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*. This can accelerate data retrieval and minimize memory and disk space requirements. See [“About Sparse and Dense Dimensions” on page 188](#).



Setting the Evaluation Order

► To set evaluation order:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Evaluation Order**.
- 3 Select the plan type and click **Go**.
- 4 From **Available Dimensions**, select dimensions and move them to **Selected Dimensions**:

-  moves selected dimensions
-  moves all dimensions
-  removes selected dimensions
-  removes all dimensions

While you can select multiple dimensions per plan type, for ease of use, Oracle recommends selecting one dimension per plan type.

- 5 For multiple dimensions, set the order of precedence by clicking  or .
- 6 Click **Save**.

- 7 Set which data forms use the dimensions: [“Setting Other Options for Data Forms” on page 125.](#)

About Reordering Dimensions

The order of dimensions is critical for the structure and performance of Essbase databases. Optimize performance by ordering dimensions according to these guidelines:

- Make Period and Account dense, and order dense dimensions from most to least dense. The most dense is usually Period, followed by Account. Dense dimensions calculate faster than sparse dimensions.
- Separate sparse dimensions into aggregating and non-aggregating dimensions. Place aggregating dimensions before non-aggregating dimensions. Order sparse dimensions from most to least dense. Aggregating dimensions, such as Account, aggregate, or consolidate, children into the parent to create new data. Non-aggregating dimensions, such as Scenario, do not consolidate children to create data.

You can use Performance Management Architect to change the order for calculating dimensions. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide*.

Allocating Memory for Supporting Detail Cache

You can specify the amount of memory for the supporting detail cache. This improves performance when users change the planning unit status. To change the default memory allocation for supporting detail cache, change the SUPPORTING_DETAIL_CACHE_SIZE property. The default is 20. See [“Setting Planning Properties” on page 30.](#)

To view supporting detail cache usage, see [“Viewing Usage Statistics” on page 67.](#)

Clearing Options when Creating or Refreshing an Application Database

If data forms have a large number of members, performance may be slow when creating or refreshing applications or opening data forms. You can improve performance by clearing the Shared Members option in the Manage Database page.

Backing Up Applications and Application Databases

Back up your applications and application databases on a daily basis. Also back up before:

- Refreshing applications
- Moving applications to another server
- Upgrading applications
- Key planning milestones

Backing up applications and their related application databases consists of:

- Backing up the application in Essbase
- Backing up the relational database for Planning and Financial Reporting
- Backing up required components of Planning

For detailed instructions, see the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.

5

Loading Data and Metadata

In This Chapter

Overview of Loading Information	77
Working with the Outline Load Utility.....	78
Loading Information with Data Integrator Adapter for Planning	104
Loading Information with DIM Adapter for Planning	105
Enabling Data Load of Essbase Data	105
Loading Information with Performance Management Architect.....	106
Loading Information with FDM and ERP Integrator	106

Overview of Loading Information

You can use these tools to load metadata and data.

Table 26 Metadata Load Tools

Metadata Load Tool	For Classic Planning	For Performance Management Architect	Comments
Outline Load utility	X		See “Working with the Outline Load Utility” on page 78.
Oracle Data Integrator (ODI) Adapter for Planning	X		See the <i>Hyperion Data Integration Management Adapter for Planning User's Guide</i> .
Data Integration Management (DIM) Adapter for Planning	X		See the <i>Oracle Data Integrator Adapter for Planning Online Help</i> .
Performance Management Architect flat files		X	See the <i>Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide</i> .
Performance Management Architect interface tables		X	See the <i>Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide</i> .

Metadata Load Tool	For Classic Planning	For Performance Management Architect	Comments
Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications	X	X	See the <i>Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide</i> .

Table 27 Data Load Tools

Data Load Tool	For Classic Planning	For Performance Management Architect	Comments
Outline Load utility	X		Loads numeric, date, and text data values. See “Working with the Outline Load Utility” on page 78.
Performance Management Architect data synchronization		X	Loads data for Planning. See the <i>Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide</i> .
Essbase flat file load	X	X	Loads numeric data for Planning. See the <i>Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide</i> .
Oracle Hyperion Financial Data Quality Management Adapter for Planning, Fusion Edition (FDM)	X	X	Loads numeric data for Planning. See the <i>Hyperion Financial Data Quality Management Administrator's Guide</i> .
ERP Integrator	X	X	See the <i>Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide</i> .

Working with the Outline Load Utility

The Outline Load utility can be used to load metadata and data for Account, Period, Year, Scenario, Version, Currency, Entity, user-defined dimensions, attributes, UDAs, and exchange rates. The utility cannot be used to create Smart Lists or Smart List members.

Use these general steps to load information with the Outline Load utility:

1. When loading data, set the `DIRECT_DATA_LOAD` and `DATA_LOAD_FILE_PATH` system properties.
2. Create a load file for each dimension or set of data that you want to load.
3. Test the load files, and then run the utility.

For detailed instructions, see these topics:

- [“Loading Data” on page 79](#) and [“Loading Metadata” on page 82](#)
- [“Command Line Parameters for the Outline Load Utility” on page 83](#)
- [“Dimension Properties” on page 89](#)

Loading Data

When loading data with the Outline Load utility, there are two ways to specify driver members. You can load to driver members that are specified on the Planning Data Load Administration page, or you can specify driver members in the .CSV load file and run the utility with the `/TR` option.

If you load data with the `/TR` option, the .CSV file must list the driver member and all other members under the Point-of-View column, regardless of their location on the data form. For example, if Jan or Descendants (YearTotal) are columns in a data form, they must be specified in the Point-of-View column. When using `/TR`, you can load one value per row in the .CSV file. You can include multiple rows, but you can specify only one data value per row.

Caution! Following these steps can affect data in the database. The Planning `DIRECT_DATA_LOAD` system property enables data to be loaded directly to Essbase. This default setting is for data to be loaded directly into Essbase. To prevent data from being loaded directly into Essbase, set `DIRECT_DATA_LOAD` to false.

► To load data with the Outline Load utility:

- 1 **Back up the application and application databases before loading information.** See the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.
- 2 **Set Planning System properties.**
 - a. Log in to the Planning application.
 - b. Select **Administration**, then **Manage Properties**, then click the **System Properties** tab.
 - c. Set the `DIRECT_DATA_LOAD` and `DATA_LOAD_FILE_PATH` properties:
 - If `DIRECT_DATA_LOAD` is set to `True`, or if you do not specify a value for this property, information is loaded directly into Essbase while the load file records are processed. For this method to work correctly, the outlines maintained in Planning and Essbase must be synchronized. The .CSV load file must not specify any Planning outline changes unless they have already been refreshed to Essbase.
 - If `DIRECT_DATA_LOAD` is set to `False`, the Outline Load utility processes the .CSV load file that you created to generate a data file (.TXT) and rule file (.RUL). This way,

the Planning and Essbase outlines do not need to be synchronized because data is not loaded at this time. You can refresh the changes at a convenient time to propagate the metadata changes to Essbase, and then load data directly into Essbase (for example, using Administration Services).

In most cases, set `DIRECT_DATA_LOAD` to `False`, and set `DATA_LOAD_PATH` to the location and name that will be used for the generated data and rules files, for example, `C:/myDirectory/App1.txt`. Ensure that these properties are set in the System Properties tab in the Manage Properties page.

- d. Restart the Planning application server.
- 3 If you want to load to driver members that are specified in Planning, set the driver members as described in this step. Otherwise, skip to the next step.**
- a. Log on to the Planning application for which data will be loaded.
 - b. Select **Administration**, then **Manage Data Load**.
 - c. Select a dimension from the **Available Data Load Dimensions** list (such as **Account**). This is the dimension for which you want to load data. For example, it may appear as a row in a Planning data form.
 - d. Select a dimension from the **Available Driver Dimensions** list (such as **Period**).
 - e. Click the member selection icon to select members of the **Driver Dimension** that will appear in the Selected Members list (such as **Jan, Feb, March**). For example, these members may appear as columns in a Planning data form.
- 4 Generate a comma-separated load file containing these columns:**
- **Driver Member:** The member into which data is loaded. You can have one driver dimension per load. Multiple members can be defined for the driver dimension. The value is passed as a string representing a numeric value, or, if a Smart List is bound to the member, as a Smart List value.
 - **Point-of-View:** All other dimensions required to determine the intersection for which to load the data. (If you are using `/TR`, include all of the members except the driver member.) The data load automatically performs cross-product record creations based on the dimension parameters in the point of view (POV). The load file creates and loads the data record for each relevant cell intersection. The value is passed as a string. The POV accepts a comma-separated list of members, including member functions. For example, `children(Q1)` is expanded to `Jan, Feb, Mar` during the load. The corresponding records are generated based on the cross product of all member combinations and the data value.
 - **Data Load Cube Name:** The name of the plan type to which data is being loaded. The value is passed as a string. Values include any plan types specified in the application, such as `Plan1`.

Example 1: In this example, `Account` was selected as the Data Load dimension on the application's Manage Data Load page. `Period` was selected as the Driver Dimension, and `Jan` was selected as the Driver member.

```
Account,Jan,Point-of-View,Data Load Cube Name  
acct1,12,"Local,ent1,Current,Ver1,FY08",Plan1
```


Example 2: In this example, Entity was selected as the Data Load dimension on the application's Manage Data Load page. Account was selected as the Driver Dimension, and Account members aUnspec, aSmart, aDate, and aText were selected as the driver members. This .CSV load file loads data into the intersection of e1, the point of view, and the Account driver members, aUnspec, aSmart, aDate, and aText.

```
Entity,Operation,Data Load Cube Name,aUnspec,aSmart,aDate,aText,Point-of-View
```

```
e1, ,Plan1,77,smart1,12-22-2008,textValue,"USD,Jan,Current, BUVersion_1,FY07"
```

Assuming these values for the driver members:

- aUnspec: Data Type Unspecified (numeric), value 77
- aSmart: Data Type Smartlist,value smartlist entry 'smart1'
- aDate: Data Type Date,value 12-22-2008
- aText: Data Type Text,value 'textValue'

If DIRECT_DATA_LOAD is set to False, the example would generate this data load file:

```
Currency Version Scenario Year Entity Period Account HSP_Rates
77 USD BUVersion_1 Current FY07 e1 Jan aUnspec HSP_InputValue
1 USD BUVersion_1 Current FY07 e1 Jan aText HSP_InputValue
20081222 USD BUVersion_1 Current FY07 e1 Jan Date HSP_InputValue
1 USD BUVersion_1 Current FY07 e1 Jan aSmart HSP_InputValue
```

Example 3: Specify drivers directly in the .CSV load file.

```
Value,Driver Member,Point-of-View,Data Load Cube Name
```

```
14,a1,"Jan,Local,e1,Current,Version1,FY08",Plan1
```

```
s11_value2,a2,"Jan,Local,e1,Current,Version1,FY08",Plan1
```

```
OutlineLoad /A:acpt1 /U:admin /M /I:c:\outline1data.csv /TR /L:c:/OutlineLogs/outlineLoad.log /X:c:/OutlineLogs/outlineLoad.exc
```

5 Test the load file and run the utility.

- a. Locate the utility, installed by default in *HYPERION_HOME*/products/Planning/bin.
- b. To confirm that the load file parses without any errors, run the utility using the /N parameter, and check the outline log file to be sure no error messages were generated. Running the utility with /N does not load data or metadata, but ensures that the .CSV load file parses successfully. For example, you could use this command line to check the load file for a Planning application called test:

```
C:\Hyperion\products\Planning\bin>OutlineLoad /A:test /U:admin /M /N /I:c:\outline1data3.csv /D:Entity /L:c:/outlineLoad.log /X:c:/outlineLoad.exc
```

- c. You can then run the utility from the command prompt without /N, using the case-sensitive command, one space, and the appropriate parameters. For example:

```
C:\Hyperion\products\Planning\bin>OutlineLoad /A:test /U:admin /M /
I:c:\outline1data3.csv /D:Entity /L:c:/outlineLoad.log /X:c:/
outlineLoad.exe
```

If you are loading data without specifying driver members within Planning, you can run the utility including /TR. For example:

```
C:\Hyperion\products\Planning\bin>OutlineLoad /A:test /U:admin /M /
N /I:c:\outline1data3.csv /TR /D:Entity /L:c:/outlineLoad.log /
X:c:/outlineLoad.exe
```

For detailed information on the parameters available for use with the Outline Load utility, see [“Dimension Properties” on page 89](#).

Loading Metadata

Metadata for Classic Planning applications can be loaded for Account, Period, Year, Scenario, Version, Currency, Entity, user-defined dimensions, attributes, and UDAs. Values can also be loaded for exchange rates. However, because exchange rate values are loaded into the Planning relational tables, not directly into Essbase, the procedure for loading metadata still applies.

The utility loads one record at a time. If a record fails to load, a message is written to the exception file, and the load process resumes with the next record. When new members are added, unspecified properties assume the default value or inherit the parent member's property as appropriate. If a member exists and no property value is specified, the property is left as is.

► To load metadata:

1 Generate the load file.

When loading metadata, the load file must contain a header record that lists the dimension, such as Account, and the member properties used by subsequent metadata records. For example, for Account, you can specify which account to load, a default alias, the operation to perform, and so on. Header records are case sensitive. They can appear in any order.

The next lines in the .CSV load file contain metadata records, listed in the order designated by the header record. Each metadata record contains a comma-separated list of property values that correspond to the header record entries. For detailed information on the properties available for each Planning member, see [“Dimension Properties” on page 89](#).

Example: This load file loads an Entity dimension with the required header record and three data records. The header record specifies the member to be loaded (Entity), the parent member (Parent) into which to load the member, and the Data Storage property to assign to the member.

```
Entity, Parent, Data Storage
e1, Entity,
e2, ,
e1, e2, Shared
```

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

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

The first data record (e1, Entity) loads Entity member e1 as a child under the root member Entity. Unspecified values assume the default. For example, if data storage is not specified, it assumes the default value, Never Share. The next data record (e2,) loads 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) loads a shared member of e1 under member e2, and sets data storage to Shared.

When loading the Currency dimension with the Default Currency Symbol, if a currency is added without a symbol specified, the symbol is set to that of a pre-defined currency of the same name (or, if the name does not match a pre-defined currency, to the currency code of the currency being added).

2 Test the load file and run the utility.

- a. Locate the Outline Load utility, installed by default in `HYPERION_HOME/products/Planning/bin`.
- b. To confirm that the load file parses without any errors, run the utility using `/N`, and check the log file to be sure no error messages were generated. For example:

```
C:\Hyperion\products\Planning\bin>OutlineLoad /A:test /U:admin /M /
N /I:c:\outline1_ent.csv /D:Entity /L:c:/outlineLoad.log /X:c:/
outlineLoad.exc
```

- c. You can then run the utility from the command prompt, using the case-sensitive command, one space, and the appropriate parameters.

For detailed information on the parameters available for the Outline Load utility, see [“Dimension Properties” on page 89](#).

Command Line Parameters for the Outline Load Utility

The following command line parameters are available for the Outline Load utility. After running the utility, you can verify the results by reviewing the exception file and log file. If no errors are reported in the log file, you can then access the imported metadata and data in the application. It is not necessary to restart the application server.

```
OutlineLoad [-f:passwordFile] [/S:server] /A:application /U:userName [/
M] [/I:inputFileName/D[U]:loadDimensionName|/
DA:attributeDimensionName:baseDimensionName] [/TR] [/N] [[/R] [/U]] [/
C] [/F] [/K] [/X:exceptionFileName] [L:logFileName] [/?]
```

Parameter	Description
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the

Parameter	Description
	command line to read the password from the full file path and name specified in <i>passwordFile</i> . See “Suppressing Password Prompts in Planning Utilities” on page 37 .
<i>/S:server</i>	Server on which the application resides; if not specified, localhost is used.
<i>/A:application</i>	Name of the Planning application to which you are importing.
<i>/U:userName</i>	User name with which to log on to the application.
<i>/M</i>	Generate fully qualified header records for loadable dimensions in the application. Use <i>/-M</i> if you do not want to display this information (default).
<i>/I:inputFileName</i>	Name of the load file that contains a header record and data records in .CSV format. You must also specify a data load dimension (<i>/D</i>).
<i>/D:loadDimensionName</i>	Dimension to be loaded, whose member fields correspond to the header record in the load file. You must also specify a load file (<i>/I</i>). See the following rows to load user-defined dimensions and attributes using <i>/DU</i> , <i>/DA[T]</i> , <i>/DAN</i> , <i>/DAB</i> , and <i>/DAD</i> .
<i>/DU:userDefinedLoadDimensionName</i>	User-defined dimension to be loaded; a dimension with this name will be created if it does not exist.
<i>/DA[T]:attributeLoadDimensionName:baseDimensionName</i>	Text attribute dimension to be loaded; an attribute dimension with this name, bound to the base dimension, will be created if it does not exist.
<i>/DAN:attributeLoadDimensionName:baseDimensionName</i>	Numeric attribute dimension to be loaded; an attribute dimension with this name, bound to the base dimension, will be created if it does not exist.
<i>/DAB:attributeLoadDimensionName:baseDimensionName</i>	Boolean attribute dimension to be loaded; an attribute dimension with this name, bound to the base dimension, will be created if it does not exist.
<i>/DAD:attributeLoadDimensionName:baseDimensionName</i>	Date attribute dimension to be loaded; an attribute dimension with this name, bound to the base dimension, will be created if it does not exist.

Parameter	Description
/TR	Load data when driver members are specified in the .CSV file in the Driver Members column. All members except the driver member must be specified in the Point-of-View column. With /TR, you can load one value per row in the .CSV file.
/N	Perform a “dry run” by parsing the load file without loading data or metadata. Use /-N (or do not specify the /N parameter) to parse the load file while loading data and metadata (default).
/TR	Load data when driver members are specified in the .CSV file in the Driver Members column. All members except the driver member must be specified in the Point-of-View column. With /TR, you can load one value per row in the .CSV file.
/O	Maintain the order of members in the load file when loading, with the exception of UDAs (default). Use /-O to ignore the order of members in the load file when loading.
/H	Order input records in parent-child order, with the exception of UDAs (default). Use /-H to load input records as they appear in the load file; this option is faster and uses less memory.
/R	Delete all members of the load dimension before performing the load. Use /-R (or do not specify the /R parameter) to keep all members of the load dimension (default). See also /U. Note: Use caution with /R; this option removes attribute bindings and process management states.
/U	Delete all planning units with the /R option, or display an error if members in planning units would be deleted. Use /-U (or do not specify the /U parameter) to prevent deleting members in planning units (default). Use /U with /R to enable deleting started planning units and deleting all members in the dimension specified in the .CSV load file.
/T	Inherit unspecified plan type settings from the parent when adding new

Parameter	Description
	members (default). Use <code>/-T</code> to force explicit setting of plan type settings for the member.
<code>/C</code>	Perform a cube refresh after the metadata load. Use <code>/-C</code> if you do not want to perform a cube refresh (default). See also <code>/F</code> .
<code>/F</code>	Create security filters when refreshing with the <code>/C</code> option. Use <code>/-F</code> if you do not want to refresh security filters (default). (This option does not provision users to the application; it only creates security filters for users that currently exist. Users can be provisioned to applications using other methods.) For this option to take effect, <code>/C</code> must also be specified.
<code>/K</code>	Lock the load dimension before loading (default), recommended. Use <code>/-K</code> if you do not want to lock the dimension (not recommended unless you are using <code>/N</code>).
<code>/X:exceptionFileName</code>	Specify the file that will contain exceptions that occur during the load. (If no file name is specified, the information is written to a file called <code>stderr</code> .)
<code>/L:logFileName</code>	Specify the file that will contain status and informational messages. (If no file name is specified, the information is written to a file called <code>stdout</code> .)
<code>/DX:HSP_Rates</code>	Load the HSP_Rates dimension and create exchange rate tables if they do not exist.
<code>/?</code>	Display usage text.

Example: Load numeric attribute dimension and values, and associate them with the Entity dimension. (An attribute dimension will be created if it does not exist, but no assignment is made of attribute values to base numbers.)

```
OutlineLoad /A:Test /U:admin /M /I:c:/outline1_attribvals_text.csv /
DAN:NumericAttrib:Entity /L:c:/outlineLoad.log /X:c:/outlineLoad.exc
```

```
NumericAttrib,Parent
```

```
One,NumericAttrib
```

```
1,One
```

```
2,NumericAttrib
```

Example: Load Exchange Rates, add EUR as a member of the Currency dimension, and change the year in the .CSV file to match an existing year in the Planning application. The Exchange Rate table is created in the Planning application if it does not exist.

```
OutlineLoad /A:Test /U:admin /M /I:c:/outline1_rates.csv /DX:HSP_Rates /
L:c:/OutlineLogs/outlineLoad.log /X:c:/OutlineLogs/outlineLoad.exc
Table, To Currency, From Currency, Method, Historical, Beg Balance, Year,
Period, Average, Ending
```

```
FX1 , USD, EUR, multiply, 1, 2, FY08, Jan, 3, 4
```

```
FX1 , USD, EUR, , , , FY09, Feb, 5, 6
```

Example: Set Weekly Distribution to Use 445

```
Account, Parent, Use 445
```

```
all,a1,1
```

Example: Load a .CSV file that contains all of the properties available for a UDA. The UDA is loaded and associated with a dimension, but it is not assigned to any member in the dimension.

```
OutlineLoad /A:Test /U:admin /M /I:c:/outline1_uda.csv /D:UDA /L:c:/
OutlineLogs/outlineLoad.log /X:c:/OutlineLogs/outlineLoad.exc
```

```
Dimension,UDA
```

```
Account,New2
```

Example: Load a .CSV file for Currency that does not specify the currency symbol. In this case, the symbol for this currency in the Planning application is set to the ISO symbol, EUR. The scale defaults to 1.

```
Currency, Parent, Symbol, Scale
```

```
EUR, , ,
```

Example: Load a .CSV file for Currency that sets the symbol to the name of the new currency. The symbol is automatically set to NewCurr1 in the Planning application for currency NewCurr1. Currency names are limited to 8 characters.

```
Currency, Parent, Symbol, Scale
```

```
NewCurr1, , ,
```

Example: Use the `-f` parameter with an encrypted password

If you have generated an encrypted password file, you can use `-f` as the first parameter in the command line to run the Outline Load utility without entering a password. For example, if you used the `PasswordEncryption` utility to create a password file called `encrypt.txt`, you could use this command line:

```
OutlineLoad -f:c:\encrypt.txt /A:acct /U:admin /M /I:c:/
outline1_accounts.csv /D:Account /L:c:/OutlineLogs/outlineLoad.log /X:c:/
OutlineLogs/outlineLoad.exc
```

Example: `/O` parameter and .CSV load file order

In the following .CSV load file, if Entity members `e1` and `e2` already exist in the Entity dimension, `e3` could be added as the last sibling, even though it is first in the load file. If `/O` is used, `e3` is

loaded as the first sibling. Because /O is the default, you must specify /O to have e3 loaded as the last sibling.

```
Entity, Parent, Data Storage, TextAttrib
e3, Entity, Store,
e2, Entity, Store,
e1, Entity, Store,
```

Example: /H parameter and parent/child order

Assume that member e1 already exists, and A and B are new members being loaded. Without /H, an error would display because member B does not exist. With /H, members are sorted internally, so B is loaded first as child of e1, and then A is loaded successfully as child of B.

```
Entity, Parent, Data Storage
A, B, Store
B, e1, Store
```

Example: /R parameter

If some members already exist in the dimension, only the members in the input load file should exist in the dimension after the load. If an error occurs during the load after the delete operation, all members of the dimension may be deleted, and the dimension may be empty. Attribute dimensions are not deleted. If a planning unit is started, no Entity members are deleted because the Entity member in the planning unit cannot be deleted.

```
Entity, Parent, Data Storage, TextAttrib
e1, Entity, Store,
e11, e1, Store, orange
e2, Entity, Store,
e21, e2, Store,
e11, e2, shared, yellow
```

Example: /T parameter

Load the Account dimension with /T to inherit plan types not explicitly specified in the load file from the parent when adding new members. Assume that member a1 already exists in the application and is valid for all three plan types. After the load completes, member a11 is valid for all three plan types, even though only Plan1 and Plan3 are specified in the load file.

```
Account, Parent, Source Plan Type, Plan Type (Plan1), Plan Type (Plan2),
Plan Type (Plan3)
a11, a1, Plan1, 1, , 1
```

Example: /-T parameter

Load the Account dimension with /-T to force explicit setting of plan types for new members. Assume that member a1 already exists in the application and is valid for all three plan types.

After the load, member a11 will be valid only for the Plan1 and Plan3 plan types specified in the load file, and not for Plan2.

Example: /TR parameter

```
OutlineLoad /A:acpt1 /U:admin /M /I:c:\outline1data.csv /TR /L:c:/
OutlineLogs/outlineLoad.log /X:c:/OutlineLogs/outlineLoad.exc
```

```
Value,Driver Member,Point-of-View,Data Load Cube Name
14,a1,"Jan,Local,e1,Current,Version1,FY08",Plan1
s11_value2,a2,"Jan,Local,e1,Current,Version1,FY08",Plan1
```

Dimension Properties

See the following sections for common member properties, and properties specific to Account, Entity, Periods, user-defined dimensions, Year, Scenario, Version, Currency, attribute dimensions, UDAs, and exchange rates.

Common Member Properties

Member properties common to several dimensions are described in this section. For properties specific to certain dimensions, see the following sections.

- **Parent:** The parent of the member being loaded to create the dimension hierarchy. When you load a member and specify a parent member that is different than the parent member in the application, the member is updated with the new parent value you specified. For example, if Member 1 has a parent value of Member A in your Planning application and you load Member 1 with a parent value of Member B, the system updates your application, making Member B the parent of Member 1. Member 1 and its descendants are moved from Member A to Member B. If you do not specify Parent, it is ignored during the load. The record is rejected if the specified parent is a descendant of the member being loaded, or does not exist in the application.
- **Alias: Default:** Alias defined for the member in the Default Alias table. If you do not specify a value, the alias is not changed in the application. If you specify <none> as the value, the alias in the application is deleted.
- **Valid For Consolidations:** Not used by Planning.
- **Data Storage:** The storage attribute for the member being loaded. This value is passed as a string. Default: Never Share. Valid values:
 - Store
 - Dynamic Calc
 - Dynamic Calc and Store
 - Shared
 - Never Share
 - Label Only

- **Two Pass Calculation:** A Boolean value to indicate whether this attribute is associated with the member being loaded. Use 0 for False and any other number for True. Default: False. For Account members, the Two Pass Calculation property can be set regardless of the Data Storage setting. For members of dimensions other than Account, the Two Pass Calculation property is valid only when the Data Storage value is Dynamic Calc or Dynamic Calc and Store. Otherwise, the record is rejected.
- **Description:** Description for the member being loaded. If you do not enter a value, new members are loaded without descriptions, and descriptions of existing members are unchanged. If you use <none> as the value, any existing description for that member is deleted.
- **Formula:** Specifies a member formula for the member. By default, there is no member formula associated with a dimension or member. You cannot load member formulas for members that are Shared or Label Only.
- **UDA:** Specifies the value of the user-defined attributes to bind to the member. Undefined UDAs are added to the dimension. You can add UDAs only to those dimensions that are already created in Planning.
- **Smart List:** Takes the name of a user-defined Smart List that is defined in the application. This value is passed as a string. The default for Smart List is <none>. Only one Smart List can be associated with a member.
- **Data Type:** The data storage value. Valid values:
 - **Currency:** Stores and displays the member value in the default currency.
 - **Non-currency:** Stores and displays the member value as a numeric value.
 - **Percentage:** Stores values as numeric values and displays the member value as a percentage.
 - **Smart List:** Stores values as numeric values and displays the member value as a string.
 - **Date:** Stores and displays the member value in the format mm/dd/yyyy or dd/mm/yyyy.
 - **Text:** Stores and displays the member's value as text.
 - **Unspecified:** Stores and displays the member value as Unspecified.
- **Operation:** Takes these values:
 - **Update:** Adds, updates, or moves the member being loaded.
 - **Delete Level 0:** Deletes the member being loaded if it has no children.
 - **Delete Idescendants:** Deletes the member being loaded and all of its descendants.
 - **Delete Descendants:** Deletes the descendants of the member being loaded, but does not delete the member itself.

Use caution when deleting members; this deletes the member, its data, and any associated planning units.
- **Process Management Enabled:** Enable for process management. By default, True.
- **Plan Type (for example, Plan1, Plan2, Plan3):** A Boolean value that indicates whether the member being loaded is used in the specified plan. Valid values: 0 for False, or any other

number for True. Default: True. The name depends on the name of the plan type in the application.

- **Aggregation** (Plan1, Plan2, Plan3): The aggregation option for the member being loaded, as related to the specified plan. This is available only if the application is valid for this plan type. This value is passed as a string. Valid values:
 - + (Addition)
 - - (Subtraction)
 - * (Multiplication)
 - / (Division)
 - % (Percent)
 - ~ (Ignore during consolidation)
 - Never (Do not aggregate, regardless of hierarchy)
- **UDA**: The value of the UDA being loaded. You can associate UDAs only with dimensions that exist in the application. If a UDA exists, its properties are modified; otherwise, the record is added.

Account Dimension Properties

Account load files can include these properties:

Account, Parent, Alias: Default, Alias: T1, Valid For Consolidations, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Account Type, Time Balance, Use 445, Use 544, Use 554, Skip Value, Exchange Rate Type, Variance Reporting, Source Plan Type, Plan Type (*Plan1*), Aggregation (*Plan1*), Plan Type (*Plan2*), Aggregation (*Plan2*), Plan Type (*Plan3*), Aggregation (*Plan3*), *AttribDim1*, *AttribDim2*

For details on these properties, see the following table and [“Common Member Properties” on page 89](#).

Notes:

- Time Balance specifies how account data flows over time. It takes a type only for members with an account type of Saved Assumption, or if the record is rejected.
- When Time Balance is Flow, records with any valid skip values are loaded, but Skip Value is disabled for all Account types.
- Skip Value can be set when Time Balance is First, Balance, or Average. These options set which values to skip when the parent value is calculated: none, #MISSING, zeros, or #MISSING and zeros.
- Plan type names and numbers depend on what is defined in the application. Base time periods cannot be added with the utility. Year, base time periods, and exchange rates cannot be deleted.
- YearTotal and BegBalance time periods cannot be modified with the utility.

- Exchange Rate Type depends on the value specified for Data Type. Valid values: Average, Ending, and Historical when Data Type is Currency, or None when Data Type is anything other than Currency.
- Variance Reporting loads account members with an account type of Saved Assumption or if the record is rejected. Expense designates the saved assumption as an expense. The actual amount is subtracted from the budgeted amount to determine the variance. Non-Expense designates the Account as not an Expense. The budgeted amount is subtracted from the actual amount to determine the variance. Values for Account types: Revenue: Non-Expense, Expense: Expense, Asset: Non-Expense, Liability: Non-Expense, Equity: Non-Expense.
- When you update or save the parent of a member, the system verifies if the Source Plan Type associated with the member being loaded is valid for the new parent. If the source plan type of a member is valid for the parent member but not for the member itself, the member is saved, but its source plan type is set to the first valid plan type. If a Source Plan Type is specified but is not valid for the parent, the record is rejected.

Table 28 Account Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Account	Member Name	Text, subject to member naming restrictions	None	Yes
Parent	Member Name	Text, subject to member naming restrictions.	None; the name of an existing member, or, if empty, the member is placed as a child under the dimension root	No (Need to specify for most Period members)
Alias:Alias_Table_Name	Alias	Text, subject to Planning member naming restrictions and aliases already defined in the alias table specified in the column header; <none> removes any alias binding for the member from the specified table	None	No One column header for each alias table defined on the dimension is displayed. Default is a pre-defined alias table defined for every dimension
Valid For Consolidations	Not Used	N/A	N/A	No
Data Storage	Data Storage	Text: Store, Dynamic Calc and Store, Dynamic Calc, Never Share, Shared,Label Only	Inherited from the parent. If the parent is the root member, the default is Never Share	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Two Pass Calculation	Two Pass Calculation	True, False, or an integer: non-zero is true; zero is false	Inherited from the parent	No
Description	Description	Text, subject to Planning maximum character restrictions; <none> removes a description	None	No
Formula	Formula	Text, subject to Essbase restrictions for valid formula syntax; <none> removes a formula	None	No
UDA	UDA	A single UDA text value, or a quoted, comma-separated list of UDA text values; non-existing UDAs are added to the dimension; existing UDA bindings are removed on subsequent saves if not re-specified; nothing specified leaves bindings as is; <none> removes all existing UDA bindings	None	No
Smart List	Smart Lists	Text, restricted to names of Smart Lists that are already defined for the application. Should be specified only if Data Type is set to Smart List; <none> removes any existing Smart List binding	None	No
Data Type	Data Type	Text: Unspecified, Currency, Non-Currency, Percentage, Smart List, Date, Text	Inherited from the Parent; Currency if the member is added under the root dimension	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Operation		Text: update; delete level 0; delete idescendants; delete descendants	Update	No
Account Type	Account Type	Text: Expense, Revenue, Asset, Liability, Equity, Saved Assumption	Inherited from the parent; Revenue if the member is added under the root dimension	No
Time Balance	Time Balance	Text: Flow, First, Balance, Average, avg_ actual, avg_365, fill	Inherited from the parent (default Time Balance values for Account types: Expense: FlowRevenue: Flow, , Asset: Balance, Liability: Balance, Equity: Balance)	No
Use 445 Use 544 Use 554 (Only one column header appears, depending on system settings. If weekly distribution is not defined for the application, no column header displays.)	Weekly Distribution (If Weekly Distribution is set to Even for the application, no distribution option displays.)	True, False, or an integer: nonzero is true; zero is false	None	No
Skip Value		Text: None, Missing, Zeros, Missing and Zeros; must be None if Account Type is Expense or Revenue	Inherited from the parent	No
Exchange Rate Type	Exchange Rate Type	Text: None, Average, Ending, Historical (None should not be specified if Data Type is set to Currency; otherwise, it should be specified)	Inherited from the parent; Average if member is added under root dimension	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Variance Reporting	Variance Reporting	Text: Non-Expense, Expense. (Must be Expense if Account Type is Expense; must be Non-Expense for all other Account types.	Inherited from the parent; Non-Expense if the member is added under the root dimension	No
Source Plan Type	Source Plan Type	Text: Plan Type names defined in the application (for example, Plan1 or Plan2)	Plan1 or the name of the first plan type defined in the application	No
Plan Type (<i>Plan1</i>)	Plan Type	True, False, or an integer: non-zero is true; zero is false	Inherited from the parent	No
Aggregation (<i>Plan1</i>)	Aggregation	Text: +, -, *, /, %, ~, Never	Inherited from the parent. If the parent is a root member, the default is +; for Year, the default is ~ (ignore)	No
Plan Type (<i>Plan2</i>)	Plan Type	True, False, or an integer: non-zero is true; zero is false	Inherited from the parent	No
Aggregation (<i>Plan2</i>)	Aggregation	Text: Text: +, -, */%, ~, Never	Inherited from the parent; if the parent is the root member, the default is +; for Year, the default is ~ (ignore)	No
Plan Type (<i>Plan3</i>)	Plan Type	True, False, or an integer: non-zero is true; zero is false	Inherited from the parent	No
Aggregation (<i>Plan3</i>)	Aggregation	Text: +, -*/%, ~, Never	Inherited from the parent; if the parent is a root member, the default is +; for Year, the default is ~ (ignore)	No
Attribute Dimension Name (this property is available for sparse Account, Entity, or user-defined dimensions)	Attributes	The name of an attribute defined in the attribute dimension: existing attribute bindings are removed on subsequent saves if not re-specified;	None	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
		nothing specified leaves bindings as is; <none> removes all existing attribute bindings for the member; one column header is displayed for each Attribute dimension defined on the dimension		

Entity Dimension Properties

Entity, Parent, Alias: Default, Alias: T1, Valid For Consolidations, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Base Currency, Plan Type (*Plan1*), Aggregation (*Plan1*), Plan Type (*Plan2*), Aggregation (*Plan2*), Plan Type (*Plan3*), Aggregation (*Plan3*), *AttribDim1*, *AttribDim2*

Notes:

- **Entity:** The Entity information being loaded.
- **Base Currency:** Displayed only for multicurrency applications. Takes the code of the currency for the Entity being loaded, as defined in the application.
- Plan type names and numbers depend on what is defined in the application.

Table 29 Entity Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default
Entity	Member Name	Text, subject to Planning member naming restrictions	None
Two Pass Calculation	Two Pass Calculation	True, False, or an integer: non-zero is true; zero is false (should be set to 1 only if Data Type is Dynamic Calc or Dynamic Calc and Store)	Inherited from the parent
Base Currency	Base Currency	Text, restricted to currency names already defined in the application	Inherited from the parent; if the member is shared, the default is the base member's currency; if the member is added under the root dimension, the base currency is the default currency defined

Column Header in .CSV Load File	Planning Property	Value	Default
			when the application was created

Period Dimension Properties

Period, Parent, Alias: Default, Alias: T1, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Type, Start Period, End Period, Aggregation (*Plan1*), Aggregation (*Plan2*), Aggregation (*Plan3*)

Notes:

- For Period, Parent should be specified for most update operations.
- Type (such as Base or Rollup) cannot be changed for existing members. You can add and update Rollup and Alternate members. You can update BegBalance and Base time periods (for example, to add an alias). You cannot add or modify YearTotal time period. The DTS Period type is recognized, but is not supported by the Outline Load utility. If you try to load DTS information, the utility displays an error in the log file.
- Start Period and End Period are valid for Rollup Period types.
- Plan type names and numbers depend on what is defined in the application.
- For properties common to Period and Account members, see [“Account Dimension Properties” on page 91](#).

Table 30 Period Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Period	Member Name	Text, subject to Planning restrictions on Period member names		
Type	Not exposed in Planning.	Text: base, rollup, year, alternate, DTS; only Rollup and alternate time periods can be loaded by the utility; BegBalance and Base periods can be modified (for example, to add an alias); YearTotal and DTS time periods cannot be loaded or modified	None	Yes

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Start Period	Text (only applicable to summary time periods).	Text, restricted to time periods already defined for the application	None	Yes
End Period	Text (only applicable to summary time periods).	Text, restricted to time periods already defined for the application	None	Yes

User-Defined Dimension Properties

User Defined Dimension Name, Parent, Alias: Default, Alias: T1, Valid For Consolidations, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Aggregation (Plan1), Aggregation (Plan2), Aggregation (Plan3)

Plan type names and numbers depend on what is defined in the application. For properties common to User-Defined and Account members, see [“Account Dimension Properties”](#) on page 91.

Table 31 User-Defined Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
<i>User-defined dimension name</i>	Member Name	Text, subject to Planning member naming restrictions. Name of the user-defined member being added to the dimension	None	Yes
Parent	Member Name or root Dimension Name	Text, subject to Planning member naming restrictions	Root dimension member	Yes

Year Dimension Properties

Year, Parent, Alias: Default, Alias: T1, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation

Notes:

- The member name for Year must be of the form *FYnn*.
- If the last existing year is less than the year value entered, the intervening years are also created. For example, if the last defined year is FY08 and you enter FY11, the utility creates FY09, FY10, and FY11.

Table 32 Year Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Year	Year member	FY nn	None	Yes

Scenario Dimension Properties

Scenario, Parent, Alias: Default, Alias: T1, Valid For Consolidations, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Start Year, Start Period, End Year, End Period, Exchange Table, Include BegBal, Process Management Enabled, Aggregation (*Plan1*), Aggregation (*Plan2*), Aggregation (*Plan3*)

Notes:

- No Year cannot be selected as Start Year or End Year. BegBalance cannot be selected as Start Period or EndPeriod.
- If not specified, Start Year and End Year and Start Period and End Period are set to default values (first and last year in the application, and first and last base period in the application).
- Plan type names and numbers depend on what is defined in the application.

Table 33 Scenario Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Scenario	Scenario Name	Text	None	Yes
Start Year		First FY year defined in the application (as determined by position)	First year in the application	No
Start Period		First base time period (as determined by position)	First base period in the application	No
End Year		Last FY year defined in the application (as determined by position)	Last year in the application	No
End Period		Last base time period (as determined by position)	Last base period in the application	No
Exchange Table		The name of an exchange rate table defined in the application	None	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Include BegBal		True, False, or an integer: non-zero is true; zero is false	False	No
Process Management Enabled		True, False, or an integer: non-zero is true; zero is false	False	No

Version Dimension Properties

Version, Parent, Alias: Default, Alias: T1, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Version Type, Process Management Enabled, Aggregation (*Plan1*), Aggregation (*Plan2*), Aggregation (*Plan3*)

Plan type names and numbers depend on what is defined in the application.

Table 34 Version Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Version	Version Name	Text	None	Yes
Version Type	Version Type	Bottom Up or Target	Bottom Up	No
Process Management Enabled	Enabled for Process Management	True, False, or an integer: non-zero is true; zero is false	False	No

Currency Dimension Properties

Currency, Parent, Alias: Default, Alias: T1, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Operation, Symbol, Scale, Triangulation Currency, Reporting Currency, Thousands Separator, Decimal Separator, Negative Style, Negative Color

If the Currency symbol is not specified, it is set by default to the ISO symbol if the currency being loaded is defined in Planning.

Table 35 Currency Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Currency	Currency Name	Text	None	Yes
Symbol	Symbol	Text, subject to Planning currency	ISO symbol if the currency being loaded is defined	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
		symbol naming restrictions	Planning; the same as the currency name if the currency being loaded is not defined in Planning	
Scale	Scale	An integer value from 0 to 9, where 0 corresponds to 1, 1 corresponds to 10, 2 corresponds to 100, and so on	No scaling	No
Triangulation Currency	Triangulation Currency	A currency defined in the application	None	No
Reporting Currency	Reporting Currency	True, False, or an integer: non-zero is true; zero is false	False	No
Thousands Separator	Thousands Separator	default; none; comma; dot; space	None	No
Decimal Separator	Decimal Separator	default; dot; comma	dot	No
Negative Style	Negative Sign	default; prefixed; suffixed; parentheses	prefixed	No
Negative Color	Negative Color	default; black; red	black	No

Attribute Dimension Properties

Attribute, Parent, Alias: Default, Operation

Notes:

- For properties common to Attribute and Account members, see [“Account Dimension Properties” on page 91](#).
- **Custom attributes:** You can load attribute values to the attribute dimension for text, numeric, Boolean, and date attributes. If you modify properties and do not specify a value, the custom attribute is not changed in the application. To remove a custom attribute, specify <none> as the value. The value is passed as a string.
 - **Update:** Adds, updates, or moves the member that is being loaded.
 - **Delete Level 0:** Deletes the member that is being loaded if it has no children.
 - **Delete Idescendants:** Deletes the member that is being loaded and all of its descendants.

- **Delete Descendants:** Deletes the descendants of the member that is being loaded, but does not delete the member itself.

Use caution when deleting members; this deletes the member, its data, and any associated planning units.

Table 36 Attribute Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Attribute	Attribute Value Name	Text, subject to Planning member naming restrictions	None	Yes

UDA Dimension Properties

Dimension, UDA, Operation

For properties common to UDA and Account members, see [“Account Dimension Properties” on page 91](#).

Table 37 UDA Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Dimension	Name of base dimension for which UDA is defined	Text, name of dimension for which the UDA will be defined; UDAs cannot be defined for Attribute dimensions	None	Yes
UDA	UDA being defined	Text, subject to Planning member naming restrictions	None	Yes

Exchange Rate Dimension Properties

Table, Description, To Currency, From Currency, Method, Historical, Beg Balance, Year, Period, Average, Ending

Table 38 Exchange Rate Dimension Properties

Column Header in .CSV Load File	Planning Property	Value	Default	Required
Table	fxTblId	Name of the exchange rates table	None	Yes
Description	Description	Description of the exchange rates table	None	No

Column Header in .CSV Load File	Planning Property	Value	Default	Required
To Currency	toCur	Currency defined in the application, to which the conversion will be applied	None	Yes
From Currency	fromCur	Currency defined in the application, from which the conversion will be computed	None	Yes
Operation	n/a	update (delete operations are not supported: delete level 0, delete idescendants, delete descendants)	update	No
Method	method	multiply; divide	multiply	No
Historical	historicalRate	numeric value	0	No
Beg Balance	begBalanceRate	numeric value	0	No
Year	yearId	A year defined in the application, such as FY08	None	Yes, if Average or Ending is specified
Period	tpId	A base time period defined in the application, such as Jan	None	Yes, if Average or Ending is specified
Average	avgVal	numeric value	None	No
Ending	endVal	numeric value	None	No

Load File Considerations

Consider these points when working with load files:

- For each dimension in the application, you create a load file with fields corresponding to the dimension properties. Each load file can contain members for only one dimension. You can define multiple members for each dimension.
- The required fields are different for each dimension being loaded. See [“Dimension Properties” on page 89](#).
- The member name must be included as a field in the load file. If the member property value is not specified, the application default value for the property is used.

- When adding new members, unspecified values assume a default value or are inherited from the parent member's property value as appropriate. If the member exists and no value is specified, it is left as is.
- When you load a member that already exists in the Planning application (for example, to change a property), if the member already exists in the application and a parent is not specified in the load file, the member is left under the existing parent. If a new parent is specified, the member is moved under the new parent.
- To specify a null value, you can use the reserved value, <none>, for example, to delete an attribute assignment.
- Header record fields can appear in any order.
- Only one dimension can be loaded per load file.
- Column headers in the load file are case-sensitive.
- The records are loaded one by one. If a record fails to load, its associated exception is written to the exception file and the load process resumes with the next record.
- If errors are logged when loading a year, and the year was loaded into the application, its properties may not be what was specified for it in the load file. Correct the load file record and reload the year to set its properties correctly.
- Parent members must exist or be loaded before their child members. In most cases, the load file must be sorted in parent-child order, either explicitly or by using /H.
- Data values containing commas and quotation marks must be enclosed in quotation marks. These examples show how commas and quotation marks are interpreted.

Table 39 Examples of Data Values Containing Commas and Quotation Marks

Value	Interpretation
"quote"quote"	quote"quote
""quotedstring""	"quotedstring"
""",quoted,"",string,""	",quoted,"",string,""
""",quoted,"""",string,"""	",quoted,"",string,""

Loading Information with Data Integrator Adapter for Planning

Use Oracle Data Integrator Adapter for Planning to connect and integrate Planning with any database through Oracle Data Integrator (ODI). The adapter provides a set of Oracle Data Integrator Knowledge Modules (KMs) for loading metadata and data into Planning applications. For information on using ODI, see the *Oracle Data Integrator Adapter for Planning User's Guide*.

Loading Information with DIM Adapter for Planning

After installing and configuring Oracle's Hyperion® Data Integration Management Adapter for Planning, you can install and configure adapters to retrieve and write data for other Oracle products. After you configure an adapter, you must configure an application connection in Workflow Manager before extracting data from sources or writing data into targets. See the *Hyperion Data Integration Management Adapter for Planning Online Help*. Oracle's Hyperion® Data Integration Management Adapter for Planning is available only for Classic Planning applications.

Enabling Data Load of Essbase Data

You specify parameters to enable data to be loaded directly into an Essbase database.

- To specify parameters for loading data:
 - 1 Select **Administration**, then **Data Load Administration**.
 - 2 For **Available Data Load Dimensions**, select a dimension, and click **Go**.
The dimension corresponds to the information to be loaded.
 - 3 For **Available Driver Dimensions**, select the dimension to which you are loading data.
 - 4 Select the members of the driver dimension.
 - 5 Click **Save**.

For detailed instructions on loading, see the Planning adapter documentation as described in [“Overview of Loading Information” on page 77](#).

Loading Information with Administration Services

- To load information with Administration Services:
 - 1 Open **Administration Services Console**.
 - 2 See the Essbase documentation for instructions on connecting to the server and entering a username and password.
 - 3 Minimize the **Administration Services Console**.
 - 4 Open **Windows NT Explorer** (**Start**, then **Programs**, and then **Windows NT Explorer**).
 - 5 Browse to the directory containing data files to load.
 - 6 Select text files to load and drag them onto **Administration Services Console** on the **Windows Task Bar** at the bottom of the screen.
 - 7 Continue holding the mouse while dragging files onto the **Administration Services Console** window, then release it.
 - 8 See the Essbase documentation for instructions on selecting the application to which to load data files.

9 Select the database to which to load data files.

Select the database corresponding to the plan type in the Planning application into which you are loading data.

File Example

This example loads data values for the first quarter of 2008 for the Europe entity and the Gross Sales account in the Actual scenario and Final version.

EUROPE	GROSS SALES	ACTUAL	FINAL	JAN	2008	150
EUROPE	GROSS SALES	ACTUAL	FINAL	FEB	2008	110
EUROPE	GROSS SALES	ACTUAL	FINAL	MAR	2008	200

If a dimension uses the same value for all rows, you can place it in the header as a page dimension, as in this example:

ACTUAL	FINAL			
EUROPE	GROSS SALES	JAN	2008	150
ASIA	NET INCOME	FEB	2008	150
EUROPE	NET INCOME	FEB	2008	110
ASIA	GROSS SALES	JAN	2008	200

Loading Information with Performance Management Architect

You can load data and metadata with these Performance Management Architect features: flat files, interface tables, and Data Synchronization. For information, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide* or Online Help.

Loading Information with FDM and ERP Integrator

For Performance Management Architect applications, metadata is loaded into Performance Management Architect through ERP Integrator. For Classic applications, metadata is loaded directly into Planning. For both Performance Management Architect and Classic applications, you can load data with Oracle Hyperion Financial Data Quality Management, Fusion Edition or ERP Integrator.

To load data with FDM, see the *Oracle Hyperion Financial Data Quality Management Administrator's Guide*. To load data and metadata with ERP Integrator, see the *Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide*.

6

Using Data Forms

In This Chapter

About Data Forms.....	107
Understanding Data Forms.....	109
Creating Data Forms.....	110
Using Business Rules.....	128
Previewing Data Forms.....	136
Managing User Variables.....	136
Selecting User Variables.....	137
Deleting User Variables.....	137
Printing Data Form Definitions.....	137
Searching for Data Forms.....	138
Modifying Data Forms.....	138
Moving Data Forms.....	139
Deleting Data Forms.....	139
Overview of Importing Data Form Definitions.....	139
Importing and Exporting Data Forms.....	143
Planning Offline Considerations.....	144

About Data Forms

Data forms are grids for entering data. You can create many data forms to meet your users' needs.

Data Forms and Plan Types

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

Accounts on data forms are editable if their source plan type matches the data form's plan type. If you add an account to a data form for a plan type other than the account's source plan type, the account is read-only on that data form.

Data Forms and Access Permissions

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

Data Forms and Currencies

For single-currency applications, all entities use the currency selected when the application was created. For multicurrency applications, the selected Currency member on data forms determines the currency in which values display. When the Currency member Local is selected for rows or columns, no currency conversion occurs for those rows or columns, and users can enter data for the entities in their native currency. If a currency member other than Local is selected, data values are converted to the selected currency for that row or column, and the data form is read-only. You can enter data in rows or columns that have Currency or Local as the selected member. See [“Designing Data Forms for Multiple Currencies” on page 108](#).

Designing Data Forms for Multiple Currencies

To allow users to work with currencies other than entities’ base currencies, perform one task:

- Select members from at least two currencies to compare converted currencies in the same data form.
- Assign the Currency dimension to the page axis and select reporting currencies as members to convert currencies for all the members in the data form. Users can then select a currency member from the page axis and launch the Calculate Currencies business rule to view values in that currency.

Designing Data Forms for Drill-Through Information

If data forms contain members whose data is loaded from a source (ERP Systems) using FDM or ERP Integrator, users can drill through to view more details for the cell data source. To enable data forms for drill through, complete these tasks when designing data forms:

- Within FDM or ERP Integrator, complete setup tasks, and load data or metadata to Planning. See the *Oracle Hyperion Financial Data Quality Management Administrator's Guide* or *Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide*. For the current release, you do not need to set properties or configure Planning for Oracle Hyperion Financial Data Quality Management, Fusion Edition.
- Give appropriate access permissions, for example, to the data form and members, as described in [Chapter 3, “Setting Up Access Permissions.”](#) Drill-through is enabled against all dimensions that are sourced from Oracle Hyperion Financial Data Quality Management

ERP Integration Adapter for Oracle Applications. If drill-through is enabled in a cell to which a user has write access, the drill-through icon continues to display when users update the cell. However, the update does not tie back to the source data when the user drills through.

- For multicurrency applications, all currencies for an entity in the source system can be loaded. Exchange rates are loaded into the exchange rate table in Planning, and currency conversion is completed within Planning.

When users print data forms that include cells with drill-through information, a drill-through icon is displayed in those cells.

Data 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.

Data Forms and Attributes

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

Data Forms and Shared Members

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

Data Forms and Calculations

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

Understanding Data Forms

Before creating data forms, understand their components:

- [“Point of View” on page 110](#)
- [“Page Axis” on page 110](#)

- [“Rows and Columns” on page 110](#)

Point of View

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

To simplify the data form, in the point of view you can specify only relevant members or include user variables. See [“Defining the Page and POV” on page 124](#) and [“Managing User Variables” on page 136](#).

Page Axis

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

You can specify multiple page drop-down lists, and select members using relationship functions or attributes. Switch between member sets by selecting them from the page axis.

You display member names or aliases on the page axis. You can specify the number of members in a page dimension that enables a search drop-down list on the data entry page, useful if dimensions contain many members. See [“Defining the Page and POV” on page 124](#).

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 data forms, they can enter data into the cell where the Unit Sales row intersects with the January column.

By default, data forms have one set of rows and columns. You can add sets of rows and columns to create asymmetrical combinations of members. See [“Creating Asymmetric Rows and Columns” on page 124](#).

Creating Data Forms

To create data forms, define:

- Data form properties. See [“Setting Data Form Properties” on page 111](#).
- Row and column layout. See [“Setting the Row and Column Layout” on page 112](#).
- Page and POV. See [“Point of View” on page 110](#) and [“Page Axis” on page 110](#).
- Member selection. See [“Selecting Members” on page 117](#).
- Display options. See [“Setting Other Options for Data Forms” on page 125](#).

- Business rules selection and properties. See [“Selecting Business Rules” on page 128](#) and [“Setting Properties for Business Rules” on page 129](#).
- Access permissions. See [“Adding Access to Data Forms and Folders” on page 47](#).

You can also create composite data forms that display several data forms simultaneously, even those associated with different plan types. Users can enter data and see results aggregated to an upper-level intersection, such as Total Revenue. Some tasks for creating composite data forms are the same as for regular data forms.

To create composite data forms, you define:

- Data form properties. See [“Setting Data Form Properties” on page 111](#).
- Row and column layout. See [“Setting Rows and Columns for Composite Data Forms” on page 114](#).
- Business rules selection and properties. See [“Selecting Business Rules” on page 128](#) and [“Setting Properties for Business Rules” on page 129](#)
- Access permissions. See [“Setting Properties for Business Rules” on page 129](#).

► To create data forms:

- 1 Select **Administration**, then **Manage Data Forms**.
- 2 Select the folder under which to store the data form.
- 3 To create a:
 - Regular data form, click **Create**.
 - Composite data form, click **Create Composite**.
- 4 The **Data Form Design Wizard** guides you in creating a data form.

Selecting Data Forms and Folders

► To select data forms or data form folders:

- 1 Select **Administration**, then **Manage Data Forms**.
- 2 Under **Data Form Folders**, perform one action:
 - To select a data form, select the folder containing the data form, and select the data form.
 - To select a data form folder, select a folder.

Setting Data Form Properties

► To set data form properties:

- 1 Select the data form.
See [“Selecting Data Forms and Folders” on page 111](#).
- 2 Click **Edit**.

- 3 Enter a name of up to 80 characters.

You can change the name later.

- 4 **Optional:** Enter a description of up to 255 characters.
- 5 If **Plan Type** is displayed, select the plan type to associate with the data form. See [“Data Forms and Plan Types” on page 107](#).
- 6 Click **Next** to lay out the rows and columns.

For regular data forms, see [“Setting the Row and Column Layout” on page 112](#). For composite data forms, see [“Setting Rows and Columns for Composite Data Forms” on page 114](#).

Setting the Row and Column Layout

On the Row/Column Layout tab, assign dimensions to columns and rows on regular data forms. (For composite data forms, see [“Setting Rows and Columns for Composite Data Forms” on page 114](#).) For each dimension, select members for users to work with. You can then set properties for columns and rows. For example, you can show member names or aliases, open data forms with the hierarchy expanded, suppress missing data, and so on.

Segments are areas of data forms with special properties, such as separator lines for columns and rows and hidden or read-only data. You can define segments for asymmetric rows and columns. You can also suppress the hierarchy so row definitions are not indented and columns do not include line breaks.

When setting the row and column layout:

- You must select at least one member for each dimension.
 - A data form must have at least one dimension assigned to the row and column axis.
 - You cannot select the same dimension for multiple axes. (You can have dimensions on multiple axes if you set user variables in the POV.)
 - To move a dimension from one axis to another, delete it from the source axis and select it for the destination axis.
- To specify dimensions and members for the column and row axis:
- 1 Set the data form’s properties (see [“Setting Data Form Properties” on page 111](#)).
 - 2 On **Row/Column Layout**, under **Column Dimensions**, for **Dimension**, select the dimension to display first in the column axis.

To add dimensions to a column, click Add Dimension. To add columns, click Add Column. See [“Adding Column Definitions” on page 116](#).

- 3 Select each dimension’s members. Either:
- Under **Members**, enter the names.

- Under **Select**, click .

See [“Selecting Members” on page 117](#) and [“Creating Asymmetric Rows and Columns” on page 124](#).

4 Repeat the previous steps to assign more dimensions to the column axis.

5 **Optional:** In **Segment Properties**, set options:

Table 40 Segment Properties for Columns

Option	Description
Hide	Hide the segment on the data form
Show separator	Create a bold border before the segment to visually distinguish it
Read-only	Create a read-only segment to enable comparison of old, read-only data with new, editable data
Suppress hierarchy	For columns, do not display indentation

6 Under **Column Properties**, select options:

Table 41 Column Properties

Option	Description
Suppress missing data	Hide columns unless at least one cell has data. Clear to display the column with “#MISSING” in cells where data is missing.
Column width	<ul style="list-style-type: none">● Small: Display 7 decimal places● Medium: Display 10 decimal places● Large: Display 13 decimal places● Size-to-Fit: Force all column headings to fit in the displayed space● Custom: Display over 13 decimal places, up to 999
Display member formula	Allow users to view member formulas that are defined for members on columns, depending on access permissions.

7 Under **Row Dimensions**, select the dimension to display first in the row axis.

To add more dimensions to a row, click Add Dimension. To add more rows, click Add Row. See [“Adding Row Definitions” on page 116](#).

8 Select each dimension’s members. Either:

- Under **Member**, enter the names.

- Under **Select**, click .

See [“Selecting Members” on page 117](#) and [“Creating Asymmetric Rows and Columns” on page 124](#).

9 **Optional:** Repeat the previous steps to assign more dimensions to the row axis.

10 In **Segment Properties**, select options:

Table 42 Segment Properties for Rows

Option	Description
Hide	Hide the segment on the data form
Show separator	Create a bold border before the segment to visually distinguish parts of the data form
Read-only	Create a read-only segment in the data form to allow comparing old, read-only data with new, editable data
Suppress hierarchy	For rows, do not display indentation
Apply to all segments	Available if columns have multiple segments: apply settings to all column segments.

11 Under **Row Properties**, select options:**Table 43** Row Properties

Option	Description
Suppress missing blocks	Aid the performance of the Suppress missing data setting when suppressing a large number of rows, such as 90% or more. The Suppress missing blocks setting can degrade performance if few or no rows are suppressed. Test data forms before and after using this setting to determine if performance is improved. You should also test data forms whenever you make significant changes to your application. With this setting, attributes may not display in data forms. Also, certain suppressed blocks may ignore Dynamic Calc members.
Suppress missing data	Select to hide rows without data unless at least one cell has data. Clear to display rows with “#MISSING” in cells when data is missing. You cannot simultaneously select Suppress Missing Data and Allow Users to Dynamically Add Rows.
Allow users to dynamically add rows	Enable users who have write access to the data form to dynamically change and refresh the data form’s definition by adding rows. Selected members that they can access display on the data form. You cannot simultaneously select Allow Users to Dynamically Add Rows and Suppress Missing Data.
Display member formula	Allow users to view member formulas that are defined for members on rows, depending on access permissions.
Apply to all segments	Available if rows have multiple segments: apply settings to all row segments.

12 Click **Next** to set up the Page and Point of View.

See [“Defining the Page and POV”](#) on page 124.

Setting Rows and Columns for Composite Data Forms

For composite data forms, you can set options for the POV, page, and number and layout of columns and rows. You also select the data forms to display on the composite data form.

► To set the row and column layout for composite data forms:





- 1 Select the composite data form.

See [“Selecting Data Forms and Folders” on page 111](#).

- 2 Click **Edit**.
- 3 Set properties (see [“Setting Data Form Properties” on page 111](#)).
- 4 Select options:

Table 44 Row and Column Layout for Composite Data Forms

Option	Description
Combine POV	Combine the POV settings when displaying the composite data form
Combine Page	Combine the Page settings when displaying the composite data form
Form Layout	Enter the number of columns or rows to display, and select Columns or Rows

- 5 **Optional:** Set the layout of columns and rows in the composite data form:
 - In Data Form Layout, enter the number of columns and select **Columns**. Select the data forms to display in the columns, and set the column order by clicking  and . The data form at the top of the list displays in the leftmost column of the composite data form.
 - In Data Form Layout, enter the number of rows and select **Rows**. Select the data forms to display in the rows, and set the row order by clicking by clicking  and . The data form at the top of the list displays in the composite data form's top row.

Setting Column and Row Properties

➤ To set properties for individual rows and columns:


- 1 Select the data form.
See [“Selecting Data Forms and Folders” on page 111](#).
- 2 Click **Edit**.
- 3 Set properties (see [“Setting Data Form Properties” on page 111](#)).
- 4 Specify dimension members for the rows and columns (see [“Setting the Row and Column Layout” on page 112](#)).
- 5 On **Row/Column Layout**, to the right of the row or column dimension whose property you are setting, click .
- 6 Select options:

Table 45 Property Options for Individual Rows and Columns

Option	Description
Display	Select Member Name or Member Alias
Start expanded	Start the member list expanded


Option	Description
Hide dimension	Hide this dimension
Enable custom attribute display	Enable custom attributes to display on the data form

7 Click **Submit**.

Adding Column Definitions

On the Additional Column Definition page, you can further define the member selections or create asymmetric columns. See [“Creating Asymmetric Rows and Columns” on page 124](#).

➤ To select dimension members in the column axis:


- 1 On the **Additional Column Definition** page, under **Select**, click .
- 2 On the **Member Selection** page, select dimension members.
- 3 Click **Submit**.
- 4 Repeat these steps as needed.
- 5 Click **Submit**.

Use the Edit Columns button on the Row/Column Layout tab to further define the column’s dimensions and members.

Adding Row Definitions

On the Additional Row Definition page, you can further refine the member selections or create asymmetric rows. See [“Creating Asymmetric Rows and Columns” on page 124](#).

➤ To further select dimension members in the row axis:

- 1 On the **Additional Row Definition** page, under **Select**, click .
- 2 On **Member Selection**, select dimension members.
- 3 Click **Submit**.
- 4 Repeat these steps as needed.
- 5 Click **Submit**.
- 6 Click **Submit**.

Use the Edit Rows button on the Row/Column Layout tab to further define row dimensions and members.

Changing Column Definitions

Use the Edit Columns Definition page to further define or remove column definitions.

- To set or change dimension members assigned to a column or remove the column definition:
 - 1 On **Row/Column Layout**, click **Edit Columns**.
 - 2 On **Edit Column Definitions**, by the column to change or delete, click:
 - Edit to further define the column's member selections.
 - Delete to delete the additional column definition (including its member selection).
 - 3 Click **Return to Data Form Layout**.

Changing Row Definitions

Use the Edit Row Definitions dialog box to further define or remove row definitions. You can also delete rows dynamically added by users when “Allow users to dynamically add rows” is selected for data forms.




- To set the dimension members assigned to a row or remove the row definition:
 - 1 On **Row/Column Layout**, click **Edit Rows**.
 - 2 On the **Edit Row Definitions** page, by the row to change or delete, click:
 - Edit to further define member selections for the row.
 - Delete to delete the additional row definition (including its member selection).
 - 3 Click **Return to Data Form Layout**.

Selecting Members

On the Member Selection page, select members for data forms, business rule runtime prompts, and Clear Cell Details. You can select members by the member name, alias, or both, depending on the setting for the member in the current application.

- To select members:
 - 1 **Displayed only for business rules having Cross Dimension or Member Range runtime prompts:** From the **Select Dimension** list, select a dimension that the business rule designer set for this runtime prompt.
 - 2 From the members list, select members. To select all members, select the check box at the top of the list.

Only members to which you have access are displayed. For runtime prompts, members are displayed that satisfy runtime prompt limits.

Search for a member in the hierarchy by clicking . For Search, enter part or all the member's name and click  or .

For runtime prompts only: The displayed hierarchy is for the application and plan type against which the business rule is launched.

3 Move members to or from the **Selected Members list:**




- Click  to add selected members.
- Click  to remove selected members.
- Click  to remove all members.
- Use the **Member** list to select a range of members based on hierarchy relationships.

Table 46 Member Relationships

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

- If attributes are defined, you can select attribute values.
Selecting a non-level 0 attribute selects all level 0 descendants and applies the operator to each, as described in [“Selecting Attribute Values as Members” on page 119](#).

Table 47 Attribute Value Selection

Operator	Attribute Values Included
Equal	Equal to the selected attribute
NotEqual	Not equal to the selected attribute
Greater	Greater than the selected attribute

Operator	Attribute Values Included
GreaterOrEqual	Greater than or equal to the selected attribute
Less	Less than the selected attribute
LessOrEqual	Less than or equal to the selected attribute

- 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.

See [“Specifying System Settings” on page 166](#) and [“Selecting Substitution Variables as Members” on page 122](#).

4 Click **Submit**.

When selecting members for data forms:

- To filter members from certain users, restrict their access permissions to members, and then refresh the plan. See [“Assigning Access to Members and Business Rules” on page 42](#).
- The order of members in the Selected Members list determines the order on data forms. To change the order, select a member and click the Up or Down Arrow in the Selected Members title bar.
- To define different sets of members for a dimension, see [“Creating Asymmetric Rows and Columns” on page 124](#).
- To set display, functionality, and printing options, see [“Setting Other Options for Data Forms” on page 125](#).
- To set up reports that include data forms or data form definitions, see [“Customizing Reports” on page 241](#).

Selecting Attribute Values as Members

If attribute members are defined, you can select attribute values on the Member Selection page, as described in [“Selecting Members” on page 117](#). For attribute members, selecting a non-level 0 attribute selects all level 0 descendants and applies the operator to each. For attributes of type numeric, date, and Boolean (where false = 0 and true = 1), evaluation is based on the minimum and maximum values. For text attributes, evaluation is based on the position from top to bottom in the hierarchy. The top position has the lowest value, and the bottom position has the highest value.

Example: Numeric attribute

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

Size

Small

1

2

Medium

3

4

Large

5

6

Table 48 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

Selected Operator	Selected Attribute Value	Result	Explanation
			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 49 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.

Selected Operator	Selected Attribute Value	Result	Explanation
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 Substitution Variables as Members

Substitution variables act as global placeholders for information that changes regularly. Each variable has an assigned value that can be changed centrally on the Essbase server. Substitution variables are especially useful for developing and reporting on rolling forecasts. When you select substitution variables as members on the data 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 data form or the report script.

Notes:

- When you open or calculate values on data forms, Essbase replaces substitution variables with values assigned to them. By default, each substitution variable is retrieved and cached from the Essbase server every five minutes (or 300 seconds). You can change the retrieval interval by adding the `SUBST_VAR_CACHE_LIFETIME` property and setting its value in seconds.
- You create and assign values to substitution variables using Administration Services Console or ESSCMD. These substitution variables are then available in Planning when you select members for a data form.
- Substitution variables must be appropriate for the context in data forms. For example, you could select a substitution variable named `CurQtr` with a value of `Qtr2` as a member of the Time Period dimension. It is not valid to select a substitution variable named `CurYr` for the Year dimension if its value is `Feb`. You can set substitution variables on the Essbase server, application, or database level. The same substitution variable can exist on multiple levels; Planning uses the first one it finds as it searches in this order: 1) database 2) application 3) server.

- Planning checks the validity of substitution variables when they are used (for example, when the data form is opened). It does not check when you design data forms, so you should test substitution variables by saving and opening data forms.
- For information on errors generated when substitution variables are calculated, you can check several logs. See the Essbase server log for information on attempts to use a substitution variable that is no longer contained in Essbase. See the Planning log for information on substitution variables that are not valid in the data form. For information about logs, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.

► To specify substitution variables in data forms:

- 1 Create the data form (see “[Creating Data Forms](#)” on page 110).
- 2 On **Member Selection**, select substitution variables the same way you select members, using the right, left, and double arrows to move substitution variables to and from **Selected Members**.

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

```
&CurrentScenario
```

- 3 Click **Submit**.

Selecting User Variables as Members

User variables act as filters in data forms, enabling planners to focus only on certain members, such as a department. Before you can associate a user variable with a data form, you must create the user variable. See “[Managing User Variables](#)” on page 136.

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

► To select user variables for data forms:

- 1 Create the data form (see “[Creating Data Forms](#)” on page 110).
- 2 On **Member Selection**, select user variables the same way you select members, using the arrows to move substitution variables to and from **Selected Members**.

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

```
Division = [User Variable]
```

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

```
Idescendants (&Division)
```


- 3 Click **Submit**.

Creating Asymmetric Rows and Columns

Asymmetric rows and columns are ones in which different sets of members are selected across the same dimension. 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 On **Row/Column Layout**, click **Add Row** or **Add Column**.
 - 2 On the **Additional Row or Column Definitions** page, next to the dimension for which to select asymmetric members, under **Select**, click .
 - 3 Select a different set of members for the row or column.
 - 4 Click **Submit**.
 - 5 Repeat these steps as needed.
 - 6 Perform one action:
 - Click **Submit** to save your selections.
 - Click **Return to Data Form Layout** to continue designing the data form.

Enabling Users to Add Rows

You can create data forms that allow users with write access to add rows with members. For example, some users budget for next year based on the current year actual data. They want to see money spent for a given account or use historical trends to build a new plan. Yet they also need to add rows to data forms to budget for new items with new dimension members. See [“Setting the Row and Column Layout” on page 112](#).

Defining the Page and POV

You can select dimensions and members for the page axis, display member names or aliases, and control their display on data forms. You can also define the POV dimensions and members for dimensions valid for the data form plan type and not assigned to a page, column, or row axis. The POV sets the unique dimension members that define intersections of data.

When you set user variables for data forms, the variable name displays on the POV toolbar. See [“Managing User Variables” on page 136](#).

- To define the page axis and POV:
 - 1 In the **Page Dimensions** area, for **Dimension**, select the dimension to display first in the page axis.
 - 2 Select each dimension’s members:
 - In **Members**, enter the names.

- Under **Select**, click .

See [“Selecting Members” on page 117](#).

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

3 Repeat the previous steps to assign multiple dimensions to the page axis.

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

4 In the **Properties** area, select or clear options for page dimensions.

- **Member Name and Alias:** Display the member name or member alias.
- **Display Member Formula:** Allow users to view member formulas that are defined for members on the page or POV, depending on access permissions.

5 In the **Point of View Dimensions** area, select each dimension’s member:

- For **Members**, enter the name.

- Under **Select**, click .

See [“Selecting Members” on page 117](#).

6 In the **Properties** area, select or clear options for point of view dimensions:

- **Visible:** Make the dimension member name or alias visible on the data form.
If **Visible** is not selected, the member name or alias does not display on the data form. Hiding irrelevant members can make data forms easier to use.
- **Member Name and Alias:** Display the member name or alias.
- **Display Member Formula:** Allow users to view member formulas that are defined for members on the page or POV, depending on access permissions.
- **Enable Dynamic User Variables:** Enable dynamic user variables for the data form.

7 Save the data form by performing one action:

- Click **Save** to save the data form and return to the **Manage Data Forms** page.
- Click **Save As** to save the current data form definition and settings under a new name. The previous data form remains intact.
- Click **Previous** to return to the previous tab.
- Click **Next** to continue setting up the data form.

Setting Other Options for Data Forms

You can set options for the data form display, functionality, and printing. 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.

You can also enable Smart Lists and account-level annotations. Users can add annotations to accounts in data forms if they have write access to the account, entity, scenario, and version members. Account level annotations can vary by different combinations of Scenario, Version, and Entity dimensions. Note:

- The Account dimension must be assigned to a row axis.
- Account, Entity, Versions, and Scenario dimensions cannot be assigned to the column axis.
- The Entity dimension can be assigned to the row, page, or POV axis.
- Version and Scenario dimensions must be assigned to the page or POV axis.

➤ To set data form display options:

- 1 In the **Precision** area, select options to set the number of decimal positions displayed for **Currency Values**, **Non-Currency Values**, and **Percentage Values**.

Specify Minimum values to add zeros to numbers with a small number of decimals. Specify Maximum values to truncate and round the decimal portion of longer numbers. For example:

Table 50 Data Precision Examples

Value	Minimum Precision	Maximum Precision	Displayed Value
100	0	Any	100
100	3	Any number >=3 or None	100.000
100.12345	Any number <=5	None	100.12345
100.12345	7	None	100.1234500
100.12345	Any number <=3	3	100.123
100.12345	0	0	100
100.12345	2	4	100.1234
100	2	4	100.00

- 2 In the **Display Properties** area, select options:

- **Display missing values as blank:** Leave data form cells empty where data does not exist. If this option is not selected, empty cells display the text “#MISSING.”
- **Allow multiple currencies per entity:** If the application supports multiple currencies, allow entities to support multiple currencies, regardless of base currency. Users can select currency for displayed cell values in data forms.
- **Enable Mass Allocate:** Enable Mass Allocate for the data form. (Users must also have the Mass Allocate role to use Mass Allocate.)
- **Enable Grid Spread:** Enable Grid Spread for the data form.

For information on using Mass Allocate and Grid Spread, see the *Oracle Hyperion Planning User's Online Help* and [“Creating Spreading Patterns” on page 253](#).

- **Message for data forms with no data:** Enter text to display in data form rows for queries that have no valid rows. Leave blank to display the default text, *There are no valid rows of data for this data form.*
- **Optional:** Select **Make data form read-only**. You cannot set this option for composite data forms.
- **Optional:** Select **Make data form hidden**. For example, hide data forms that are part of composite data forms or are accessed from menus or task lists.
- **Enable cell-level document:** Enable users to add, edit, and view documents in cells in the data form, depending on access permissions (the default). To prevent users from using documents in a data form, clear this option. To use cell documents, see the *Oracle Hyperion Planning User's Online Help*. For information on access, see [Chapter 3, "Setting Up Access Permissions."](#)

3 In the **Printing Options** area, set preferences for displaying information in data forms:

Table 51 Printing Options

Option	Description
Include supporting detail	<p>Include supporting detail as extra rows in PDF files. Specify how supporting detail is displayed:</p> <p>Normal Order: Prints supporting detail in the same order as on the Supporting Detail page, after the member it is associated with</p> <p>Reverse Order: Prints supporting detail in reverse order, before the member associated with it. Supporting detail for children displays above parents, and the order of siblings is preserved</p>
Format data	Apply number format settings from the data form to the displayed data
Apply precision	Apply the data form's precision settings (desired number of decimal points) to the displayed data in PDF files
Show account annotations	If account annotations are enabled for the data form, select to display account annotations in PDF files
Show cell text	Display text notes associated with cells
Show attribute members	If attribute members are selected in the data form, display them in PDF files
Show currency codes	If the data form supports multiple currencies, display currency codes in PDF files





You can also create reports for data forms, as described in ["Customizing Reports"](#) on page 241.

4 In the **Smart View Options** area, set options:

- **Optional** (not available for composite data forms): Select **Enable for Smart Slice** to enable the data form for ad hoc analysis in Smart View. (See the *Oracle Hyperion Smart View for Office User's Guide*.)
- **Optional** (not available for composite data forms): Select **Enable Offline Usage** to enable the data form to be used offline.

This setting is applicable when the application is enabled for offline (the default setting). You can prevent the current application from being used offline by selecting Administration, then Manage Properties, then the Application Properties tab, and then changing `ENABLE_FOR_OFFLINE` to `False`. See [“Setting Planning Properties” on page 30](#).

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

-  moves selections
-  moves all
-  removes selections
-  removes all

6 If you selected multiple menus, set the order in which they are displayed by clicking  or .

Multiple menus are displayed sequentially, with separators between them.

7 Click **Save**.

Using Business Rules

For Performance Management Architect applications, you create business rules using Calculation Manager. Classic applications can use business rules created with Calculation Manager or with Business Rules, a module of Administration Services. With appropriate access, users can launch business rules from Planning. Business rules can also prompt users for input when rules are launched.

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

For information on:

- Creating business rules with Business Rules, see the *Hyperion Business Rules Administrator's Guide*.
- Creating business rules with Calculation Manager, see the Calculation Manager part of the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.
- Using runtime prompts, see [“About Runtime Prompts” on page 130](#).

Selecting Business Rules

On the Business Rules tab, you can associate multiple business rules with a data form, by plan type. Users can launch associated business rules from the data form to calculate and allocate values. You can set whether each business rule associated with a data form automatically launches when the data form is opened or saved.

- To select business rules for data forms:
- 1 Select the data form to which to add business rules (see [“Selecting Data Forms and Folders” on page 111](#)).
 - 2 Click **Edit**.
 - 3 Set properties (see [“Setting Data Form Properties” on page 111](#)).
 - 4 On **Row/Column Layout**, specify dimensions and members for rows and columns (see [“Setting the Row and Column Layout” on page 112](#)).
 - 5 On **Page/Point of View Layout**, specify dimensions and members for the page axis (see [“Defining the Page and POV” on page 124](#)).
 - 6 On **Other Options**, set display preferences (see [“Setting Other Options for Data Forms” on page 125](#)).
 - 7 On **Business Rules**, for **Plan Type**, select the plan type.
 - 8 From the **Business Rules** list, select the business rules to associate with the data form.
 - 9 Move the business rules to **Selected Business Rules** by clicking the **Right arrow** or **Double Right Arrows**.

By default, the Calculate Data Form and Calculate Currencies business rules are selected. Calculate Data Form is automatically created for data forms to calculate subtotals. Calculate Currencies is created for data forms that include multiple currencies in a row, column, or page, to enable converting values among the available currencies. You can clear Calculate Currencies if you use customized calc scripts to calculate currency conversions. You can clear Calculate Data Form to prevent planners from calculating data in data forms.

- 10 To set the order of selected business rules (the order in which rules are displayed and launched), select a business rule in **Selected Business Rules** and click the up or down arrow.

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

- 11 To set properties for the business rules, click **Properties**.
See [“Setting Properties for Business Rules” on page 129](#).
- 12 Perform one action:

- Click **Save**.
- Click **Save As** to save the current data form definition under a new name. The previous data form remains intact.
- Click **Previous**.
- Click **Next**.

You can select only business rules that you can access.

Setting Properties for Business Rules

You can specify whether business rules associated with data forms launch automatically when users open or save the data 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 POV

axes. You set certain properties for business rules when creating data forms. See [“Creating Data Forms” on page 110](#) and [“Understanding Runtime Prompts” on page 131](#).

➤ To set business rule properties:

- 1 **For Business Rule Properties**, select **Run on Load** next to a business rule to launch it automatically when the data form is opened.

Business rules having runtime prompts cannot launch on load.

- 2 Select **Run on Save** next to a business rule to launch it automatically during save.

For business rules containing runtime prompts that are set to Run on Save, before the rules are saved, users are prompted to enter the runtime prompt.

- 3 **Optional:** If a business rule has runtime prompts, to match the default member selection on the runtime prompt window to the current members in the page and POV axes of the open data form, select **Use Members on Data Form**.

To learn how this option interacts with other settings and conditions, see [“Understanding Runtime Prompts” on page 131](#).

- 4 **Optional:** To hide the runtime prompt value from the user, select **Hide Prompt**, which automatically selects **Use Members on Data Form**.

After saving the data form, the next time you return to this page, Use Members on Data 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 data form's Page/POV)
- No dimensions are repeated in the runtime prompt

- 5 Click **OK**.

About Runtime Prompts

When launched, business rules can prompt users for such variables as members, text, dates, or numbers. Prompts should be specific and tell users what type of data is expected. For example:

- Select a month.
- Enter the expected number of customer visits per quarter.
- What percentage change in earnings do you expect next month?

When launching business rules with runtime prompts, Planning validates the value entered, but not the business rule. To set the default member selection in a runtime prompt, see [“Setting Properties for Business Rules” on page 129](#). To understand how other settings and conditions affect runtime prompts, see [“Understanding Runtime Prompts” on page 131](#).

By default, the values for processed runtime prompts in the application are stored in the database and available for viewing from the Job Console (select Tools, then Job Console). If many users are running business rules with runtime prompts, tracking these values consumes significant

system resources. To improve performance, you can turn off this function so Planning does not capture runtime prompt values. To do so, add the `CAPTURE_RTP_ON_JOB_CONSOLE` property to the properties table, with the property value of `FALSE` (to turn it back on again, change its value to `TRUE`). See [“Setting Planning Properties” on page 30](#).

Understanding Runtime Prompts

The display and values of runtime prompts are affected by such aspects as:

- Whether there are valid members on the data form's Page/POV and whether the Use Members on Data Form and Hide Prompt options on the Business Rule Properties tab are selected (see [“Setting Properties for Business Rules” on page 129](#))
- Whether the Runtime Prompt is hidden during design, set when designing the runtime prompt (for business rules created with Business Rules, see the *Hyperion Business Rules Administrator's Guide* and for business rules created with Calculation Manager, see the Calculation Manager part of the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*)
- Whether the Use Last Value property is set when designing the business rule

Principles:

1. When launched from a data form, the values of members in the Page/POV take precedence if the Use Members on Data 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/POV members.

This is not the case when business rules associated with composite data forms are launched on save or from the left-hand pane or when business rules are launched from the Tools, then Business Rules menu. In these cases, the Use Members on Data Form setting is ignored, hidden runtime prompts get design-time values, and the last saved value takes precedence.

2. If the Use Last Value option is selected for the runtime prompt at design time, and if any of these conditions exist:
 - Use Members on Form is not selected
 - A runtime prompt is launched from the Tools, then Business Rules menu
 - Values cannot be pre-filled from the context

Then the precedence of runtime prompt values is determined by:

- a. The last saved value takes precedence.
- b. If a ruleset is launched, the value overridden at the ruleset-level at design-time is used.
- c. If a business rule is launched, the value overridden at the rule-level at design-time is used. If it is not overridden at the rule-level, the runtime prompt value at design-time is used.

Runtime prompts that are hidden at design time never use the last saved value. In these cases, the Use Last Value setting is ignored.

3. The Use Members on Data 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).
4. For Cross Dimension runtime prompts: the runtime prompt is not hidden unless all the prompts in the runtime prompt can be pre-filled from the Page/POV. The runtime prompt is displayed with some values pre-filled from the Page/POV and others filled according to Principles 1 and 2.

This table describes the result on runtime prompts of these settings and conditions:

Table 52 How Member Availability and Other Settings Affect Runtime Prompts

Availability of member on the Page/POV	Use Members on Data Form option is selected	Hide Runtime Prompt property is set during runtime prompt design	Hide Prompt option is selected for the data form	Result on Runtime Prompt
The member is available on the Page/POV 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 Page/POV member.
The member is available on the Page/POV to use as the runtime prompt value.	Yes	No	Yes	If all runtime prompts can be pre-filled from the Page/POV context and are valid and within limits, the runtime prompts are not displayed. However, if even one runtime prompt value cannot be pre-filled from the Page/POV context, then all runtime prompts display, with values pre-filled wherever possible. All others follow Principle 2.
The member is available on the Page/POV to use as the runtime prompt value.	Yes	No	No	The runtime prompt is displayed to users, with values pre-filled from the Page/POV.
The member is not available on the Page/POV 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 2.

Availability of member on the Page/POV	Use Members on Data Form option is selected	Hide Runtime Prompt property is set during runtime prompt design	Hide Prompt option is selected for the data form	Result on Runtime Prompt
				For example, the data form context cannot be passed because the dimension of the runtime prompt is on rows or columns, so the Hide Prompt setting is ignored and the runtime prompt displayed.
The member is not available on the Page/POV to use as the runtime prompt value.	Yes	No	Yes	The runtime prompt is displayed to users, with values pre-filled according to Principle 2.
The member is not available on the Page/POV to use as the runtime prompt value.	Yes	No	No	The runtime prompt is displayed to users, with values pre-filled according to Principle 2.
The member is available on the Page/POV 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.
The member is available on the Page/POV 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 2.
The member is not available on the Page/POV 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.
The member is not available on the Page/POV 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 2.

When hidden runtime prompt values are ambiguous, note:

- If the data form context cannot be passed in (because the dimension is on the row or column, for example), hidden runtime prompts are displayed.
- With hidden Cross Dimension runtime prompt types, if all prompts cannot be passed in from the context, the runtime prompt displays with values pre-filled context values and design time values for dimensions on the row or column. For example, if the Cross Dimension has runtime prompts for Period, Entity, and Scenario, and Entity is defined on the row, then the runtime prompt displays with the POV Scenario -> design time value for Entity -> Page Period.
- If the context can be passed in for the runtime prompt value, but it is out of limits, then the runtime prompt is displayed with the context value pre-filled.
- If there is more than one runtime prompt of type Member or type Cross Dimension combined, then the runtime prompts are displayed with the context values pre-filled. For example, if there is a member type runtime prompt for the Entity dimension and a Cross Dimension type runtime prompt with one prompt for the Entity dimension, then both runtime prompts are displayed. This rule does not apply to Calculation Manager rulesets or Business Rules sequences
- When launched from the Tools, then Business Rules menu, runtime prompts are hidden and the design-time value (overridden at the rule or ruleset level) is used to launch the business rule. In Calculation Manager, a variable cannot be hidden unless a value is provided. 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.

Launching Business Rules With a Utility

Using the `CalcMgrCmdLineLauncher.cmd` utility, administrators can launch—from the Command Prompt—business rules or rulesets created with Calculation Manager.

Notes:

- Use the `/Validate` option to check the command syntax before you launch the business rule or ruleset.
- You cannot launch cross-application rulesets with this utility.
- If the launched business rule has a runtime prompt, before running `CalcMgrCmdLineLauncher.cmd`, create a file containing the runtime prompt values. You can either manually create an ASCII file that contains the runtime prompt values, or you can automatically generate the values file by selecting the Create Runtime Prompt Values File option on the Runtime Prompts page.
- If the launched ruleset has a runtime prompt, default launch values are used, and you do not specify a runtime prompt values file.

► To launch a business rule or ruleset with `CalcMgrCmdLineLauncher.cmd`:

1 When launching a business rule (not a ruleset) having runtime prompts, generate a file containing the runtime prompt values by either:

- Specifying the name of the runtime prompt values file that you generated on the Runtime Prompts page (see the *Oracle Hyperion Planning User's Online Help*).
- Creating a runtime prompt ASCII file that contains, on separate lines, each runtime prompt name and its value, separated by a double colon (::). For example:

```
CopyDataFrom: :Jan
```

```
CopyDataTo: :Apr
```

This file specifies that the value for the runtime prompt named `CopyDataFrom` is `Jan`, and that the value for the runtime prompt named `CopyDataTo` is `Apr`.

Save the file in the `HYPERION_HOME/products/Planning/bin` directory. Alternately, when executing the utility, specify the full path to the runtime prompt file.

2 From the `bin` directory, enter this command at the Command Prompt, one space, and the parameters, each separated by a space:

```
CalcMgrCmdLineLauncher.cmd [-f:passwordFile] /A:appname /U:username /
D:database [/R:business rule name | /S:business ruleset name] /F:runtime
prompts file [/validate]
```

If you installed Planning to the default directory, `CalcMgrCmdLineLauncher.cmd` is installed in `HYPERION_HOME/products/Planning/bin`.

Table 53 CalcMgrCmdLineLauncher Parameters

Parameter	Purpose	Required?
<code>[-f:passwordFile]</code>	If an encrypted password file is set up, you can use this option as the first parameter in the command line to run the utility with the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37.	No
<code>/A:appname</code>	Specify the Planning application from which the business rule is launched	Yes
<code>/U:username</code>	Specify the administrator's user name	Yes
<code>/D:database</code>	Specify the name of the plan type against which the calculation is launched	Yes
<code>[/R:business rule name /S:business ruleset name]</code>	Specify the name of the business rule or ruleset to be launched	Yes. Specify either a business rule or a ruleset, but not both.

Parameter	Purpose	Required?
<i>/F:runtime prompts file</i>	Specify the name of the file containing business rule's runtime prompt names and values	Yes, when launching a business rule having a runtime prompt. When launching a ruleset, default launch values are used. If you specify a runtime prompts file when launching a ruleset, the runtime prompts file is ignored.
<code>[/validate]</code>	Checks the command syntax only; does not launch the business rule or ruleset. Any errors are displayed on the console and written to the Calculation Manager log file. If the <code>CalcMgrLog4j.properties</code> file is in the <code>Classpath</code> , the log file is generated in the <code>HYPERION_HOME/logs/Planning</code> directory. Each rule in a ruleset is validated.	No
<code>/?</code>	Print the syntax and options for <code>CalcMgrCmdLineLauncher.cmd</code>	No

For example to launch the rule named Depreciate, using runtime prompt values in a file called `Values.txt`, enter:

```
CalcMgrCmdLineLauncher.cmd /A:planapp /U:admin /D:plan1 /R:Depreciate /F:Values.xml
```

- 3 If prompted, enter your password.

Previewing Data Forms

While you are designing data forms, you can preview the dimensions that are assigned to the POV, columns, rows, and page axes. Previewing does not display member attributes, alias, or data associated with data forms.

- To preview a data form's design, select **Preview**.

Managing User Variables

You can set user variables to limit the number of members displayed on a data form, helping users focus on certain members. For example, if you create a user variable called `Division` for the `Entity` dimension, users can select a member for their own division. You can create any number of user variables for each dimension, and select user variables for any axis in the data form. See [“Defining the Page and POV” on page 124](#).

The typical sequence of steps:

1. If necessary, create the appropriate parent-level members in the dimension outline.
2. Define user variables for each dimension you want users to be able to filter.
See [“Selecting User Variables” on page 137](#).
3. When designing the data form, associate the user variable with the data form.
See [“Selecting User Variables as Members” on page 123](#).
4. Instruct users to select a member for the user variable associated with the data form.
Before users can open data 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 data form or in preferences. See the *Oracle Hyperion Planning User’s Online Help*.

Selecting User Variables

► To select user variables:

- 1 Select **Administration**, then **Manage User Variables**.
- 2 Click **Add**.
- 3 On the **Select User Variable** page, for **Dimension Name**, select the dimension for which to create a user variable.
- 4 For **User Variable Name**, enter the name of the user variable.
- 5 Click **OK**.

You can now associate the user variable with a data form. See [“Selecting User Variables as Members” on page 123](#). After that, planners can select members for the user variable. See the *Oracle Hyperion Planning User’s Online Help*.






Deleting User Variables

► To delete user variables:



- 1 Select **Administration**, then **Manage User Variables**.
- 2 Select the user variable to delete.
- 3 Click **Delete**.
- 4 Click **OK**.

Printing Data Form Definitions

Administrators can print data form definition reports that include information on dimension members, business rules, access permissions, and other data form components. You can also create reports for data form definitions, as described in [“Customizing Reports” on page 241](#).

- To create and print data form definition reports:
 - 1 Select **Administration**, then **Reporting**.
 - 2 Select **Data Forms**.
 - 3 Select the data form definitions to print:
 - To select data forms, move them to **Selected Data Forms** by selecting them and clicking  or .
 - To remove data forms, select them from **Selected Data Forms** and click  or .
 - 4 **Optional:** Select **Include Member Selection List** to include column and row members on the report.
 - 5 **Optional:** Select **Include Business Rules** to include associated business rules.
 - 6 Click **Create Report**.
Adobe Acrobat generates a consolidated report, including:
 - Plan type
 - Description
 - Column dimension and members and additional column definitions
 - Row dimension and members and additional row definitions
 - Page and Point of View dimensions
 - Data form access permissions
 - Associated business rules
 - 7 To print the report, click  on the Adobe Acrobat toolbar.

Searching for Data Forms

- To search for data forms:
 - 1 Select **Administration**, then **Manage Data Forms**.
 - 2 For **Search**, enter part or all the data form name.
Ignoring capitalization, Search finds the next match.
 - 3 Click  to search forward (down) or  to search backwards (up).

Modifying Data Forms

- To modify data forms:
 - 1 Select the data form.
See “[Selecting Data Forms and Folders](#)” on page 111.

- 2 Click **Edit**.
- 3 Modify the data form. See:
 - “Setting Data Form Properties” on page 111
 - “Setting the Row and Column Layout” on page 112
 - “Defining the Page and POV” on page 124
 - “Selecting Members” on page 117
 - “Setting Other Options for Data Forms” on page 125
 - “Selecting Business Rules” on page 128
 - “Changing Access to Data Forms and Folders” on page 47

Moving Data Forms

➤ To move data forms:

- 1 Select the data form.

See “Selecting Data Forms and Folders” on page 111.

You can move multiple data forms simultaneously if they are in the same folder.

- 2 Click **Move**.
- 3 Select the destination folder.
- 4 Click **OK**.

Deleting Data Forms

➤ To delete data forms:

- 1 Select the data form.

See “Selecting Data Forms and Folders” on page 111.

- 2 Click **Delete**.
- 3 Click **OK**.

Overview of Importing Data Form Definitions

Use the `ImportFormDefinition` utility to import a data form definition, but not data, from a text-based file into a Planning data form. You can import rows, columns, or both. Planning imports only rows or columns that contain data. You must run the utility on a Windows system.

To import data form definitions:

1. Prepare the data form for importing.

- See [“Preparing Data Forms” on page 140](#).
2. Prepare the data file.
See [“Preparing Data Files” on page 140](#).
 3. Run the utility.
See [“Importing Data Form Definitions” on page 141](#) and [“ImportFormDefinition Examples” on page 142](#).

Preparing Data Forms

Before importing rows and columns for data form definitions, create the data form by setting up dimensions on the row, column, page and POV, depending on your requirements. Typically you define column layout when defining data forms and use `ImportFormDefinition` to import only rows. The resulting data form looks like the data form definition.

The rows imported from the data file are based on members specified on the data form and import options, determining which data is imported. See [“ImportFormDefinition Examples” on page 142](#).

Preparing Data Files

`ImportFormDefinition` imports data from a text-based file in Essbase Column Export Format. You can generate the file directly, or, if you have a method for loading data into Essbase, you can load the data and generate the file.

To create an Essbase Column Export Format file using Essbase, use Administration Services Console, and select Database, then Export. Specify the Server File Name and select Export in Column Format. (Oracle recommends that you also select Level 0 Data.) You need not modify the file after exporting it from Essbase.

If you generate the file yourself:

- The file's first line represents the column of the data file. It must be a list of members from one dimension.
- Each line after the first line must include a member from every dimension other than the one representing the column, followed by data.
- Member names must be enclosed in double quotation marks.
- The delimiter must be a space.
- Data must not be enclosed in double quotation marks.
- Data cells that are blank must include `#MISSING`.

The layout of the data form, not the format of the data file, determines how the resulting data form displays. You can use the same data file to load different data forms.

Importing Data Form Definitions

By default, the ImportFormDefinition utility is in: *HYPERION_HOME/products/Planning/bin*.

► To launch ImportFormDefinition:

1 From the bin directory, enter the command using this syntax:

```
ImportFormDefinition [-f:passwordFile] /A:appname /U:username /
F:formname /D:filename [/AR] [/AC] [/SR] [/SC] [/KC] [/KR]
```

Table 54 ImportFormDefinition Utility Parameters

Setting	Purpose	Required?
<code>[-f:passwordFile]</code>	If an encrypted password file is set up, you can use this option as the first parameter in the command line to run the utility with the full file path and name specified in <i>passwordFile</i> . See “Suppressing Password Prompts in Planning Utilities” on page 37.	No
<code>/A</code>	Application name.	Yes
<code>/U</code>	Administrator user name.	Yes
<code>/F</code>	Data form name.	Yes
<code>/D</code>	Name and location of the Essbase Column Export Format data file. The location can be the full path and file name, or any format required for the operating system to find the file.	Yes
<code>/AR</code>	Add rows from the data file (on by default). Disable by specifying <code>/-AR</code> . For example, you can define rows in Planning and import only column definitions.	No
<code>/AC</code>	Add columns from the data file (on by default). Disable by specifying <code>/-AC</code> .	No
<code>/KC</code>	Keep member selections for columns in the data form (on by default). Clear columns by specifying <code>/-KC</code> . Data forms must have at least one column definition. If you clear columns but do not add them, the data form is not saved, and an error displays.	No
<code>/KR</code>	Keep member selections for rows in the data form (on by default). Clear member selections in rows by specifying <code>/-KR</code> . Data forms must	No

Setting	Purpose	Required?
	have at least one row definition. If you clear rows but do not add them, the data form is not saved, and an error displays.	
/SR	Sort rows in the data form (on by default). Disable by specifying /-SR.	No
/SC	Sort columns in the data form (on by default). Disable by specifying /-SC.	No

2 If prompted, enter your password.

For example:

```
ImportFormDefinition /A:MyPlan /U:Admin /F: "My Budget" /
D:exportfilename /AR /-AC
```

Creating the data form can take some time, depending on the amount of data in the file.

`ImportFormDefinition` imports the definition to the Planning data form, ensuring that all cells having data in the data file are represented on the data form. For cells in the data file that contain #MISSING, rows or columns are not added to the data form.

Notes:

- If you disable importing rows or columns, `ImportFormDefinition` filters imported data by rows or columns defined on the data form.
- If you run `ImportFormDefinition` more than once, it merges new results with the existing data form definition, and, if you also specify sorting, sorts new and existing rows or columns by dimension order.

ImportFormDefinition Examples

Define members on each axis of the data form appropriately and define import options correctly because this affects which data is imported.

To import only rows that contain data for the specified columns:

1. In Planning, specify the data form columns (for example, Descendants Inclusive of YearTotal).
2. For the dimension for which to import members to the row, add the dimension root to the data form design.

For example, to load accounts on the row, place the Account dimension root on the data form's row.

3. When you run `ImportFormDefinition`, use these options: `/AR /-AC`

Accounts are loaded from the data file if members from each dimension making up a cell match members on the data form columns, pages, and POV. The added rows are filtered by members

on the page. For example, if you place some members on the page, only accounts that contain data for those members are added to the data form. Rows are filtered by the members on the POV. If the data file contains data for the salary account for 2008, but only 2009 is on the POV, the salary account is not added to the row although it exists in the data file.

Importing and Exporting Data Forms

Administrators can use `FormDefUtil.cmd` (Windows) or `FormDefUtil.sh` (UNIX) to move data form definitions between Planning applications. You can export or import data form definitions to or from an XML file, useful for moving from a development to a production environment.

The utility uses a command line interface and, by default, is installed in: `HYPERION_HOME/products/Planning/bin`.

► To launch the `FormDefUtil` utility:

1 Enter the command from the `bin` directory, using this syntax:

```
formdefutil [-f:passwordFile] import|export filename|formname|-all server
name user name application
```

Parameter	Purpose	Required?
<code>[-f:passwordFile]</code>	If an encrypted password file is set up, you can use this option as the first parameter in the command line to run the utility with the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37 .	No
<code>import export</code>	Import or export the data form definition.	Yes
<code>filename formname -all</code>	When used with <code>import</code> , specify the XML file containing the data form definition. When used with <code>export</code> , specify the data form to export to XML. Use <code>-all</code> with <code>import</code> or <code>export</code> to import or export all XML files or data form definitions in the current application.	Yes. <code>-all</code> is optional.
<code>server name</code>	Server name on which the Planning application resides.	Yes
<code>user name</code>	Administrator's name.	Yes
<code>application</code>	When used with <code>export</code> , the name of the Planning application containing the data form definitions to export. When used with <code>import</code> , the name of the Planning application to which to import the data form definition.	Yes

2 If prompted, enter your password.

When you export data form definitions, the utility creates an XML file in the current directory and logs errors in `FormDefUtil.log` in the directory from which you run the utility (by default, `bin`). You can copy the utility to any directory and launch it from there to save files to another directory.

Examples:

- To import one file:

```
FormDefUtil.cmd import c:\Hyperion\products\Planning\bin\form1.xml
localhost admin APP1
```

- To export one file:

```
FormDefUtil.cmd export Form1 localhost admin APP1
```

- To export all data form definitions:

```
FormDefUtil.cmd export -all localhost admin APP1
```

- To import all data form definitions:

```
FormDefUtil.cmd import -all localhost admin APP1
```

Planning Offline Considerations

Offline Planning allows users to take data forms offline and work while disconnected from the server. Note these important considerations.

Offline Data Form Considerations

- Data forms used offline should be self-contained.

To ensure that values calculated offline are correct, all necessary dependencies pertinent to data forms taken offline must be available offline. All members, data forms, member formulas, business rules, and dynamic calculations on which a data form depends for accurate results offline must also be taken offline.

- Data not downloaded from the server is not available offline.
- Dependent data required for business rules to run offline must be taken offline.

Offline User Considerations

Considerations for users working on offline data forms:

- Although offline users can select Sync Back To Server, offline users must have write permission for a cell to save the changed cell value to the server.
- If a data form member is deleted on the server while a user is working with that data form and member offline, the offline user's changes to that member are not saved to the server when Sync Back To Server is selected.

- If more than one user modifies the same data, the last values synchronized back are saved. To prevent data loss or miscalculations, use planning units and process management on the server to control data access.
- Offline users can enter data in multiple currencies just as online users can. However, currency conversion is currently not supported when working offline. When users change currencies offline, values are not recalculated and displayed in the new currency.
- Leave the data form property Enable Offline Usage set when a data form is used offline, which allows users to save data changed offline back to the server.
- Use planning units to prevent two users from working with a data form simultaneously. An offline user can lose access to members taken offline if an online user working with the same data form clears the values in a row or column that has Suppress Missing Data set.
- Smart View does not currently support hidden data forms; they are not downloaded when users take data forms offline that are associated with hidden forms.
- Smart View does not currently support composite data forms; they are not listed when users select data forms to take offline.

Business Rule Considerations for Offline Calculations

Considerations for running business rules offline:

- Business rule calculations can use only data and objects taken offline; rules do not have access to data or objects stored on the server.
- Runtime prompts are not supported for business rules with the Run on Load property set.
- Business rules set to Run on Save that require user input for a runtime prompt are not supported when synchronizing back to the server.
- The Run on Load and Run on Save options can use only business rules and dependent data available offline.

7

Managing the Budgeting Process

In This Chapter

About the Budgeting Process	147
Overview of Managing Task Lists	149
Managing Task Lists	149
Copying Data	160
Clearing Cell Details	161

About the Budgeting Process

You can track budgets and review status, process issues, and planning unit ownership using planning units. Budget cycle time is reduced:

- Approval path is independent of organizational structure
- Exceptions and problem areas are highlighted
- Audit information includes annotations and process status
- Reviews include annotations and comments

Planning Units

Planning units are combinations of scenario, version, and entity. Scenarios and versions are the basis of the review cycle. Entities submit planning data for a scenario and version. For example, a planning unit might consist of a version (Best Case), an entity (New York), and a scenario (Actual).

Starting the Review Process

The review process begins at the lowest level of the planning unit hierarchy. The owner of that level completes the assigned tasks. Once the tasks for that level of the hierarchy are done, the current owner promotes the budget, which alerts the owner of the next highest level that the budget requires her attention. The planning unit moves from one level of the planning unit hierarchy to the next level above, until the budget process is complete. Throughout the process, the owner of each stage of the budget process can return the budget to the previous level's owner for any necessary changes.

► To start planning units:

- 1 Select **File**, then **Process Management**, then **Manage Process**.
- 2 For **Scenario**, select a scenario.
- 3 For **Version**, select a version.

The scenarios and versions displayed are enabled for process management.

- 4 Click **Go**.
- 5 Select **Start** by the entity to start.

For budget administrators, entities can be displayed as a tree or a flat list. In tree view, you can expand the hierarchy. In flat list view, click a column header to sort the list.

- 6 **Optional:** To remove planning units from the planning process or from being tracked in the system, select **Exclude**.

After administrators exclude planning units, all associated annotations and history are discarded. Planning unit status is returned to Not Started and the owner is set to No Owner. Data values are retained. For information on working with planning units, see the *Oracle Hyperion Planning User's Online Help*.

Entity Hierarchy

Parent/child relationships between entities affect the review process:

- When you promote or reject a parent, its children are promoted or rejected unless they are Approved. The owner for the parent becomes the owner of the children.
- When you approve a parent, its children are approved.
- After all children are promoted to the same owner, the parent is promoted to the owner.
- After all children are signed off, parent status changes to Signed Off.

You cannot change the status of a parent if its children have different owners. If the children are promoted to or signed off by different users, the parent has no owner and only budget administrators can change its status.

Setting Up E-mail for Workflow Notification

Application owners must specify an application's e-mail server before others can enable e-mail notification. After e-mail notification is configured, users receive e-mail when they become owners of planning units. The subject of the e-mail is New Owner: Plan (Scenario, Version, Entity). This feature is available for SMTP e-mail systems only; to specify the e-mail server, see [“Specifying System Settings” on page 166](#).


If the e-mail server name is valid, the application owner becomes the source of all e-mail notifications. If the e-mail server name is not valid, nobody can enable e-mail notification.

Printing Planning Unit Annotations

Administrators can check planning unit status by reporting on annotations for a set of scenarios, versions, and entities. Reports can be based on process status. The planning unit title, author, date, and annotations are displayed. Annotation text displays chronologically, with the most recent entry first.

For Thai language systems, time stamps in annotations and audit reports display in the Gregorian calendar, due to a limitation in the JDK (Java Development Kit) 1.3.1 for application servers.

► To create and print reports on planning unit annotations:

- 1 Select **Administration**, then **Reporting**.
- 2 Select **Planning Unit Annotations**.
- 3 Click  to select Scenarios, Versions, and Entities.
- 4 Select status states.
- 5 Click **Create Report**.

Adobe Acrobat generates a report with this planning unit information:

- Application name
- Selected Scenarios, Versions, and Entities
- Planning unit title and status
- Origin date
- Author
- Annotation content

- 6 Click **Print**  on the Adobe Acrobat toolbar.

Overview of Managing Task Lists

Task lists guide users through the planning process by listing tasks, instructions, and due dates. Administrators and interactive users create and manage tasks and task lists.

Managing Task Lists

Task lists contain groups of tasks.

► To manage task lists:

- 1 Select **Administration**, then **Manage Task Lists**.
- 2 Use the **Manage Task Lists** page to configure task list folders and create and manage task lists.

Creating Task List Folders

- To create task list folders:
 - 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
 - 2 In the **Task List Folders** area, click the folder in which to create the task list folder.
 - 3 Click **Create**.
 - 4 In the dialog box, enter the name of the task list, and click **OK**.

Renaming Task Lists

- To rename task lists:
 - 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
 - 2 In the **Task List Folders** area, click the folder with the task list to rename.
 - 3 Select the task list.
 - 4 Click **Rename**.
 - 5 Enter the new task list name, then click **OK**.

Renaming Task List Folders

- To rename task list folders:
 - 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
 - 2 In the **Task List Folders** area, select the folder to rename.
 - 3 Click **Rename**.
 - 4 Enter the new Task List name, then click **OK**.

Moving Task List Folders

- To move task list folders:
 - 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
 - 2 In the **Task List Folders** area, select the folder to move.
You cannot move the Task Lists folder.

- 3 Above the **Task List Folders** area, click **Move**.
- 4 Select the destination folder, and click **OK**.

Deleting Task List Folders

► To delete task list folders:

- 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
- 2 In the **Task List Folders** area, select an empty folder to delete.
You cannot delete the Task Lists folder.
- 3 Click **Delete**.
- 4 Click **OK**.

If you select a folder containing additional folders, an error message is displayed.

Creating Task Lists

Task lists organize groups of tasks for users. You must create task lists before creating tasks.

► To create task lists:

- 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
- 2 In the **Task List Folders** area, click the folder in which to create the task list.
- 3 Above the **Task List** area, click **Create**.
- 4 Enter the task list name, and click **OK**.
- 5 To define the task list, see:
 - [“Adding Instructions to Task Lists” on page 151](#).
 - [“Adding and Defining Tasks” on page 152](#).
 - [“Assigning Access to Task Lists” on page 157](#).

Adding Instructions to Task Lists

► To add instructions:

- 1 Open the **Manage Task Lists** page.
See [“Managing Task Lists” on page 149](#).
- 2 In the **Task List Folders** area, click the folder containing the task list to modify, and select the task list.

- 3 For **Task List**, select the task list to modify, and click **Edit**.
- 4 For **Edit Task List**, select **Instructions**.
- 5 Enter instructions for the task list.
- 6 Click **Save** and **Close**.

Adding and Defining Tasks

After creating task lists, you can add and define tasks such as entering data in data forms and running required business rules. See [“Adding Tasks to Task Lists” on page 152](#) and [“Setting Task Properties” on page 153](#).

Adding Tasks to Task Lists

You can set completion dates and alerts for tasks. Alerts display on task lists as colored circles:

- Green: On schedule
- Yellow: approaching due date
- Red: overdue

You can also set up e-mail messages, for example, to alert users that a task was not completed by its due date. Alert messages are sent after an “alert date” that you set, and are repeated until the due date is reached for a task. You must configure an e-mail server (see [“Specifying System Settings” on page 166](#)).

► To add tasks to task lists:

- 1 Select **Administration**, then **Manage Task Lists**, select a task list, and click **Edit**.
- 2 Click **Add Child**.
- 3 For **Task**, enter a task name.
- 4 For **Type**, select the task type:
 - URL Task: Opens a specified URL
 - Web Data Form: Opens a data form
 - Business Rule: Launches a business rule that you specify
 - Workflow: Starts the review process with a specified scenario and version
 - Descriptive: Descriptive information
- 5 For **Instructions**, enter information that explains how to complete the task.
- 6 **Optional:** To enter a due date for the task, select **Due Date** and select:
 - a. The month, day, and year. (You can change the date display format in Planning preferences. See [“Specifying System Settings” on page 166](#).)
 - b. The hours, minutes, and AM or PM.
- 7 **Optional:** To send e-mail message for uncompleted tasks, select **Due Date**:

- a. Select **Repeat Every**, and enter a number.
 - b. Select the frequency for e-mail reminders.
- 8 Optional:** To send e-mail messages after the alert date and before the due date, select **Due Date**:
- a. Set the date and time to begin sending messages by selecting the month, day, year, hours, minutes, and AM or PM.
 - b. In the Alert area, select Repeat Every and enter a number.
 - c. Select the frequency for e-mail reminders.
- 9 Optional:** To make task completion dependent on completing a primary task, select **Dependency**.
- 10** Click **Save** and **Close**.
- 11** Set properties (see [“Setting Task Properties” on page 153](#)).

Setting Task Properties

You must set properties for all task types except Descriptive tasks.

► To set task properties:

- 1 Select **Administration**, then **Manage Task Lists**, select a task list, and click **Edit**.
- 2 Select a task, and click **Edit**.
- 3 Click **Property**.
- 4 In **Properties**:

Table 55 Setting Task Properties

Type of Task	Action
URL	Enter a fully qualified URL to associate with this task, such as <code>http://www.company_name.com</code> . Optionally, select Use Single Sign On to enable users to open a URL for another product that accepts single sign-on (see the <i>Oracle Hyperion Enterprise Performance Management System Security Administration Guide</i>). To link to Oracle Hyperion Financial Reporting, Fusion Edition in the EPM Workspace, single sign-on is not required. Instead, include the ObjectID to link to (see the <i>Oracle Enterprise Performance Management Workspace Developer's Guide</i>).
Web data form	For Data Form Folder, select the folder containing the data form associated with this task. For Data Form, select the data form for users to complete.
Business rule	For Plan Type, select the plan type associated with the business rule to execute. For Business Rule Name, select the business rule to execute.
Workflow	For Scenario and Version, select the scenario and version users will work in.

For Descriptive tasks, no updates are required.

- 5 Click **Save**.
- 6 Click **Close**.

Editing Task Lists

You can use the Edit Task List dialog box to update task lists. See:

- “Editing Tasks” on page 154
- “Copying Tasks” on page 155
- “Moving Tasks” on page 155
- “Moving Task Lists” on page 155
- “Reordering Task Lists” on page 156
- “Deleting Tasks” on page 156
- “Deleting Task Lists” on page 156

Editing Tasks

Use the Edit Task dialog box to modify the type of task, its instructions, the due date and alert date, and e-mail reminder messages.

► To edit tasks:

- 1 Select **Administration**, then **Manage Task Lists**, select a task list, and click **Edit**.
- 2 Select a task, and click **Edit**.
- 3 In **Edit Task**:
 - Modify the task name.
 - Select another task type:
 - URL Task: Opens a specified URL
 - Web Data Form: Opens a data form
 - Business Rule: Launches a specified business rule
 - Workflow: Starts the review process with a specified scenario and version
 - Descriptive: Describes an action that users must take
- 4 For **Instructions**, modify instructions for completing the task.
- 5 **Optional**: To enter a due date for the task, select **Due Date**, and enter the date and time by which users must complete this task.
- 6 **Optional**: To send e-mail messages if tasks are not completed by the due date, select **Due Date**.
 - a. Select Repeat Every and enter a number.
 - b. Select the frequency for e-mail reminders.
- 7 **Optional**: To send e-mail messages after the alert date and until the due date, select **Due Date**:
 - a. Select the month, day, year, time, and AM or PM.
 - b. Select Repeat Every and enter a number.
 - c. Select the frequency for e-mail reminders.

- 8 **Optional:** To make completion of this task depend on completing a primary task, select **Dependency**.
- 9 **Optional:** To edit task properties, click **Property**. See [“Setting Task Properties” on page 153](#).
- 10 Click **Save**.
- 11 Click **Close**.

Copying Tasks

- To make a copy of a task list:
 - 1 Select **Administration**, then **Manage Task Lists**, select a task list, and click **Edit**.
 - 2 Select the task list to copy.
 - 3 Click **Save As**.
 - 4 Enter the name of the task list, and click **OK**.
 - 5 Click **Close**.

Moving Tasks

- To move tasks:
 - 1 Select **Administration**, then **Manage Task Lists**, select a task list, and click **Edit**.
 - 2 Select a task, then click the up or down arrow.
 - 3 Click **Save** and **Close**.
- To cut and paste tasks:
 - 1 Select **Administration**, then **Manage Task Lists**, select a task list, and click **Edit**.
 - 2 Select a task, then click **Cut**.
 - 3 To move the task to a new position, select the task to appear above it.
 - 4 Click **Paste**.
 - 5 Click **OK**.



Moving Task Lists

- To move task lists:
 - 1 In **Manage Task Lists**, select the folder with the task list to move.
See [“Managing Task Lists” on page 149](#).
 - 2 Select the task list.
 - 3 Click **Move**.
 - 4 Select the destination folder.

- 5 Click **OK**.

Reordering Task Lists

- To reorder task lists:

- 1 In **Manage Task Lists**, click the folder with the task list, and select the task list.
See “[Managing Task Lists](#)” on page 149.
- 2 Click  or .

Deleting Tasks

- To delete tasks:

- 1 In **Manage Task Lists**, click the name of the folder with the task.
See “[Managing Task Lists](#)” on page 149.
- 2 Select the task list that contains the task.
- 3 Select the task list with the task to delete.
- 4 Click **Edit**.
- 5 Select tasks to delete, and click **Delete**.
- 6 Click **OK**.

Deleting Task Lists

- To delete task lists:

- 1 In **Manage Task Lists**, click the folder with the task to delete, and select the task list.
See “[Managing Task Lists](#)” on page 149.
- 2 Select task lists to delete, and click **Delete**.
- 3 Click **OK**.

Linking Tasks to Planning Web Pages

Use Copy Document Link to link tasks in task lists to application pages. You can copy and paste the URL address from a page in the Planning application to a task. The task list user can access the Planning page within the task list. See “[Creating Task Lists](#)” on page 151.

- To copy URLs to task lists:

- 1 Access the page in the Planning application to which to link a task.
- 2 Select **Administration**, then **Copy Document Link**.

- 3 Open the task list to which to link a task.
- 4 Select the task to which to link the copied Planning page and select **Edit**.
- 5 Click **Properties**.
- 6 From the Microsoft Internet Explorer menu, select **Edit**, then **Paste**.
- 7 Click **Save**.
- 8 Click **Close**.

Assigning Access to Task Lists

You can determine who can view and modify task lists. By default, administrators can manage and assign access permissions for task lists. Interactive users and planners can access task lists in Basic mode based on access permissions.

Note: Being assigned to a task list means being able to access and complete tasks in the task list. It does not mean being able to assign tasks to someone else.

Adding Access to Task Lists

- To assign access to task lists:
- 1 In **Manage Task Lists**, select the task list to modify.
See [“Managing Task Lists” on page 149](#).
 - 2 In the **Task List** area, click **Assign Access**.
 - **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
 - **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.
 - 3 Click **Add Access**.
 - 4 Select the users or groups to access the task list.
 - Click **Users** to display all user names; click **Groups** to display all groups.
 - If there are multiple pages of users and groups, type the page number to go to in **Page**, and click **Go**.
 - Click **Start** or **End** to navigate to the first or last page.
 - Click **Prev** or **Next** to move to the previous or next page.
 - 5 For **Type of Access**, select how users or groups can use the task list:
 - **Assign:** View and use
 - **Manage:** Modify
 - **Manage and Assign:** View, use, and modify

- None: No access
- 6 Click **Add**.
 - 7 Click **Close**.

Changing Access to Task Lists

► To change access to task lists:

- 1 For **Manage Task Lists**, select the folder and task list to modify.

See [“Managing Task Lists” on page 149](#).

- 2 Click **Assign Access**.

- **Optional:** To migrate a user or group's changed identity or their position in the user directory from Shared Services Console to Planning, click **Migrate Identities**.
- **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.

- 3 Select users or groups, and click **Edit Access**.

- 4 For **Type of Access**, select:

- Assign: View and use
- Manage: Modify
- Manage and Assign: View, use, and modify
- None: No access

- 5 Click **Set**.

- 6 Click **Close**.

Removing Access to Task Lists

► To remove access to task lists:

- 1 On **Manage Task Lists**, select the folder and task list to modify.

See [“Managing Task Lists” on page 149](#).

- 2 Click **Assign Access**.

- **Optional:** To migrate a user or group's changed identity or their position in the user directory from Oracle's Hyperion® Shared Services Console to Planning, click **Migrate Identities**.
- **Optional:** To remove deprovisioned or deleted users or groups from the Planning database to conserve space, click **Remove Non-provisioned Users/Groups**.

- 3 Select the user or group, and click **Remove Access**.

- 4 Click **OK**.

- 5 Click **Close**.

Importing and Exporting Task Lists

Administrators can use `TaskListDefUtil.cmd` (Windows) or `TaskListDefUtil.sh` (UNIX) to move task list definitions between Planning applications. You can export or import task list definitions to or from an XML file.

`TaskListDefUtil` uses a command line interface and, by default, is installed in:
`HYPERION_HOME/products/Planning/bin`.

► To launch the `TaskListDefUtil` utility:

1 Enter the command from the `bin` directory, using this syntax:

```
TaskListDefUtil [-f:passwordFile] import|exportFILE_NAME/  
TASK_LIST_NAME|-all SERVER_NAME USER_NAME APPLICATION
```

Parameter	Purpose	Required?
<code>[-f:passwordFile]</code>	If an encrypted password file is set up, you can use this option as the first parameter in the command line to run the utility with the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37 .	No
<code>import export</code>	Import or export the task list definition.	Yes
<code>FILE_NAME TASK_LIST_NAME -all</code>	When used with <code>import</code> , specify the XML file containing the task list definition. When used with <code>export</code> , specify the task list to export to XML. Use <code>-all</code> with <code>import</code> or <code>export</code> to import or export all XML files or task list definitions in the current application.	Yes (<code>-all</code> is optional)
<code>SERVER_NAME</code>	Server name on which the Planning application resides.	Yes
<code>USER_NAME</code>	Administrator's name.	Yes
<code>APPLICATION</code>	When used with <code>export</code> , the name of the Planning application containing the task list definitions to export. When used with <code>import</code> , the name of the Planning application to which to import the task list definition.	Yes

2 If prompted, enter your password.

When you export task list definitions, the utility creates an XML file in the current directory and logs errors in `TaskListDefUtil.log` in the directory from which you run the utility (by default, `bin`). You can copy the utility to any directory and launch it from there to save files to another directory.

Examples:

- To import one file:

```
TaskListDefUtil.cmd import C:\Hyperion\products\Planning\bin
\TaskList1.xml localhost admin APP1
```

- To export one file:

```
TaskListDefUtil.cmd export TaskList1 localhost admin APP1
```

- To export all task list definitions:

```
TaskListDefUtil.cmd export -all localhost admin APP1
```

- To import all task list definitions:

```
TaskListDefUtil.cmd import -all localhost admin APP1
```

Copying Data

You can copy plans from one dimensional intersection to another, including relational data and supporting detail. For example, you can copy *Budget, FY07, Final* to *Forecast, FY08, First Draft*.

Notes:

- Selected Copy Data settings are preserved for the current session only.
- Copied dimension members must be present in the selected plan types.
- Data must be copied into cells that can accept data. For example, you cannot copy data into read-only or dynamic cells.
- You can copy account annotations, supporting detail, and cell text. You cannot copy planning unit annotations.
- You cannot use this feature with attributes, so do not select attributes to be copied.
- Essbase data is copied regardless of selections for Copy Data Options.
- Because this is an administrative function, Planning assumes you have full access to data you copy. You are not prevented from copying to planning units that are approved.
- This feature does not calculate data. To perform calculations, such as increasing the forecast by 5%, apply the business rule after copying data.
- For Copy Data to be successful, you must select at least one member for Scenario, Account, Entity, Period, and Version dimensions.

➤ To copy data:

- 1 Select **Administration**, then **Copy Data**.
- 2 For **Plan Type**, select a plan type, and click **Go**.

You can copy from one plan type at a time. When you click Go, appropriate dimensions are displayed for this plan type.

- 3 For **Static Dimensions**, enter the members for the data intersections:
 - a. For **Dimension**, select a dimension from which to copy.

- b. For **Members**, click **Member Selection** to make a selection from which to copy. You can select multiple members. You must select at least one member for Scenario, Account, Entity, Period, and Version dimensions.
 - 4 Optional:** To add another static dimension to the list, click **Add Dimension** and enter dimension members. (To remove a dimension, select **None - Select a Dimension**. The dimension moves to the **Dimensions with Source and Destination** area.)
 - 5** For **Dimensions with Source and Destination**, enter dimensions into which to copy data:
 - a. For **Source**, click **Member Selection**
 - b. For **Destination**, click **Member Selection**.
 - 6** For **Copy Data Options**, select the type of information to copy.
 - 7** Click **Copy Data**.
- Data is copied from one intersection to the other. If data is not copied successfully, a message displays. You can also check the log file.
- 8** To copy data for another plan type, select another plan type in [step 2](#), and repeat the procedure.

Tip: To view the execution status of Copy Data, see “Checking Job Status” in the *Oracle Hyperion Planning User's Online Help*.

Clearing Cell Details

You can clear these cell details for a plan type: account annotations, supporting detail, cell text, and cell-level documents. For information on creating and viewing account annotations, supporting detail, cell text, and cell-level documents, see the *Oracle Hyperion Planning User's Online Help*.

Notes:


- You cannot delete planning unit annotations.
- Because this function is for administrators and interactive users, Planning assumes you have full access to details you delete.
- Oracle recommends that you back up the application before performing this procedure. See the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.
- You can also clear cell details with SQL scripts. See “[Deleting Application Information Using SQL](#)” on page 169.

➤ To clear cell details:

- 1** Select **Administration**, then **Clear Cell Details**.
- 2** For **Plan Type**, select a plan type, and click **Go**.

When you click Go, appropriate dimensions are selectable for this plan type.

- 3** Select members for the data intersections:

- a. For **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. See “[Selecting Members](#)” on page 117.

Note: For every dimension selected, you must select at least one member. If a dimension is not selected, Planning includes all its members when clearing cell details.

4 Optional: Further refine the data intersection by specifying more members:

- To select another dimension so you can select its members, click **Add Dimension**.
- To select all dimensions in the plan type, click **Add All Dimensions**.

Select members for the displayed dimensions.

5 Specify the type of information to delete by selecting at least one option from **Clear Options: Account Annotations, Supporting Details, Cell Text, or Cell-level Document.**

6 Click **Clear.**

A Confirmation page displays your selections.

7 Click **Finish to proceed, or **Back** to change your selections.**

If Clear Cell Details is successful, data is deleted from the plan type. If data is not deleted successfully, a message displays. You can also check the log file.

8 Optional: To view the execution status of Clear Cell Details and review the information that was deleted, select **Tools, then **Job Console**.**

See “Checking Job Status” in the *Oracle Hyperion Planning User's Online Help*.

9 To clear cell details for another plan type, select another plan type in [step 2](#), and repeat the procedure.



Working With Applications

In This Chapter

Managing Data Forms and Folders	163
About Setting Preferences	165
Deleting Application Information Using SQL	169

Managing Data Forms and Folders

Use the Data Form Management and the Business Rule Folders pages to manage folders and data forms.

Task	Topic
Create folders	See “Creating Folders” on page 163
Move folders	See “Moving Folders” on page 164
Delete folders	See “Deleting Folders” on page 164
Create data forms	See “Creating Data Forms” on page 110
Assign access to data forms and folders	See “Assigning Access to Data Forms and Folders” on page 46
Move data forms	See “Moving Data Forms” on page 139
Delete data forms	See “Deleting Data Forms” on page 139
Rename folders	See “Renaming Folders” on page 165

To view all data forms or business rules in a Calculation Manager folder, click the folder’s name in the lefthand folders area. To select all the data forms, select the check box at the top of the forms list.

Creating Folders

Use folders to hierarchically organize data 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 cannot:

- Delete folders unless they are empty

- Select multiple folders
 - Rename, move, or delete the top-level folder called respectively, Data Forms and CalcMgrRules, which contain the application's folders and forms
- To create folders:
- 1 For data form folders: Select **Administration**, then **Manage Data Forms**.
For Calculation Manager business rule folders: Select Administration, then Business Rule Security.
 - 2 Select the folder under which to create the folder.
 - 3 Above the folders list, click **Create**.
 - 4 Enter the folder name.
 - 5 Click **OK**.

Moving Folders

When you move folders, all nested folders, data forms, and Calculation Manager business rules within them are also moved.

- To move folders:
- 1 For data form folders: Select **Administration**, then **Manage Data Forms**.
For business rule folders: Select Administration, then Business Rule Security.
 - 2 Select the folder to move.
 - 3 Click **Move**.
 - 4 Select the destination folder to which to move the selected folder.
 - 5 Click **OK**.

Deleting Folders

- To delete folders:
- 1 For data form folders: Select **Administration**, then **Manage Data Forms**.
For Calculation Manager business rule folders: Select Administration, then Business Rule Security.
 - 2 Select the folder to delete.
 - 3 Click **Delete**.
 - 4 Click **OK**.

Renaming Folders

► To delete folders:

1 For data form folders: Select Administration, then Manage Data Forms.

For Calculation Manager business rule folders: Select Administration, then Business Rule Security.

2 Select the folder to rename.

3 Click Rename.

4 Click OK.

About Setting Preferences

On the Preferences page, all users can set individual preferences. Administrators and application owners can specify global settings. Preference selections affect only the current application.

Preference options depend on user type. Planners and interactive user types can access Application Settings and Display Options. See the *Oracle Hyperion Planning User's Online Help*.

See:

- [“Setting Personal Preferences” on page 166](#)
- [“Setting Application Defaults” on page 166](#)
- [“Specifying System Settings” on page 166](#)
- [“Limiting Use of Applications During Maintenance” on page 168](#)
- [“Specifying Custom Tools” on page 169](#)
- [“Setting Display Options” on page 169](#)
- [“Setting Printing Options” on page 169](#)

Administrators can control which tabs display for setting personal preferences, application defaults, and system settings.

Show Option	Description
Current Application Defaults	Accesses the Application Settings tab and Display Options tab. The values set on these tabs become application defaults. Users can override defaults, and can revert to defaults by selecting Use Application Defaults on preference tabs. See the <i>Oracle Hyperion Planning User's Online Help</i> .
Advanced Settings	Accesses the System Settings tab and Custom Tools tab. See “Specifying System Settings” on page 166 and “Specifying Custom Tools” on page 169 .

Setting Personal Preferences

All users can set personal preferences in the current application. For Application settings, Display Options, and Printing Options, see the *Oracle Hyperion Planning User's Online Help*.

► To set personal preferences:

- 1 **Select File, then Preferences.**
- 2 **Click Planning, then perform an action:**
 - Select **Application Settings** to set e-mail options, select an alias table, and set options for member selection and workflow.
 - Select **Display Options** to set options for number formatting, page selection, warnings for large data forms, and the number of dimensions to show on a page.
 - Select **Printing Options** to specify how pages are printed.
 - Select **User Variables Options** to limit the number of members that display on a data form by setting a user variable. See [“Managing User Variables” on page 136](#).

Selecting Use Application Default resets the value to the current application default.

Setting Application Defaults

Administrators can specify defaults for the current application. Users can override application defaults by setting preferences. They can restore application defaults by selecting Use Application Default where available. For Application Settings and Display Options, see the *Oracle Hyperion Planning User's Online Help*.

► To set application defaults:

- 1 **Select Administration, then Application Settings.**
- 2 **For Show, select Current Application Defaults.**
- 3 **Click Go.**
- 4 **Perform one action:**
 - Select **Application Settings** to set e-mail options, select an alias table, and set options for member selection, workflow, and attribute dimension date format.
 - Select **Display Options** to set options for number formatting, page selection, warnings for large data forms, and the number of dimensions to show on a page. Another option sends you directly to the page you most recently visited, the next time you log on.
- 5 **Click Save.**

These settings become application defaults, which users can select with Use Application Defaults.

Specifying System Settings

Only administrators can specify application-wide settings.

- To specify system settings:
 - 1 Select **Administration**, then **Application Settings**.
 - 2 For **Show**, select **Advanced Settings**.
 - 3 Click **Go**.
 - 4 Select **System Settings**.
 - 5 Set options:

Table 56 System Settings

Option	Description
E-mail Server	The server name that hosts e-mail services for application users, such as <code>mail.oracle.com</code>
E-mail Character Set	The character set for e-mail messages: UTF-8: Unicode encoding format Regional Setting : The system's regional setting
Shared Services URL	The URL for the Shared Services server. Click Register Shared Services and assign the application to a project (see “Assigning Applications to Shared Services Projects” on page 168).
Enable Use of the Application for	Determine whether users can access the application in maintenance mode, such as during backups. See “Limiting Use of Applications During Maintenance” on page 168 .
Enable Display of Substitution Variables	Set how substitution variables display in the Member Selection dialog box when users respond to runtime prompts in business rules: <ul style="list-style-type: none"> ● Display All: Display all substitution variables ● Display None: Do not display substitution variables ● Enable Filtering: Display only substitution variables that are valid for the runtime prompt
Select User	Assign an administrator as the application owner. (By default, the person who creates the application is the application owner. The owner may grant ownership to another administrator.)
Display Users' Full Names	Yes: Show full names (such as Victoria Hennings) and user names (such as VHennings). No: Do not display. With this selected, two users cannot have identical full names.
Calculation Module	<p>For Classic applications only: Select whether to use Business Rules or Calculation Manager as the calculation module.</p> <ul style="list-style-type: none"> ● If, after creating or modifying business rules, you switch from Business Rules to Calculation Manager, you can migrate the business rules through the Calculation Manager user interface. ● If, after creating or modifying business rules, you switch from Calculation Manager to Business Rules, the rules are not accessible in Business Rules. You can, however, switch back to the Calculation Manager calculation module to use the rules. <p>Tip: To evaluate which calculation module you prefer before committing to it for multiple applications, first experiment with one calculation module on one application.</p> <p>Note: Performance Management Architect applications must use Calculation Manager, so for those applications, the Calculation Manager option is selected and cannot be changed.</p>

- 6 To specify system settings for each application, repeat these steps, then click **Save**.

Assigning Applications to Shared Services Projects

- To assign a Planning application to a Shared Services project:
 - 1 Select **Administration**, then **Application Settings**.
 - 2 For **Show**, select **Advanced Settings**.
 - 3 Click **Go**.
 - 4 Select **System Settings**.
 - 5 Click **Register Shared Services**, then select an option for **Assign Application to Project**:
 - **New Project**. Enter the project name in the text box (available only if you have the Project Manager role in Shared Services).
 - **Existing Projects**. Select the project to which to assign the current application.
 - 6 Click **Submit** and **Save**.

Limiting Use of Applications During Maintenance

Administrators can grant and withdraw access to applications. If users are logged on to the application and administrators withdraw their access, users receive a message and are forced off the system. This is useful when administrators must perform system maintenance or backups, during which no users should be logged on. Restricting use of an application as described here has no effect on Smart View users.

- To grant or withdraw user access to applications:
 - 1 Select **Administration**, then **Application Settings**.
 - 2 For **Show**, select **Advanced Settings**.
 - 3 Click **Go**.
 - 4 Select **System Settings**.
 - 5 For **Application Maintenance Mode**, select an option for **Enable Use of the Application For**:
 - **All users**: All users can log on or continue working with the application.
 - **Administrator**: Only other administrators can log on. Other users are forced off and prevented from logging on until the option is reset to **All Users**.
 - **Owner**: Only the application owner can log on. All other users are prevented from logging on. If they are currently logged on, they are forced off the system until the option is reset to **All Users** or **Administrators**. Only the application owner can restrict other administrators from using the application.
 - 6 If your selection is more restrictive than the current setting, click **OK**.
 - 7 Click **Save**.

Specifying Custom Tools

Administrators can specify custom tools, or links, for users on the Tools page. Users having access to links can click links from the Tools menu to open pages in secondary browser windows.

- To specify custom tools:
 - 1 Select **Administration**, then **Application Settings**.
 - 2 For **Show**, select **Advanced Settings**.
 - 3 Click **Go**.
 - 4 Select **Custom Tools**.
 - 5 For each link:
 - For **Name**, enter the displayed link name.
 - For **URL**, enter a fully qualified URL, including the `http://` prefix
 - For **User Type**, select which users can access the link.
 - 6 Click **Save**.

Setting Display Options

Administrators can set the number of items that display on the dimensions page and Add Access pages.

- To set the number of items that display:
 - 1 Select **File**, then **Preferences**, and then **Display Options**.
 - 2 Enter values:
 - Show the Specified Members on Each Dimensions Page
 - Show the Specified Records on Each Assign Access Page
 - 3 Click **Save**.

Setting Printing Options

To set printing options, see [“Setting Other Options for Data Forms”](#) on page 125.

Deleting Application Information Using SQL

Planning provides SQL files to delete this information:

- Account annotations. See [“Deleting Account Annotations”](#) on page 170.
- Supporting detail associated with scenarios. See [“Deleting Supporting Detail Associated With a Scenario”](#) on page 170.

You can use the Clear Cell Detail feature to clear account annotations, supporting detail, cell text, and cell-level documents. See [“Clearing Cell Details” on page 161](#).

Deleting Account Annotations

Use the `aaddelete.sql` file, installed in the `sql` directory, to delete account annotations. It includes SQL queries that delete annotations for selected account names. To use the Clear Cell Detail feature to clear account annotations, see [“Clearing Cell Details” on page 161](#).

- To delete account annotations associated with account names:
 - 1 **Stop the Web application server.**
 - 2 **Update the SQL queries section of the `aaddelete.sql` file corresponding to your type of relational database by substituting the name of the account whose annotations to delete.**
 - 3 **Run queries in the `aaddelete.sql` file appropriate for your relational database.**

Example: deleting account annotations for `Account1`:

```
DELETE
FROM HSP_ACCOUNT_DESC
WHERE ACCOUNT_ID=(SELECT OBJECT_ID FROM HSP_OBJECT
WHERE OBJECT_NAME='ACCOUNT1')
INSERT INTO HSP_ACTION
(FROM_ID, TO_ID, ACTION_ID, OBJECT_TYPE, MESSAGE, ACTION_TIME,
PRIMARY_KEY) VALUES (0,0,2,18,NULL,GETDATE(),NULL)
```

Deleting Supporting Detail Associated With a Scenario

You can use the `sddelete.sql` file, installed in the `sql` directory, to delete supporting detail associated with scenarios. It includes SQL queries that delete supporting detail for selected scenarios. To use the Clear Cell Detail feature to clear supporting detail, see [“Clearing Cell Details” on page 161](#).

- To delete supporting detail associated with scenarios:
 - 1 **Stop the Web application server.**
 - 2 **Update the SQL queries section of the `sddelete.sql` file corresponding to your type of relational database by substituting the scenario name with the supporting detail to delete.**
 - 3 **Run queries in the `sddelete.sql` file appropriate for your relational database.**
 - 4 **Start the Web application server.**

Example: Deleting supporting detail associated with a scenario

Supporting detail for the Actual scenario is deleted:

```
DELETE
```

```
FROM HSP_COLUMN_DETAIL_ITEM
WHERE DETAIL_ID IN
    (SELECT DETAIL_ID
     FROM HSP_COLUMN_DETAIL
     WHERE DIM1 =
        (SELECT OBJECT_ID
         FROM HSP_OBJECT
         WHERE OBJECT_NAME = 'ACTUAL')));

DELETE
FROM HSP_COLUMN_DETAIL
WHERE DIM1 =
    (SELECT OBJECT_ID
     FROM HSP_OBJECT
     WHERE object_name = 'Actual');
```


9

Working with Menus

In This Chapter

Creating and Updating Menus	173
Working with Menu Items	173
Adding or Changing Menu Items	174

Creating and Updating Menus

Administrators can create right-click menus and associate them with data forms, enabling users to click rows or columns in data forms and select menu items to:

- Launch another application, URL, or business rule, with or without runtime prompts
- Move to another data form
- Move to Manage Process with a predefined scenario and version

The context of the right-click is relayed to the next action: the POV and the Page, the member the user clicked on, the members to the left (for rows), or above (for columns).

When designing data forms, use Other Options to select menus available for Data Form menu item types. As you update applications, update the appropriate menus. For example, if you delete a business rule referenced by a menu, remove it from the menu.

➤ To create, edit, or delete menus:

1 Select Administration, then Manage Menus.

2 Perform one action:

- To create a menu, click **Create**, enter the menu's name, and click **OK**.
- To change a menu, select it and click **Edit**.
- To delete menus, select them, click **Delete**, and click **OK**.

Working with Menu Items

Edit Menu displays menu items on the current menu, including names, labels, required dimensions, icon, and type: URL, Data Form, Business Rule, Workflow, or Menu Header.

- To work with menu items:
 - 1 Select **Administration**, then **Manage Menus**.
 - 2 Select a menu and click **Edit**.
 - 3 **First time only:** To add the first item to the menu, click **Add Child** and **Save**.
 - 4 Select a menu item and:
 - To add menu items below the selected item, click **Add Child** (available for Menu Header menu types).
 - To add menu items at the same level as the selected item, click **Add Sibling**.
 - To edit menu items, click **Edit**.
 - To delete menu items, click **Delete**.
 - To change the order of menu items within the same level, click the Up or Down Arrow. You can move multiple items.

Use Edit Menu Item to define the menu item properties.
 - 5 Click **Save**.

Click Save As to save the current selections under a new menu name.

Adding or Changing Menu Items

- To define menu items:
 - 1 Select **Administration**, then **Manage Menus**.
 - 2 Select a menu and click **Edit**.
 - 3 Select the menu item and click **Edit** or **Add Sibling**.
 - 4 Define the menu item:

Table 57 Menu Items

Item	Description
Menu Item	Enter a unique name containing only alphanumeric and underscore characters, with no special characters or spaces
Label	Enter text to display 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 <code>File</code> directly or to the name of a resource, such as <code>LABEL_FILE</code> , which can be localized.
Icon	Optional: In context of the Planning server, enter the path and filename to a graphic to display by the menu. For example: <code>/HyperionPlanning/Images/globe.gif</code>
Type	Select the menu item type to determine available Properties. No properties are available for Menu Header.

Item	Description
	<ul style="list-style-type: none"> ● Data Form: Launch a selected data form. The member selection context for the member, page, and POV is retained when users right-click in the source data form. If the target data form contains these dimension members on the page, its page is set to match the context. ● URL: Navigate to the specified URL ● Business Rule: Launch the selected business rule ● Workflow: Move to Manage Process to work with planning units ● Menu Header: Create a menu under which you can create children menu items. To display a separator bar on the menu at this item, enter one hyphen as the Label. In this case, the Required Dimension list is not available.
Required Dimension	Select the dimension for which the menu item displays. For example, if you select Account, users can right-click Account members on a data form to open the menu. Selecting No Required Dimension makes the menu available whenever the user right-clicks in the data form.

5 Define menu item properties, which differ for menu item types:

Type	Options
Data Form	<ol style="list-style-type: none"> a. From Data Form Folder, select the folder containing the destination data form. b. From Data Form, select the data form.
URL	<ol style="list-style-type: none"> a. For URL, enter the complete URL to which to direct the user. For example: <code>http://server name/HFM/Logon/HsvLogon.asp</code>. b. Select Use Single Sign-on to append the SSO token to the URL. c. Select Include Context in URL to include the context.
Business Rule	<ol style="list-style-type: none"> a. For Plan Type, select the plan type for which the business rule is available. b. For Business Rules, select the business rule to launch. c. From View Type, select how to display runtime prompt pages: <ul style="list-style-type: none"> ● Classic View: Use the default Planning view ● Streamline View: Display each runtime prompt on a different line d. Optional: For Window Title, enter a title to display instead of Runtime Prompts. e. Optional: For OK Button Label, enter the text to display for the OK button. f. Optional: For Cancel Button Label, enter the text to display for the Cancel button. g. Optional: Select Launch in a Separate Window to launch the business rule in a separate browser window.
Workflow	Specify the planning unit to which the user is directed by selecting a scenario and a version.

6 Click **Save**.

10

Working with Classic Application Administration

In This Chapter

About Creating Applications with Performance Management Architect and Classic Administration	177
Setting Up Classic Applications	177
Working with Alias Tables	185
Working with Dimensions	187
Setting up Dynamic Time Series Members	226
Working with Classic Applications	228
Using Smart Lists, UDAs, and Member Formulas	232

About Creating Applications with Performance Management Architect and Classic Administration

You can create applications using Performance Management Architect or Classic application administration. Different menus and options are available for each type of application. For example, for applications created with Performance Management Architect application administration, you manage Smart Lists within Performance Management Architect. For applications created with Classic administration, you can select Administration, then Smart Lists to use the feature within Planning. Classic application administration tasks are described in this chapter. To use Classic application administration, you must be assigned the appropriate roles, as described in the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.

Setting Up Classic Applications

For Classic Planning applications, use the Classic Application Wizard to create and delete applications, and register with Shared Services. When you create Classic applications, you can set them up to calculate business rules using Business Rules or Calculation Manager. After you set up the calculation, you can change the calculation module as described in “[Specifying System Settings](#)” on page 166.

► To set up Classic applications:

- 1 Perform one task:

- From Planning, select **Administration**, then **Create Application**, **Manage Data Source**, **Delete Application**, or **Register Application**.
 - From EPM Workspace, select **Navigate**, then **Administer**, then **Classic Application Administration**, and then **Planning Administration**. Then select **Create Application**, **Manage Data Source**, **Delete Application**, or **Register Application**.
- 2 See these topics to complete the task:
- [“Creating Applications” on page 178.](#)
 - [“Managing Data Sources” on page 178.](#)
 - [“Deleting Applications” on page 185.](#)
 - [“Registering Applications” on page 184.](#)

Creating Applications

► To create and update Classic applications:

- 1 Start the **Classic Application Wizard** (see [“Setting Up Classic Applications” on page 177](#)).
- 2 Define the application by completing information in the tabs. See:
 - [“Selecting Data Sources” on page 181](#)
 - [“Managing Data Sources” on page 178](#)
 - [“Setting up Currencies” on page 183](#)
 - [“Specifying Plan Types” on page 183](#)
 - [“Reviewing Application Information” on page 184](#)

Managing Data Sources

Each Planning application must be associated with a data source, which links the relational database and the Essbase server. To use the Classic Application Wizard to create, update, test, and delete data sources for Classic Planning applications, see:

- [“Creating Data Sources” on page 179](#)
- [“Editing Data Sources” on page 180](#)
- [“Checking Connections” on page 180](#)
- [“Deleting Data Sources” on page 181](#)

For Planning applications created in Performance Management Architect, you manage data sources in Performance Management Architect. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide*.

Data sources must be associated with instances, also called clusters. To update clusters, use the EPM System Configurator. See the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* or Online Help.

Creating Data Sources

To create data sources for Classic Planning applications, enter the data source name and description, select a relational database, and specify details for the relational database and Essbase server. The relational database and Essbase server password information is stored as encrypted.

You can also set the application to Unicode mode. Unicode-mode applications support multiple character sets. For Unicode-mode applications, Essbase uses the UTF-8 encoding form to interpret and store character text. Character-based artifacts in Unicode-mode applications, such as member and alias names, can include characters from different languages. For more information, see the *Oracle Essbase Database Administrator's Guide*.

► To create data sources for Classic Planning applications:

1 In the Classic Application Wizard, click **Manage Data Source**.

See “[Setting Up Classic Applications](#)” on page 177.

2 Click **Create Data Source**.

3 Enter the data source name and description.

4 Select a relational database.

5 Specify relational database details:

- **Server:** Server hosting the database
- **Port:** Port (for default ports, see *Oracle Hyperion Enterprise Performance Management System Installation Start Here*)
- **Database:** Database name
- **User:** Database username
- **Password:** Database password

6 Specify Essbase server details:

- **Server:** Server name
- **User:** Server username
- **Password:** Server password

7 **Optional:** To set the application to Unicode mode, select **Unicode Mode**.

For more information, see the *Oracle Essbase Database Administrator's Guide*.

8 Click **Finish**.

9 At the message that the data source was created successfully, click the X to close the message.

10 Use the EPM System Configurator to select a cluster (instance) to use for the application. See the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* or Online Help.

Editing Data Sources

For Classic Planning applications, you can use the wizard to update the data source name, description, relational database, and details for the relational database and Essbase server.

➤ To edit data sources:

1 In the Classic Application Wizard, click **Manage Data Source**.

See “[Setting Up Classic Applications](#)” on page 177.

2 Select the data source to edit, and click **Edit Data Source**.

3 Update the data source name and description.

4 Select a relational database.

5 Specify relational database details:

- **Server:** Server hosting the database
- **Port:** Port. For information about default ports and how to change them, see *Oracle Hyperion Enterprise Performance Management System Installation Start Here*.
- **Database:** Database name
- **User:** Database username
- **Password:** Database password

6 Specify Essbase server details:

- **Server:** Server name
- **User:** Server username
- **Password:** Server password

7 **Optional:** To set the application to Unicode mode, select **Unicode Mode**.

For more information, see the *Oracle Essbase Database Administrator's Guide*.

8 Click **Finish**.

9 Use the EPM System Configurator to select an instance (cluster) to use for the application.

See the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* or Online Help.

10 At the message that the data source was updated successfully, click the X to close the message.

Checking Connections

You can test the connections to the database or Essbase.

➤ To check connections:

1 In the Classic Application Wizard, click **Manage Data Source**.

See “[Setting Up Classic Applications](#)” on page 177.

- 2 Click a button for the connection to test:
 - Check Database Connection.
 - Check Essbase Connection.
- 3 At the message that the connection is successful, click the X to close the message.

Deleting Data Sources

For Classic Planning applications, you can use the Classic Application Wizard to delete data sources that are not associated with an application.

- To delete data sources:
- 1 In the Classic Application Wizard, click **Manage Data Source**.
See [“Setting Up Classic Applications” on page 177](#).
 - 2 Select the data source to delete, and click **Delete Data Source**.
 - 3 At the message that the data source was deleted successfully, click the X to close the message.

Selecting Data Sources

Specify the application name and description, register the application with Shared Services, and select Business Rules or Calculation Manager to calculate business rules. For information about Calculation Manager, see the Hyperion Calculation Manager part of the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

A default instance (cluster) is set up when you install and configure Planning. To update clusters with the Oracle's Hyperion Enterprise Performance Management System Configurator, see *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* or Online Help.

- To select data sources and the calculation module:
- 1 In the Classic Application Wizard, click **Create Application**.
See [“Setting Up Classic Applications” on page 177](#).
 - 2 Click **Data Sources**.
See [“About Updating Instances and Clusters” on page 38](#).
 - 3 Enter the name of the application.
The name can contain up to eight characters. It must not be the same as an existing Essbase application.
 - 4 Enter a description of the application.
 - 5 Select a Shared Services project.
 - 6 Select an instance (cluster) to use for the application.

See *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* or Online Help.

- 7 For **Calculation Module**, select the calculation module to use for the application, either **Business Rules** or **Calculation Manager**.
- 8 **Optional:** To make this a sample application, select **Sample Application**.
See [Appendix C, “Sample Application.”](#)
- 9 Perform one action:
 - For new applications, continue defining the application by setting up the calendar.
 - For existing applications, click **Finish**.

Setting up the Calendar

The calendar establishes the application’s base time periods, starting fiscal year and month, and total number of years. Select the base time period and monthly distribution pattern based on the number of fiscal weeks in a month. The base time period options are the bottom-level time periods in the application. You can create a custom base time period, such as weeks or days. Use monthly distribution patterns to determine how data entered into a summary time period is distributed or spread among the base time period you select. During data entry, users can enter data into summary time periods, such as years or quarters. Planning distributes these values over the base time periods that constitute the summary time period.

► To set up the calendar:

- 1 In the **Classic Application Wizard**, click **Calendar**.

See [“Setting Up Classic Applications”](#) on page 177.

- 2 Select a **Base Period** option to set how calendars roll up:

- **12 Months:** Four quarters per year; months roll up into parent quarters and quarters into years.
- **Quarters:** Quarters roll up into years.
- **Custom:** A custom time period, such as weeks or days.

- 3 Select the **Fiscal Start Year**.

This defines the starting fiscal year for the application. You cannot change this after creating the application. Before specifying the fiscal start year, consider how much historical data your organization needs and wants in the application.

- 4 Select the **Fiscal Start Month**.

This is the month in which the fiscal year starts for the application.

- 5 If you set the base time period as **12 Months**, select a **Weekly Distribution** option: **Even**, **445**, **454**, or **544**.

Weekly distribution sets the monthly distribution pattern, based on the number of fiscal weeks in a month. This determines how data in summary time periods spreads within the base time

period. When users enter data into summary time periods, such as quarters, the value is distributed over base time periods in the summary time period.

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

6 Select the **Total Years** for the application.

This defines the number of years in the calendar. You can add more years to the calendar after the application is created.

7 Perform one action:

- For new applications, continue defining the application by setting up currencies.
- For existing applications, click **Finish**.

Setting up Currencies

Specify the default currency for entities in the application, and establish if the application supports currency conversions. Multiple currency support (also called currency overrides) is available for level 0 members, regardless of their base currency.

► To set up currencies:

1 In the Classic Application Wizard, click **Currencies**.

See [“Setting Up Classic Applications” on page 177](#).

2 Specify the default currency for entities in the application.

3 Select **Yes** for multicurrency applications, or **No** for single-currency applications.

After the application is created, you cannot change this option. Multiple currency support is available for level 0 members, regardless of base currency. If you select Yes, two additional dimensions are created, Currency and HSP_Rates.

4 Perform one action:

- For new applications, continue defining the application by specifying plan types.
- For existing applications, click **Finish**.

Specifying Plan Types

Specify one to three plan types for the application. A separate Essbase database is created for each plan type. You cannot change the name or number of plan types after creating an application.

As you create accounts, entities, and other elements of the application, you associate them with plan types, so the database for each plan type contains only information relevant to the plan type. This optimizes application design, size, and performance.

➤ To select plan types:

1 In the Classic Application Wizard, click **Plan Types**.

See [“Setting Up Classic Applications” on page 177](#).

2 For each plan type in the application, select the **Plan Type** and specify a plan name.

You must select at least one Planning plan type. You can have up to three Planning plan types, and names can contain up to eight characters. (Although it is possible to enter more than eight bytes using single-byte and double-byte characters, an error message displays when the Essbase database is created.)

3 Click **Finish** to review application information and create or update the application.

Reviewing Application Information

➤ To review application information before creating or updating the application:

1 In the Classic Application Wizard, click **Finish**.

See [“Setting Up Classic Applications” on page 177](#).

2 Review the selected settings. Some settings cannot be changed after the application is created.

3 **Optional:** To modify application settings, click **Previous**, update the application information, and click **Finish** to check the updated settings.

4 To create or update the application, click **Finish**.

Registering Applications

You can use the Classic Application Wizard to reregister the Shared Services project for an application.

➤ To register applications:

1 Perform one action:

- From Planning, select **Administration**, then **Register Application**.
- From EPM Workspace, select **Navigate**, then **Administer**, then **Classic Application Administration**, and then **Planning Administration**. Then select **Register Application**.

See [“Setting Up Classic Applications” on page 177](#).

2 Select an existing Planning application, and click **Register**.

3 Select a Shared Services project, and click **OK**.

Deleting Applications

You can use the Classic Application Wizard to delete Classic applications. Oracle recommends backing up the application before deleting. See [“Backing Up Applications and Application Databases” on page 74](#).

► To delete applications:

1 **Back up the application.**

2 **Perform one action:**

- From Planning, select **Administration**, then **Delete Application**.
- From EPM Workspace, select **Navigate**, then **Administer**, then **Classic Application Administration**, and then **Planning Administration**. Then select **Delete Application**.

See [“Setting Up Classic Applications” on page 177](#).

3 **Select an existing Classic application, and click **Delete Application**.**

4 **At the confirmation message, click **OK** if you want to proceed with the deletion.**

Working with Alias Tables

You can create and update alias tables, and set a default alias table for the application. Follow naming conventions in [Appendix B, “Naming Restrictions for Essbase.”](#)

About Alias Tables

You can assign alternate names, or aliases, to Planning Account, Currency, Entity, Scenario, Period, Version, Year, and user-defined dimension members. Planning allows up to 10 aliases per dimension member, including the default alias.

When creating a Planning application, Essbase creates an empty default alias table in the database outline. If you do not create other alias tables, all aliases are stored in this default table. You cannot delete the default table.

You can create up to nine alias tables in Essbase. If you add or change aliases or alias tables, you must refresh the application. Changes are not in effect until the database is updated. To view the database outline, open Administration Services Console, select Outline, then Aliases, then Set Table, and select an alias table. You can use only alias tables created from within Planning. Alias tables created outside Planning are deleted during application refresh.

Multiple alias tables support these language combinations:

- English, French, German, Spanish, and Italian
- Japanese and English
- Korean and English
- Turkish and English

You can set alias tables to display members in applications. Planners can set alias tables in preferences.

Creating Alias Tables

- To create alias tables:
 - 1 Select **Administration**, then **Alias Tables**.
 - 2 Click **Add**.
 - 3 In **Explorer User Prompt**, enter a name.
 - 4 Click **OK**.

Editing or Renaming Alias Tables

- To edit or rename alias tables:
 - 1 Select **Administration**, then **Alias Tables**.
 - 2 Select the alias table.
 - 3 Click **Edit**.
 - 4 For **Explorer User Prompt**, enter a name.
 - 5 Click **OK**.

Deleting Alias Tables

- To delete alias tables:
 - 1 Select **Administration**, then **Alias Tables**.
 - 2 Select the alias table.

You cannot delete the default alias table.
 - 3 Click **Delete**.
 - 4 Click **OK**.

Clearing Alias Tables

You can clear the contents of alias tables.

- To clear alias tables:
 - 1 Select **Administration**, then **Alias Tables**.
 - 2 Select the alias table to clear.

Clearing the alias table removes the contents of the table but does not remove the table.

- 3 Click **Clear Values**.
- 4 Click **OK**.

Copying Alias Tables

- To copy alias tables:
 - 1 Select **Administration**, then **Alias Tables**.
 - 2 Select the alias table.
 - 3 Click **Copy**.
 - 4 Select the destination alias table.

The destination alias table must exist. Copying does not create tables.

- 5 Click **Copy**.

Setting a Default Alias Table

If you create alias tables with aliases for Account, Currency, Entity, Scenario, Period, Version, Year, and user-defined dimensions and members, you can select a default alias table for the application. Users can set preferences for which set of aliases (stored in an alias table) to use for displaying member and dimension names.

- To select the application's default alias table:
 - 1 Select **Administration**, then **Application Settings**.
 - 2 In **Current Application Defaults**, select **Application Settings**.
 - 3 For **Alias Table**, select a default alias table.
 - 4 Click **Save** (or **Reset**).

Working with Dimensions

You can create and update dimensions by selecting Administration, then Dimensions.

Dimension Overview

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

About Dimensions and Members

Members are components of dimensions.

About Sparse and Dense Dimensions

Sparse dimensions lack data values for the majority of member combinations. Dense dimensions have data values for the majority of member combinations. Essbase requires at least one dense dimension. Custom attributes cannot be assigned to dense dimensions. Planning designates the Account and Period dimensions as dense, and remaining dimensions as sparse. To optimize performance for sparse dimensions, Planning searches for and calculates only occupied data values in each dimension combination. This decreases calculation time and lowers disk usage. You can modify these settings. See [“Optimizing Application Performance” on page 73](#).

About Dimension Hierarchies

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

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

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

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

Working with Dimension Hierarchies

Task	Topic
Specify or change dimension properties.	Click Edit.
Add a dimension.	Click Add Dimension.
Search for a dimension member.	See “Finding Dimensions or Members” on page 190 .
Expand or collapse the dimension hierarchy.	Click Expand or Collapse.
Add or edit a dimension member.	Click Add Child or Add Sibling.

Task	Topic
Move a dimension member.	See “Moving Members Within the Dimension Hierarchy” on page 190.
Delete a dimension member.	See “Deleting Members” on page 203.
Assign access to a dimension member.	See “Assigning Access to Members and Business Rules” on page 42.
View a member’s ancestors.	Click Show Ancestors.

Expanding and Collapsing Dimension Hierarchies

- To expand dimensions or members:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 For **Dimension**, select the dimension and member to expand.
 - 3 Perform one action:
 - Click **Expand**.
 - Press the **Right Arrow**.
 - Click **+**.
 - Click the closed folder.



- To collapse dimensions or members:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 For **Dimension**, select the dimension to expand.
 - 3 Select the dimension or member to collapse.
 - 4 Perform one action:
 - Click **Collapse**.
 - Press the **Left Arrow**.
 - Click **=**.
 - Click the open folders.

Navigating Dimension Hierarchies

- Press Up Arrow to move to the previous member.
- Press Down Arrow to move to the next member.
- In Page, enter the page to view and click Go or press Enter.
- Click Start, Prev, Next, or End to view other pages.

By default, 14 members are displayed per page. You can change this by setting preferences for Show the Specified Members on Each Dimensions Page.

Finding Dimensions or Members



- To find dimension members in dimension hierarchies:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 For **Dimension**, select the dimension for the member.
 - 3 For **Search**, select name, alias, or both.
 - 4 Enter the member name, alias, or partial string for which to search.
 - 5 Click **Search Down**  or **Search Up** .

Sorting Members

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

- To sort members:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 For **Dimension**, select the dimension for the members.
 - 3 On **Dimensions**, select the members whose children or descendants you want to sort.
 - 4 For **Sort**, select children or descendants.

Sorting by children affects only members in the level immediately below the selected member. Sorting by descendants affects all descendants of the selected member.



- 5 Click  to sort by ascending order or  to sort by descending order.
- 6 Click **OK**.

The next time you create or refresh the database, the Essbase outline is generated with members in the order that is displayed.

Moving Members Within the Dimension Hierarchy

You can move one member or a group of members in the same branch. If you move Account members whose Valid For Plan Type settings differ from their new parent, the moved member settings change to match the setting of the new parents. If you move members whose Source Plan Type settings differ from their new parent, the moved member Source Plan Type is reset to match the first valid plan type.

- To move members or branches among siblings:
 - 1 Select **Administration**, then **Dimensions**.

- 2 For **Dimension**, select the dimension for the members to move.
- 3 Select the member or branch to move.
- 4 Perform one action:
 - Click  to move the member up one position.
 - Click  to move the member down one position.

► To move members, including parents and children:

- 1 Select **Administration**, then **Dimensions**.
- 2 For **Dimension**, select the dimension with the members to move.
- 3 Select the member or branch to move.
- 4 Click **Cut**.

You cannot Cut members after adding or editing dimensions, navigating to different pages, deleting members, or logging off Planning. Not available for root dimension members.
- 5 Click the destination level under which to move the members.
- 6 Click **Paste**.
- 7 Click **OK**.
- 8 Update and validate business rules and reports.

Minimizing and Restoring Columns

► To minimize and restore columns:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the column to minimize or restore.
- 3 Perform one action:
 - To minimize a column, double-click its heading, or right-click and select **Minimize**.
 - To return a column to its original width, double-click a minimized column heading, or right-click and select **Restore**.
 - To restore all minimized columns to original widths, right-click a column heading and select **Restore All**.

Planning saves column widths for each dimension independently. Column widths are saved for the duration of the session when you add, edit, or delete a dimension member, or when you select another dimension.

Viewing a Member's Ancestors

- To view a member's ancestors:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 For **Dimension**, select a dimension.
 - 3 Select the member in the dimension hierarchy.
 - 4 Click **Show Ancestors**.
 - 5 Click **Close**.

Determining Where Members Are Used in Applications

- To view where members are used in applications:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select the dimension whose member's usage you want to view.
 - 3 Click **Show Usage**.
 - 4 At the bottom of the **Member Usage** window, select where in the application to view the member's usage.
Options are appropriate for the selected dimension member.
 - 5 Click **Go**.
 - 6 Click **Close**.

About Custom Dimensions

Planning includes two custom dimensions: Account and Entity. You can edit the names of these dimensions, and create up to 13 user-defined dimensions. Use Account and user-defined dimensions to specify data to gather from planners. Use Entity to model the flow of planning information in the organization and establish the plan review path.

Aggregation Options

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

- + Addition
- - Subtraction
- * Multiplication
- / Division
- % Percent
- ~ Ignore

- Never (do not aggregate, regardless of hierarchy)

Storage Options

Table 58 Storage Options

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

About Dynamic Calc

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

Dynamic Calc Versus Dynamic Calc and Store

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

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

Note:

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

About Store Data Storage

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

About Shared Data Storage

Use Shared to allow alternate rollup structures in the application.

About Never Share Data Storage

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

About Label Only Data Storage

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

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

Caution! Do not design data forms in which label-only parents follow their first child member, as you cannot save data in the first child member. Instead, create data forms with label-only parents selected before their children, or do not select label-only parents for data forms.

About Entities

Entities typically match your organization's structure, such as geographical regions, departments, or divisions. Create entity members for groups that submit plans for approval. Entity members help define budget review, or workflow (see [“Managing the Budgeting Process” on page 147](#)).

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

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

Base Currency

For multicurrency applications, specify each entity member's base currency. The default base currency for entity members is the currency specified when creating the application. For example, if U.S. Dollars is the default currency, you may specify Yen as the base currency for the Japan entity and U.S. Dollars for the United States entity. When using data forms having values for the Japan entity, if the display currency is set to U.S. Dollars, values are converted to U.S. Dollars using the rates in the exchange rate table (assuming Yen is the local currency and U.S. Dollars is the reporting currency).

About Accounts

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

Account Types

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

Examples of Using Account Types

Table 59 Using Account Types

Account Type	Purpose
Expense	Cost of doing business
Revenue	Source of income
Asset	Company resource
Liability and Equity	Residual interest or obligation to creditors
Saved assumption	Centralized planning assumptions ensuring consistency across the application

Summary of Account Types

Table 60 Summary of Account Types

Account Type	Time Balance	Variance Reporting
Revenue	Flow	Non-Expense
Expense	Flow	Expense
Asset	Balance	Non-Expense
Liability	Balance	Non-Expense
Equity	Balance	Non-Expense
Saved Assumption	User-defined	User-defined

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

Time Balance Property

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

Table 61 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: 10
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	Jan: 10 Feb: 15 Mar: 20 Q1: 15

Time Balance Property	Description	Example
	calculated: (1) Multiply each month's value in Q1 by the number of days in the month, (2) Sum these values, (3) Divide the total by the number of days in Q1. Assuming it is a leap year, the result is calculated: $(10 * 31 + 15 * 29 + 20 * 31) / 91 = 15$	
Weighted Average - Actual_365	Weighted daily average, based on 365 days in a year, assuming that February has 28 days; does not account for leap years. In the example, the average for Q1 is calculated: (1) Multiply each month's value in Q1 by the number of days in the month, (2) Sum these values, (3) Divide the total by the number of days in Q1. Assuming it is not a leap year, the result is calculated: $(10 * 31 + 15 * 28 + 20 * 31) / 90 = 15$	Jan: 10 Feb: 15 Mar: 20 Q1: 15

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

Account Types and Variance Reporting

An account's variance reporting property determines whether it is treated as an expense when used in Essbase 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 62 Effect of Skip Options When Time Balance is Set to First

Skip Option	Description	Example
None	Zeros and #MISSING values are considered when calculating parent values (the default). In the example, the value of the first child (Jan) is 0, and zeros are considered when calculating the parent value, so Q1 = 0.	Jan: 0 Feb: 20 Mar: 25 Q1: 0

Skip Option	Description	Example
Missing	Excludes #MISSING values when calculating parent values. In the example, the value of the first child (Jan) is #MISSING, and #MISSING values are not considered when calculating parent values, so Q1 = second child (Feb), or 20.	Jan: #MISSING Feb: 20 Mar: 25 Q1: 20
Zeros	Excludes zero values when calculating parent values. In the example, the value of the first child (Jan) is 0, and zero values are not considered when calculating parent values, so Q1 = the second child (Feb), or 20.	Jan: 0 Feb: 20 Mar: 25 Q1: 20
Missing and Zeros	Excludes #MISSING and zero values when calculating parent values. In the example, the value of the first child (Jan) is zero, and the value of the second child (Feb) is missing. Because missing and zero values are not considered when calculating parent values, Q1 = the third child (Mar), or 25.	Jan: 0 Feb: #MISSING Mar: 25 Q1: 25

Saved Assumptions

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

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

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

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

Data Type and Exchange Rate Type

Data type and exchange rate type determine how values are stored in account members, and the exchange rates used to calculate values. Available data type for account members' values:

- Currency - Stores and displays in the default currency.
- Non-currency - Stores and displays as a numeric value.
- Percentage - Stores a numeric value and displays as a percent.
- Date - Displays as a date.
- Text - Displays as text.

For accounts with the Currency data type, available Exchange Rate types (valid for any time period):

- Average - Average exchange rate
- Ending - Ending exchange rate
- Historical - Exchange rate in effect when, for example, earnings for a Retained Earnings account were earned or assets for a Fixed Assets account were purchased.

Accounts, Entities, and Plan Types

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

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

Entities and Plan Types

Typically entity members prepare different plans. When defining entity members, specify plan types for which they are valid. Because data forms are associated with plan types, you can control which entity members can enter data for each plan type.

Accounts and Plan Types

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

About User-Defined Custom Dimensions

You can add up to 13 user-defined custom dimensions. For example, you could add a dimension called Project to budget operating expenses for each project. You define properties, including name, alias, plan type, security, attributes, and attribute values.

Caution! You cannot delete custom dimensions after you create them.

User-defined custom dimensions differ from the Entity and Account dimensions in that you assign valid plan types at the dimension level, not at the member level. All members of a user-defined custom dimension are valid for plan types assigned at the dimension level.

Adding or Editing User-Defined Custom Dimensions

User-defined custom dimensions must conform to guidelines listed in [Appendix B, “Naming Restrictions for Essbase.”](#)

Table 63 Properties for User-Defined Custom Dimensions

Property	Value
Dimension	Enter a unique name.
Alias	Optional: Select an alias table. Enter a unique alternate name for the dimension.
Description	Optional: Enter a description.
Valid for Plan Types	Select plan types for which the dimension is valid. Clearing this option makes all members of the dimension invalid for the deselected plan type.
Apply Security	Allow security to be set on the dimension members; must be selected before assigning access rights to dimension members. Otherwise, dimensions have no security and users can access members without restriction.
Data Storage	Select a data storage option. The default is Never Share.

➤ To add or change user-defined dimensions:

- 1 Select **Administration**, then **Dimensions**.
- 2 Click **Add Dimension**.
- 3 Set or change properties.
- 4 Click **Save**.
- 5 Click **OK**.

Click Refresh to revert to the previous values and keep the page open.

Setting Dimension Properties

Dimension properties must conform to guidelines listed in [Appendix B, “Naming Restrictions for Essbase.”](#)

Table 64 Dimension Properties

Property	Value
Dimension	Enter a dimension name.
Alias	Optional: Select an alias table and enter an alternate name of up to 80 characters. Follow the dimension naming restrictions.

Property	Value
Description	Optional: Enter a description.
Valid for Plan Types	Select plan types for which the dimension is valid. Not available for Entity or Account dimensions.
Apply Security	Allow security to be set on dimension members. If you do not select this option, there is no security on the dimension, and users can access its members without restriction. Must be selected before assigning access rights to dimension members.
Data Storage	Select data storage options.
Enable custom attribute display	Display available and selected attributes for dimensions with associated attributes. Enable custom attribute display for dimensions with attributes.

► To set properties for dimensions:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the dimension.
- 3 Click **Edit**.
- 4 In **Dimension Properties**, set property options.
- 5 Click **Save** to save information to the relational database and see changes in the dimension hierarchy.

Click Refresh to revert to the previous values on the page and keep the page open.

Working with Members

You can assign access rights to members, rearrange the dimension member hierarchy, and share members of the Entity, Account, and user-defined custom dimensions.

Adding or Editing Members

Members must conform to guidelines listed in [Appendix B, “Naming Restrictions for Essbase.”](#)

Table 65 Member Properties

Property	Value
Name	Enter a name.
Description	Optional: Enter a description.
Alias	Optional: Select the alias table to store the alias name. Enter an alternate name for the member.
For Account members only: Account Type	Select the account type.
For Account members only: Variance Reporting	If the account type is Saved Assumptions, for Variance Reporting, select Expense or Non-Expense. Designate the saved assumption as a revenue, asset, liability, or equity account.

Property	Value
For Account members only: Time Balance	For Time Balance, select Flow or Balance.
For Account members only: Exchange Rate Type	For Exchange Rate Type, select an option.
For Account members only: Data Type	For Data Type, select a data type.
Distribution	Sets the weekly distribution. Available for leaf Account members if the option was selected when creating the application and the base time period is 12 months.
Data Storage	Select a data storage property used in building an Essbase cube. The default is Never Share for new custom dimension members (except root members).
Two Pass Calculation	Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with Dynamic Calc or Dynamic Calc and Store properties.
For Entity members only: Base Currency	Select the Entity member's base currency.
Plan Type	<p>Select plan types for which the member is valid.</p> <p>Select an aggregation option for each selected plan type. You can select a source plan only if multiple plan types are valid for the member. Only plan types and aggregation options for which the member's parent is valid are available. If the parent is not valid for a plan type or aggregation option, neither is the child member. Deselecting a plan type for an account or entity parent member deselects it for all descendents of that parent.</p> <p>Caution! Deselecting a plan type for dimension members after data is entered into applications may result in loss of data when applications are refreshed. For account members, data is lost if the deselected plan type is the source plan type.</p>
For Entity members only: Base Currency	Select the base currency for the Entity member.
For Account members only: Source Plan Type	Select the source plan type for the member. A shared member is a pointer to the base member and is not stored in Essbase; this is disabled for shared members. The Source Plan Type of a shared Account member matches the source plan type of the base member, even though Source Plan field is unavailable because it does not apply to shared members.
Smart Lists	Optional: Select a Smart List to associate with the member.

➤ To add or edit members:

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

2 Select the dimension.

3 Perform one action:

- To add a child member, select the parent level of the dimension hierarchy to which to add a member and click **Add Child**.
- To add a sibling, select the level of the dimension hierarchy to which to add a sibling and click **Add Sibling**.

- To edit a member, select that member from the dimension hierarchy and press **Enter** or click **Edit**.
- 4 On **Member Properties**, set or change member properties.
If you do not see the new member on the page, click **Next**.
 - 5 Click **Save** to save information to the relational database and see changes in the dimension hierarchy.
 - 6 Refresh the database so edited members are visible to planners entering data.
 - 7 After creating a dimension member, you typically complete these tasks:
 - Assign access. See [“Assigning Access to Members and Business Rules”](#) on page 42.
 - Specify attributes.

Deleting Members

Caution! Each data value is identified by a set of dimension member values and a plan type. Deleting dimension members or deselecting the plan type results in data loss when refreshing applications. Deleting entity members deletes all planning units (including data) associated with them. Before beginning this procedure, perform a backup. See [“Backing Up Applications and Application Databases”](#) on page 74 and the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.

Before deleting members, understand where in the application they are used (in which data forms, planning units, exchange rates, and so on) by using **Show Usage**.

You must delete the entity member throughout Planning before deleting it from Dimensions. For example, if the entity member is used in a data form, you must delete it from the data form before deleting it from Dimensions.

When deleting a large subtree of entities, you can improve performance if you first exclude planning units for the subtree (by excluding the root member) for all scenarios and versions. See [“Starting the Review Process”](#) on page 147.

- To delete members:
- 1 Select **Administration**, then **Dimensions**.
 - 2 Select the dimension whose member you want to delete.
 - 3 From the dimension hierarchy, select the entity member to delete.
 - 4 Click **Delete**.
Deleting a base member also deletes its shared members.
 - 5 Click **OK**.
 - 6 Update and validate business rules and reports.

Deleting Parent Members

Caution! Data values are identified by a set of dimension member values and a plan type. Deleting dimension members or deselecting the plan type results in data loss when refreshing the application. Before beginning this procedure, perform a backup. See [“Backing Up Applications and Application Databases” on page 74](#) and the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.

➤ To delete a parent member and all its descendants from the dimension hierarchy:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the dimension whose member and descendants to delete.
- 3 Select the member whose branch to delete.
- 4 Click **Delete**.
- 5 Click **OK**.

Viewing Member Properties from Data Forms

➤ To view member properties from data forms:

- 1 In the data form, select the member and right-click.
- 2 Select **Show member properties in outline**.

The Dimensions page displays the member highlighted in the hierarchy.

- 3 **Optional:** Select **Edit** to view the member's properties, then click **Cancel**.

Working with Shared Members

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

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

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

Shared members cannot be moved to another parent member. You must delete shared members and recreate them under different parent members. Shared members must be at the lowest level

(level zero) in the hierarchy and cannot have children. The base member need not be level zero. You can enter data in shared members, and values are stored with base members.

Shared members are displayed similarly to base members in the dimension hierarchy for member selection in Business Rules and Smart View. For Oracle's Hyperion® Business Rules, you might not be able to select shared members for actions such as Copy Data.

Creating Shared Members

You create shared members the same way as other members, with these differences:

- The base member cannot be the parent of the shared member.
- You must give the shared member the same name as its base member. It can have a different description.
- You must select Shared as the Data Storage for the shared member.

Working with Attributes

Use attributes to group members using the same criterion. You can assign attributes to sparse dimensions only. You cannot assign attributes to label-only members. Attribute dimensions do not have aggregation properties because parents are dynamically calculated by Essbase.

The Account dimension is usually defined as dense, so you cannot assign attributes to it unless it is changed to sparse for all plan types. If you change a dimension from sparse to dense, all attributes and attribute values for that dimension are automatically deleted.

Attributes can have data types of text, date, Boolean, and numeric, as described in [“Understanding Attribute Data Types” on page 206](#). Attribute names must conform to guidelines listed in [Appendix B, “Naming Restrictions for Essbase.”](#) When attributes are defined, you can use the Member Selection dialog box to select attribute functions, such as Equal and GreaterOrEqual. See [“Selecting Members” on page 117](#).

► To create and change attributes, attribute values, and aliases:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select a sparse dimension for which to define an attribute, attribute value, or alias.

Only sparse dimensions can contain attributes.

- 3 Select the top level in the dimension hierarchy, and click **Edit**.
- 4 In the Dimension Properties dialog box, click **Custom Attributes**.

If the dimension is not sparse, Custom Attributes is not available.

- 5 Select options.
 - To create attributes, click **Create**. Type an attribute name, and select a data type: Text, Date, Boolean, or Numeric. See [“Understanding Attribute Data Types” on page 206](#). You cannot modify the data type after the attribute is created.

Plan type options are available for Entity dimension attributes only. You cannot change this setting after the attribute is created.

- To modify attributes, click **Modify**, and update the attribute name.
- To set aliases for attributes, select an attribute and an attribute value, click **Alias**. Select an alias table, type an alias name, and click **Close**.

6 Click **Close**.

When you click **Close**, the hierarchy is validated and an error displays if issues are detected. For example, date attribute values must be entered in the correct format, and numeric and date attribute dimensions must have at least one attribute value defined.

7 Update and validate business rules and reports.

Understanding Attribute Data Types

Attribute dimensions can have a data type of text, numeric, Boolean, or date that enables different functions for grouping, selecting, or calculating data. The attribute type applies only to level 0 members of the attribute dimension.

- Text attributes enable basic attribute member selection and attribute comparisons in calculations. When you perform such comparisons, Essbase compares characters. For example, a package type **Bottle** is less than a package type **Can** because **B** precedes **C** in the alphabet.
- Numeric attribute dimensions use numeric values for the names of level 0 members. You can include the names (values) of numeric attribute dimension members in calculations. For example, you can use the number of ounces specified in an **Ounces** attribute to calculate profit per ounce for each product. You can also associate numeric attributes with ranges of base dimension values, for example, to analyze product sales by market population groupings.
- Boolean attribute dimensions in a database contain only two members. When a Boolean attribute dimension is added in Planning, two attribute values, **True** and **False**, are created for this attribute dimension by default. A base dimension, such as **Account** or **Entity**, can be associated with only one attribute dimension that has the Boolean data type.
- Date attributes can specify the date format as month-day-year or day-month-year, and sequence information accordingly. You can use date attributes in calculations, for example, comparing dates in a calculation that selects product sales since 12-22-1998. Users can set the date format by selecting an option in **Attribute Dimension Date Format** in **Application Settings** preferences.

For detailed information about attributes and attribute values, see the *Oracle Essbase Database Administrator's Online Help*.

Deleting Attributes

When you delete an attribute, all attribute values associated with the attribute are also deleted. Attribute values are removed from members to which they had been assigned, and the attribute is removed from dimensions to which it was assigned.

- To delete attributes:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select the sparse dimension for which to delete an attribute, and click **Edit**.
 - 3 Click **Custom Attributes**.
 - 4 Select the attribute to delete.
 - 5 Above the **Attributes** column, click **Delete**.
 - 6 Click **OK**.
 - 7 Update and validate business rules and reports.

Working with Attribute Values

Attribute values provide users with another way of selecting dimensions members when using data forms. Data values for attribute values are dynamically calculated but not stored.

Creating Attribute Values

You can define attribute values for sparse dimensions, which are typically the Entity and user-defined custom dimensions. After you define an attribute value for a dimension, you can assign it to members of that dimension.

- To create attribute values:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select the sparse dimension for which to create an attribute value.
 - 3 Select the top level in the dimension hierarchy.
 - 4 Click **Custom Attributes**.
 - 5 On the **Manage Attributes and Values** page, select the attribute for which to specify a value.
 - 6 Above the **Attribute Values** column, click **Create**. If the options are available, you can click **Add Child** or **Add Sibling**.
 - 7 On **Create Attribute Value**, in **Name**, enter a name.
 - 8 Press **Enter** or **Save**.
 - 9 At **Create Attribute**, click **Cancel**.

Assigning Attribute Values to Members




You can assign attribute values members of a dimension that are defined as sparse for all plan types. Attribute values must be assigned to the same-level sparse dimension members. Otherwise, errors display during refresh.

➤ To assign attribute values to members:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the sparse dimension for whose member you want to assign an attribute value.
- 3 In the **Dimension** hierarchy, select a member to which to assign an attribute value.
- 4 Click **Edit**.

For members assigned attribute values: Click View to change a member's attribute value.

- 5 Select **Attribute Values**.
- 6 Select attribute values to assign to the member.
- 7 Perform an action:

- To assign the value to the selected member, click .
- To remove a value from the selected member, select the value to remove and click .
- To remove all values from the selected member, click .

- 8 Click **Save**.

Modifying Attribute Values

➤ To modify attribute values:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the sparse dimension for which to modify an attribute value.
- 3 Select the top level in the dimension hierarchy.
- 4 Click **Custom Attributes**.
- 5 For **Attributes**, select the attribute containing the value to modify.
- 6 For **Attribute Values**, select the attribute value.
- 7 Above **Attribute Values**, click **Modify**.
- 8 On **Modify Attribute Value**, in **Name**, enter a name.
- 9 Click **Save**.

Deleting Attribute Values

When you delete an attribute value, it is removed from custom dimension members to which it is assigned.

➤ To delete attribute values:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the sparse dimension containing the attribute for which to delete a value.

- 3 Select the top level in the dimension hierarchy.
- 4 Click **Custom Attributes**.
- 5 For **Attributes**, select the attribute containing attribute values to delete.
- 6 For **Attribute Values**, select attribute values to delete.

To select all attribute values for deletion, select Attribute Values.

- 7 Above the **Attribute Values** column, click **Delete**.
- 8 Click **OK**.
- 9 Update and validate business rules and reports.

Customizing Calendars

Use the Period dimension to work with the yearly calendar rollup structure. When creating the application, the administrator specifies the base time periods that span the Planning database. Use the Year dimension to add years to the calendar.

Table 66 Calendar Tasks

Task	See Topic
Define how years roll up.	“Defining How Calendars Roll Up” on page 209.
Create and edit Summary Time Periods.	“Creating and Editing Summary Time Periods” on page 210.
Delete Summary Time Periods.	“Deleting Summary Time Periods” on page 210.
Work with the fiscal year and period hierarchy.	“Working with Years” on page 211.

Defining How Calendars Roll Up

Table 67 Calendar Roll Up

Base Time Period	Roll Up
12 Months	Four quarters are created per year. Months roll up into parent quarters and quarters roll up into years.
Quarters	Quarters roll up into years.
Custom	No default rollup structures. A flat list of the custom base time periods displays.

After the application calendar is created, you cannot change the base time period or reduce the number of years in the calendar. Administrators can change the names, descriptions, aliases, and ranges of the summary time periods in the hierarchy.

You can have up to 100 calendar years and 500 time periods in an application. Actual limits are a function of calendar years and time periods. How many time periods and years you can set also depends on whether your application uses multiple currencies. Oracle recommends these practical limits for an application:

- 400 time periods per year and 27 years
- 360 time periods per year and 30 years

Creating and Editing Summary Time Periods

You can change such aspects as name, description, alias, starting period, and ending period. However, you cannot change the order of base time periods or skip base time periods. The range cannot extend beyond the current fiscal year.

You must work from the top of the hierarchy to the bottom when creating summary time periods. (Otherwise, Planning views the rollup structure as asymmetrical and you cannot continue.) The summary time period displays in the hierarchy as a parent of the selected item. To enforce a balanced hierarchy, all base members must be the same number of levels from the root.

► To create or edit summary time periods:

1 Select Administration, then Dimensions.

2 Select Period.

3 Perform one action:

- To add a time period, select the level in the dimension hierarchy above which to add, and click **Add**.
- To edit a time period, select the time period and click **Edit**.

4 For Name, enter or change the name for the summary time period.

5 Optional: For Description, enter a description.

6 Optional: For Alias, select an alias table to use. Enter an alias.

The default table is used if you do not select one.

7 For Start Period, select the starting period.

The range cannot extend beyond the current fiscal year. For summary time periods, Start Period displays the first child, or all children except the first child of the sibling above it.

8 For End Period, select the ending period.

For summary time periods, End Period displays the last child, or all children from the Start Period through the next sibling's children, except the last child.

9 Click Save.

Deleting Summary Time Periods

When you remove a summary time period from the hierarchy, its children are moved into another summary time period:

- If you delete the first summary time period, children are moved into the next sibling of the summary time period.

- If you delete the last summary time period, children are moved into the previous sibling of the summary time period.
- If you delete a summary time period from the middle of a hierarchy, children are moved into the previous sibling of the summary time period.

➤ To delete summary time periods:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Period**.
- 3 For **Period hierarchy**, select the summary time period to delete.

You cannot delete base time periods.

- 4 Click **Delete**.
- 5 Click **OK**.

Working with Years

Use the Year dimension to work with calendar years. You can add years, change the current year or period, and assign aliases to time periods or years.

Table 68 Year Tasks

Task	See Topic
Add years to the calendar.	“Adding Years to the Calendar” on page 211.
Set the fiscal year and work with years.	“Setting the Fiscal Year” on page 211.

Adding Years to the Calendar

You can add years to the calendar, but you cannot reduce the number of calendar years without creating a database.

➤ To add years to the calendar:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Year**.
- 3 Click **Add Years**.
- 4 For **Number of Years to Add**, enter the number of years to add to the calendar.
- 5 Click **Add Years**.

Setting the Fiscal Year

You can change the time period or current year.

- To change the current year or time period:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Year** or **Period**.
- 3 Click **Options**.
- 4 On **Set Current Time Period and Year**, from **Current Year**, select the current year.

Current Month and Current Year set defaults for the Month and Year when scenarios are created. For example, if Current Year is set to FY08 and Current Month is set to Aug, when users create scenarios, these values display as defaults in the Start Yr, Start Period, End Yr, End Period fields.

- 5 For **Current Time Period**, select the current period.
- 6 Click **Save**.

Renaming Time Periods

You can rename root-level, base time periods, and user-defined summary time periods.

Assigning Aliases to Summary Time Periods

You can assign an alternate name, or alias, to years on the Edit Aliases dialog box. You can also assign and change aliases to base time periods and summary time periods.

- To assign or change a year's alias:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Year**.
- 3 Select the year.
- 4 Click **Edit Year**.
- 5 On **Edit Aliases**, for **Alias Table**, select the alias table to use.
- 6 Enter an alias name.
- 7 Click **Save**.

Editing the BegBalance Member

You can edit the BegBalance member of the Period dimension. As the first time period in the application, the BegBalance member is useful for entering beginning data when you start a new application, fiscal year, or calendar year. You can rename and describe BegBalance and give it an alias.

- To edit the BegBalance member:

- 1 Select **Administration**, then **Dimensions**.
- 2 For **Dimensions**, select **Period**.
- 3 Select the first member, **BegBalance**.

- 4 Click **Edit**.
- 5 For **Edit Period**:
 - Enter a name.
 - Enter a description.
 - Select an alias table to use for the BegBalance member, and enter an alias.
- 6 Click **Save**.

Setting Up Currencies

You can plan, forecast, and analyze financial information in one or more currencies. You can create, edit, and delete currencies. Administrators control:

- Which currencies applications use, including for reporting
- How currencies display in reports and data forms
- How currencies convert to other currencies
- Whether a triangulation currency converts currencies
- When currency conversions occur

Enabling Multiple Currencies

If applications support multiple currencies, you can enable multiple currencies per entity on data forms. See [“Setting Other Options for Data Forms” on page 125](#). When selecting business rules for data forms, you can select the Calculate Currencies business rule to convert values among the available currencies. See [“Selecting Business Rules” on page 128](#).

Working with Multiple Currencies

If multiple currencies are enabled, users can see values converted from the local currency to a reporting currency and can override a cell’s base currency.

Note:

- When the local currency is selected on data forms, the default stored and displayed currency for cells is the entity’s base currency (which you specify). Users can enter data values only into local currency members. If the local currency member is selected, all currencies specified for the application are available as input types.
- Currencies can be converted only to reporting currencies. Users cannot enter data into cells displayed in reporting currencies. The application’s main currency is by default a reporting currency. You can change which currencies are reporting currencies.
- You can load values into a reporting currency by using Essbase Adapter to populate values directly into Essbase.

- Currencies defined for the application are valid currencies for data entry. Valid currencies for data entry are displayed in a list that users access by clicking Currency link during data entry.
- To get meaningful results, roll up values in one common reporting currency. If members of a subtotal have mixed currencies, the currency type is blank and the currency symbol does not display. For example, adding 10 US dollars and 10 Japanese yen to a value of 20 makes no sense.
- Applications with 500 time periods can successfully run currency conversion calc scripts only if the time periods have default names, TP 1 through 500. Otherwise, the conversion calc script you try to create exceeds the 64K limit.
- User-defined currency conversion calc scripts created when a database is created or refreshed may be available in Smart View, depending on user access. When using user-defined currency conversion calc scripts, Oracle recommends changing the order so the currency conversion calc script is first, before Calculate Data Form.
- Currency codes associated with input values are stored as numeric values in Essbase. These codes are calculated in dimension formulas, calc scripts, and business rules. The calculated values of these currency codes may translate to currency codes that are incorrect or invalid. Where there are children with mixed currencies, review calculated results on the upper levels.
- If a parent has multiple children, of whom only one child has an overridden currency, the parent inherits the overridden currency code in Essbase (which is not displayed on data forms).
- In certain cases, parent entities display #MISSING when trying to convert to a selected currency. Ensure that a currency rate is entered for each combination of local currencies and selected currencies on data forms or reports. Currency combinations must exist for all mixed-currency children entities and parent members.
- Input of multiple currencies to one entity is not supported in Oracle Hyperion Smart View for Office, Fusion Edition. If worksheets include mixed currency types, users could inadvertently enter values in the wrong currency.

About the Calculate Currencies Business Rule

The Calculate Currencies business rule is based on the dimensions and members on the data form. It converts data from the local currency to the reporting currency specified on the data form, applying the exchange rate conversions. It:

- Does not calculate subtotals. To subtotal values, run the Calculate Data Form business rule (or a customized business rule that includes aggregation) after converting currencies.
- Ignores #MISSING values.
- Can be turned on or off by associating or disassociating it with data forms during data form design.
- Is set by default to not run when saving data.

Exchange Rate Types

These exchange rates are associated with currencies: Historical, Average, and Ending. The exchange rate type for each account is specified in the Member Property dialog box. For average and ending rate types, enter values for all time periods. For historical rate types, enter one rate value that is used for all time periods, including the Beginning Balance period. For the Beginning Balance period, enter one rate value used for that time period for average and ending rate types.

Planning supports currency conversion by triangulation through a triangulation currency.

Scaling

You can specify scaling data values when displayed in certain currencies. For example, you can set the scaling for Yen to Thousands, then enter 10,000 as a value for the Japan entity on a data form with the Local member selected for the Currency dimension. When you select Yen as the currency member for the data form, the scaling is applied and 10 displays as the value for Japan.

Number Formatting

You can determine the initial display of numerical values for non-currency and currency data types in data forms:

- Thousands separator:
 - None: 1000
 - Comma: 1,000
 - Dot: 1.000
 - Space: 1 000
- Decimal separator:
 - Dot: 1000.00
 - Comma: 1000,00
- Negative number sign:
 - Prefixed minus: -1000
 - Suffixed minus: 1000-
 - Parentheses: (1000)
- Negative number color:
 - Black
 - Red

Reporting Currencies

A reporting currency is the currency in which your company prepares financial statements. Planning supports currency conversion from local currencies to one or more reporting

currencies. Converted reporting currency values are stored and read-only for all users. An application's default currency is the default reporting currency. You can disable a currency as a reporting currency.

Checking How Currencies are Used

You can view how applications use currency: whether a currency is the default, is used for triangulation currency or by an entity, or has a conversion or exchange relationship with other currencies.

- To see how currencies are used:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select **Currencies**.
 - 3 Select the currency for which you want information.
 - 4 Click **Show Usage**.

Creating Currencies

Select from a predefined list or create your own. You can specify:

- The three-letter code
 - The symbol
 - A description of up to 256 characters
 - The scaling factor to use when values are displayed
 - The triangulation currency to use for currency conversion
 - The alias table to use to display aliases
 - Number formatting, including thousands separator, decimal separator, negative sign, and color
 - Whether it is a reporting currency
- To create currencies:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select **Currency**.
 - 3 Click **Add**.
 - 4 Perform one action:
 - To add a predefined currency, select **Select Standard Currencies**.
 - To create a currency, select **Create New Currency** and specify properties:
 - For Code, enter an abbreviation or identifier of up to three characters.
 - **Optional:** For Description, enter a name, such as Japanese yen.
 - For Symbol, enter a symbol or select a symbol from the list.

- **Optional:** For **Scale**, select how to enter and display the currency. For example, 3 yen represents 3000 yen if scaling is set to thousands.
- **Optional:** For **Triangulation Currency**, select the currency to use as the common third currency for conversion.
- **Optional:** For **Alias Table**, select the alias table to use.
- **Optional:** For **Alias**, enter a name for the currency alias.
- **Optional:** For **Thousands Separator**, select how to display the thousands separator (it must differ from the decimal separator).
- **Optional:** For **Decimal Separator**, select how to display numbers with decimal values (it must differ from the thousands separator).
- **Optional:** For **Negative Sign**, select how to display negative numbers:
 - Prefixed minus: -1000.
 - Suffixed minus: 1000-
 - Parentheses: (1000)
- **Optional:** For **Negative Color**, select the display color.
- **Optional:** Select **Reporting Currency** to set the currency as a reporting currency.

5 Click **Save**.

Editing Currencies

► To edit currencies:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Currency**.
- 3 Select the currency to edit.
- 4 Click **Edit**.
- 5 **Modify properties:**
 - To change the currency's symbol, for **Symbol**, enter or select the symbol.
 - For **Description**, enter a descriptive name.
 - For **Scale**, set how to enter and display the currency.
 - For **Triangulation Currency**, select the currency to use as the common third currency for conversion.
 - **Optional:** For **Thousands Separator**, select how to display the thousands separator (it must differ from the decimal separator).
 - **Optional:** For **Decimal Separator**, select how to display numbers with decimal values (it must differ from the thousands separator).
 - **Optional:** For **Negative Sign**, select how to display negative numbers:
 - Prefixed minus: -1000.

- Suffixes: 1000-
 - Parentheses: (1000)
 - **Optional:** For Negative Color, select the display color.
 - **Optional:** Select Reporting Currency to set the currency as a reporting currency.
 - For **Alias Table**, select an alias table and modify the currency alias.
 - Select or clear **Reporting Currency**.
 - You cannot modify the Code property.
- 6 Click **Save**.

Deleting Currencies

You cannot delete the default currency.

➤ To delete currencies:

- 1 Select **Administration**, then **Dimensions**.
- 2 For **Dimension**, select **Currencies**.
- 3 For **Currency**, select the currency to delete.
- 4 Click **Show Usage** to determine if the currency is the default currency, a triangulation currency, or associated with an entity. You cannot delete a currency that meets these criteria.

If you delete a currency defined in the exchange rate table, it is deleted from the table.

- 5 Click **Close**, **OK**, **Delete**, and **OK**.
- 6 Update and validate business rules and reports.

Specifying Exchange Rates

Use exchange rates to convert values from one currency to another. You can:

- Enable budget preparers in various countries to create plans in other currencies
- Show summary report data in a currency
- Summarize values from multiple currencies into one currency

For example, you might specify yen as the base currency for the Japan entity and US dollars for the United States entity. When you display a data form having values for the Japan entity and the data form's display currency is set to US dollars, the exchange rates for the yen is used to convert the values for Japan to US dollars. If the display currency is set to yen, the exchange rates for US dollars converts values for the United States entity to yen.

To specify exchange rates, you must set up multiple currencies when creating applications.

About Exchange Rate Tables

Each application has a default currency specified when the application is created. When you specify exchange rate tables, only the default currency and triangulation currencies are available as destination currencies. You can enter exchange rates from source currencies to default or triangulation currencies.

You can create multiple exchange rate tables. Each table is typically associated with multiple scenarios, but each scenario can be associated with only one exchange rate table. When creating scenarios, select the exchange rate table for converting currencies.

Enter conversion values between the default currency and currencies defined in the Exchange Rates page. Exchange rate tables span all application time periods, so you can apply exchange rates to all scenarios. When creating or modifying exchange rate tables, you must refresh the application to store them in the plan types. See [“Creating and Refreshing Application Databases” on page 67](#).

Hsp_Rates Dimension

Multiple-currency applications include the Hsp_Rates dimension for storing exchange rates. It includes these members and others that store currency rates:

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

When generating reports or loading data, refer to the Hsp_InputValue member. When loading data, you must load data against the local currency. You need not refer to the Hsp_InputCurrency member.

By default, the Hsp_Rates dimension is set to Dense. You can change this (see [“Optimizing Application Performance” on page 73](#)).

Triangulation

Planning supports currency conversion by triangulation through an interim currency called the triangulation currency. If you modify a currency’s triangulation currency, you must re-enter exchange rates for the triangulation currency property and refresh the application to transfer and store the exchange rates. You cannot select the application’s default currency as a triangulation currency.

Calculation Method

When you input exchange rates for converting between currencies, you can select Multiply or Divide as the calculation method. For example, if you select 1.5 as the rate for converting British Pounds to US dollars, and select multiply as the calculation method, 1 British Pound is converted to 1.5 US dollars.

Setting Up Scenarios

Each scenario/version combination contains data for accounts and other dimensions of each entity. After users enter data for an entity for a scenario and version, they can submit or promote the data for the entity to other users for review and approval.

Task	Topic
Create a scenario.	See “Creating Scenarios” on page 221.
Edit a scenario.	See “Editing Scenarios” on page 221.
Delete a scenario.	See “Deleting Scenarios” on page 222.
Copy a scenario.	See “Copying Scenarios” on page 222.
Delete supporting detail associated with a scenario.	See “Deleting Supporting Detail Associated With a Scenario” on page 170.

About Scenarios

Use scenarios to:

- Apply different planning methods.
- Create forecasts.
- Enter data into scenarios.
- Associate scenarios with different time periods or exchange rates.
- Assign user access rights by scenario.
- Report on scenarios.
- Compare and analyze scenarios.

You can group applications into multiple plans with individual review cycles. Scenarios can cover different time spans.

Time Periods

Assign each scenario a range of years and time periods, and specify the Beginning Balance time period. When users access data forms, they can enter into that scenario only years and periods within the range. Years and periods outside of the range display as read-only. You can modify the time range.

Exchange Rate Table

If an application converts currencies, assign an exchange rate table to the scenario. By assigning different exchange rate tables to scenarios, you can model the effects of currency rate assumptions.

Access Rights

Specify access rights to members of the Scenario dimension for groups or users. Access rights determine users or groups can view or modify data. A user or group can have only one of these access rights: Read, Write, or None. Access rights for a user can be combined based on groups to which the user belongs.

Creating Scenarios

► To create scenarios:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Scenarios**.
- 3 Click **Add**.
- 4 For **Scenario**, enter a name.
- 5 **Optional:** For **Description**, enter a description.
- 6 For **Start Yr.**, **Start Period**, **End Yr.**, and **End Period**, select the time period to associate with the scenario.
- 7 **Optional:** For **Exchange Rate Table**, select an exchange rate table to associate with the scenario.

If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversions.

- 8 **Optional:** For **Alias Table**, select an alias table to associate with the scenario, and enter a description.
- 9 **Optional:** Select **Include BegBal as Time Period** to include the BegBalance time period in this scenario for currency conversion.
- 10 **Optional:** Select **Enabled for Process Management** to include this scenario in process management.
- 11 Click **Save**.

Editing Scenarios

► To modify scenarios:

- 1 Select **Administration**, then **Dimensions**.
- 2 For **Dimension**, select **Scenario**.
- 3 Select the scenario to edit.
- 4 Click **Edit**.
- 5 **Optional:** For **Scenario**, enter a name.
- 6 **Optional:** For **Description**, enter a description.
- 7 For **Start Yr.**, **Start Period**, **End Yr.**, and **End Period**, select the time period to associate with the scenario.
- 8 **Optional:** For **Exchange Rate Table**, select an exchange rate table to associate with the scenario.

If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversion.

- 9 **Optional:** For **Alias Table**, select an alias table to associate with the scenario, and enter the description.
- 10 **Optional:** Select **Include BegBal as Time Period** to include the BegBalance time period in this scenario for currency conversion.
- 11 **Optional:** Select **Enabled for Process Management** to use this scenario in process management.
- 12 Click **Save**.

Deleting Scenarios

When you delete scenarios, all planning units that use the scenario (including data) are deleted. You cannot delete scenarios used in planning units that are started, or assigned to an axis on a data form. You must first remove references to scenarios from data forms and assign different scenarios.

- To delete scenarios:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select **Scenarios**.
 - 3 Select the scenarios to delete. At least one scenario must remain in the application.
 - 4 Click **Delete**.
 - 5 Click **OK**.
 - 6 Update and validate business rules and reports.

Copying Scenarios

Only scenario properties are copied. Data values and access rights associated with the original scenario are not copied to the new scenario.

- To copy scenarios:
 - 1 Select **Administration**, then **Dimensions**.
 - 2 Select **Scenarios**.
 - 3 Select the scenario to copy.
 - 4 Click **Copy Scenario**.
 - 5 For **Copy to Scenario**, enter a name.
 - 6 **Optional:** For **Description**, enter a description.
 - 7 For **Start Yr.**, **Start Period**, **End Yr.**, and **End Period**, select the time period to associate with the scenario.
 - 8 **Optional:** For **Exchange Rate Table**, select an exchange rate table to associate with the scenario.

If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversion.

- 9 **Optional:** For **Alias Table**, select an alias table to associate with the scenario, and enter a description.

- 10 **Optional:** Select **Include BegBal as Time Period** to include the BegBalance time period in this scenario for currency conversion.
- 11 **Optional:** Select **Enabled for Process Management** to include this scenario in process management.
- 12 Click **Save**.

Specifying Versions

Use versions to group data used by applications.

Task	Topic
Create a version.	See “Creating Versions” on page 224 .
Edit a version.	See “Editing a Version” on page 224 .
Delete a version.	See “Deleting Versions” on page 225 .

About Versions

Use the Scenario and Version dimensions to create plans to be reviewed and approved. Each scenario/version combination contains data for accounts and other dimensions of each entity. After users enter data for an entity for a scenario and version, they can submit or promote the data for the entity to other users for review and approval. Use versions to:

- Allow multiple iterations of a plan
- Model possible outcomes based on different assumptions
- Manage dissemination of plan data
- Facilitate target settings

Target and Bottom Up Versions

You can create target and bottom up versions. With bottom up versions, you enter data into bottom level members; parent level members are display-only and do not permit data entry. Parent member values are aggregated from bottom level members.

For target versions, you can enter data for members at any level in the hierarchy. You can use business rules to distribute values from parent members to their descendants. Use target versions to set high-level targets for your plan. Planners working with bottom up versions can reference these targets when they enter plan data.

Target versions use top-down budgeting. Workflow Tasks are not allowed, and children of target members must be blank (for example, #MISSING) to enable data input at the top level. Target members must be set to Store (Dynamic Calc overrides data input with sum of children).

Creating Versions

► To create versions:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Version**.
- 3 Click **Add**.
- 4 For **Version**, enter the name of the version you are adding to the application.
- 5 For **Type**, select the type of version to display for the application:
 - Standard Target - Values are entered from the parent level down.
 - Standard Bottom Up - Values are entered at the lowest member level and aggregated upward.
- 6 **Optional:** For **Description**, enter a description.
- 7 **Optional:** For **Alias Table**, select an alias table to associate with the version, and enter a description.
- 8 **Optional:** Select **Enabled for Process Management** to include this version in process management.
This option is not available for target versions.
- 9 Click **Save**.

Editing a Version

You can change the version name and access rights.

► To modify versions:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Version**.
- 3 Select the version to edit.
- 4 Click **Edit**.
- 5 **Optional:** For **Version**, modify the name of the version.
- 6 **Optional:** For **Typelist**, modify the type of version to display for the application:
 - Standard Target - Values are entered from the parent level down.
 - Standard Bottom Up - Values are entered at the lowest member level and aggregated upward.
- 7 **Optional:** For **Description**, enter a description.
- 8 **Optional:** For **Alias Table**, select an alias table to associate with the version, and enter a description.
- 9 **Optional:** Select **Enabled for Process Management** to include this version in process management.
This option is not available for target versions.
- 10 Click **Save**.
- 11 Update and validate business rules and reports.

Deleting Versions

You cannot delete versions that are used in planning units that are started or are assigned to axes on data forms. You must remove references to versions from data forms and assign another version to axes. At least one version must remain in the application.

► To delete versions:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Versions**.
- 3 Select the versions to delete.
- 4 Click **Delete**.
- 5 Click **OK**.

If planning units are started, you cannot delete versions associated with them.

- 6 Update and validate business rules and reports.

Displaying Versions



► To display versions:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select **Version**.
- 3 For **Display**, select the version types to display.
 - **Standard Target**: Values are entered from the parent level down.
 - **Standard Bottom Up**: Values are entered at the lowest member level and aggregated up.

Sorting Version and Scenario Members

You can sort version and scenario members in ascending or descending order. Sorting members affects the Essbase outline itself.

► To sort version and scenario members:

- 1 On **Dimensions**, select the **Scenario** or **Version** dimension.
- 2 For **Sort**:
 - To sort by ascending order, click .
 - To sort by descending order, click .
- 3 Click **OK**.

The next time you create or refresh the database, the Essbase outline is generated with members placed in the same order as on the Dimensions tab.



Moving Scenario and Version Members in the Dimension Hierarchy

You can change the order of scenario and version members in the dimension hierarchy.

► To move a member's position in the dimension hierarchy:

1 On **Dimensions**, select the scenario or version to move.

2 Perform one action:

- To move the member up, click .
- To move the member down, click .

3 Click **OK**.

The next time you create or refresh the database, the Essbase outline is generated with members placed in the same order as displayed on the Dimensions tab.

Setting up Dynamic Time Series Members

You can use Dynamic Time Series (DTS) members to create reports that show period-to-date data, such as quarter-to-date expenses. DTS members are created automatically during application creation, and can be used with members of the Period dimension. To set up DTS, you enable a predefined DTS member and associate it with a generation number (and, optionally, an alias table and alias name). For example, to calculate quarter-to-date values, you can enable the Q-T-D member and associate it with generation number 2. You can then use the Q-T-D DTS member to calculate monthly values up to the current month in the quarter.

Planning provides eight predefined DTS members:

- H-T-D: History-to-date
- Y-T-D: Year-to-date
- S-T-D: Season-to-date
- P-T-D: Period-to-date
- Q-T-D: Quarter-to-date
- M-T-D: Month-to-date
- W-T-D: Week-to-date
- D-T-D: Day-to-date

Caution! Oracle recommends that you perform a backup before using the DTS feature. See [“Backing Up Applications and Application Databases” on page 74](#). If you are using the Y-T-D or P-T-D member, you must rename the Year or Period dimension so it does not conflict with the reserved Dynamic Time Series generation names, Year and Period. Before using Y-T-D, rename the Year dimension; before using P-T-D,

rename Period. After doing so, you must update all application artifacts affected by these changes, such as member formulas and business rules, and any reports that reference the dimension by name.

The DTS members provide up to eight levels of period-to-date reporting. Your data and database outline determine which members you can use. For example, if the database contains hourly, daily, weekly, monthly, quarterly, and yearly data, you can report day-to date (D-T-D), week-to-date (W-T-D), month-to-date (M-T-D), quarter-to-date (Q-T-D), and year-to-date (Y-T-D) information. If the database contains monthly data for the past 5 years, you can report year-to-date (Y-T-D) and history-to-date (H-T-D) information, up to a specific year. If the database tracks data for seasonal time periods, you can report period-to-date (P-T-D) or season-to-date (S-T-D) information.

It is recommended 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. For detailed information, see the *Oracle Essbase Database Administrator's Online Help*.

► To set up Dynamic Time Series members:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the **Period** dimension, and click **DTS**.
- 3 Select **Enabled** for the DTS series to use: **H-T-D, Y-T-D, S-T-D, P-T-D, Q-T-D, M-T-D, W-T-D, or D-T-D**.
- 4 Select a **generation**.

The number of generations displayed depends on the number of generations in the time dimension. You cannot associate DTS members with the highest generation (the dimension root).

- 5 **Optional:** Select an alias table and type an alias name. (If necessary, resize the window to view the fields.) See [“Working with Alias Tables” on page 185](#).
- 6 Click **Save**.

Additional Supported Essbase Features

Additional Essbase features are now supported for Planning dimensions. You can add children and siblings to Scenario, Version, and Period dimensions, and you can use the cut, paste, expand, and collapse features to work with their dimension hierarchies (see [“Working with Dimension Hierarchies” on page 188](#)). You can also use shared members for these dimensions, and can set two pass calculations at the root level for all dimensions. For example, you can:

Feature	More Information
In Scenario and Version dimensions, create hierarchies and use shared members. If you assign children to bottom-up versions, these versions display as read-only parents on data forms.	See “Setting Up Scenarios” on page 220 and “Specifying Versions” on page 223 .
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 Years” on page 211 , “Editing the BegBalance Member” on page 212 , and “Editing Exchange Rate Tables” on page 70 .
Turn on two pass calculation at the root level, for example, for Account. Caution! Two pass calculation is ignored on any non-Account member not set to Dynamic Calc. When using this setting, consider the impact on currency conversion scripts.	See “Adding or Editing Members” on page 201 .
For attributes, create hierarchies and assign aliases.	See “Working with Attributes” on page 205 .

Note: For multicurrency applications, the pre-built currency conversion calc scripts do not function properly if you change data storage to dynamic for any Scenario, Version, Period, or Year member, or for certain dimension roots (such as Entity, Version, Currency, and custom dimensions). When changing data storage, consider the impact on currency conversion scripts.

Considerations for Alternate Hierarchies in Period Dimensions

When using alternate hierarchies, the parent for an alternate hierarchy member can either be the root member of the Period dimension, such as Periods, or another alternate hierarchy member.

Working with Classic Applications

You can create and update Classic applications if you are assigned the Shared Services Dimension Editor and Planning Application Creator roles. For detailed information, see the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.

Refreshing Application Databases Using a Utility

The CubeRefresh utility creates or refreshes the database. Before it runs, it ensures that the application is not locked. While it runs, the application is locked when metadata is updated in the database. For example, if users are assigning access, this message displays: “Cannot process your request because the application is being refreshed.” This utility must be run on the same machine as the Planning server, not on a remote server.

Caution! Oracle recommends that you back up the application before creating or refreshing. See [“Backing Up Applications and Application Databases” on page 74](#). Following these steps affects data in the database. When you use the create or refresh options, data might be replaced or erased, and Planning plan types rebuilt. For important information, see [“Considerations for Working with Essbase” on page 28](#).

► To refresh the application database using a utility:

1 Back up the application. See [“Backing Up Applications and Application Databases” on page 74](#).

2 Locate the `CubeRefresh.cmd` utility by navigating to the `bin` folder with the command prompt.

If you installed Planning to the default location, the file is in: `HYPERION_HOME/products/Planning/bin`.

3 Launch `CubeRefresh.cmd` by entering the command and its parameters in the command line:

```
CubeRefresh.cmd [-f:passwordFile] /A:application_name /U:user_name [/C|/R] /D [/F[S|V]] [/RMIPORT:rmi_port] [/L] [/DEBUG]
```

- **Optional:** If an encrypted password file is set up, you can specify `[-f:passwordFile]` as the first parameter in the command line, where `passwordFile` is the full file path and name for the password file. See [“Suppressing Password Prompts in Planning Utilities” on page 37](#).
- **application_name:** The name of the Planning application on which to run the create or refresh.
- **user_name:** The administrative user who has rights to create or refresh the application.
- **/C or /R:** The function to perform on the application:
 - **/C:** Create the database outline.
 - **/R:** Refresh the database outline.
- **/D:** Specify the database during create or refresh.
- **/F:** Use security filters for all users of the application. Optionally, use with **S**, **V**, or **SV**:
 - **/FS:** Generate shared member security filters.
 - **/FV:** Validate security filters, but do not save the filters.
 - **/FSV:** Validate the shared member security filters.
- **/RMIPORT:** Specify an RMI port number different from the preset value of 11333.
- **/-L:** Default option, to create or refresh by connecting to an application server on a local or remote machine. (You can use **/L** to create or refresh without connecting to an application server, for example, if the application server is not running.)
- **/DEBUG:** Specify detailed error messages.

4 If prompted, enter your password.

5 View the result of the application refresh, including errors. Completion status is shown on the console.

This example shows a command line used to refresh a database for all users logged on to the specified application. It uses the default RMI port and includes shared member security filters:

```
CubeRefresh.cmd /A:appl /U:admin /R /D /FS
```

Refreshing Databases with Level 0 Members Set to Dynamic Calc

If a level 0 member in the database outline is set to Dynamic Calc or Dynamic Calc and Store, Planning successfully creates or refreshes, even if the members are not associated with a member formula. Planning adds a placeholder formula for members that do not currently have a member formula.

Working With Essbase Partitions

If you use Essbase Partitioning, partitions can access data shared between databases on different applications or servers. For information on partitions, see the *Oracle Essbase Database Administrator's Guide*.

Before refreshing application databases in Planning, remove the Essbase partition definitions. This prevents overwriting the Essbase partitions or corrupting the Essbase database. After refreshing, you can redefine the Essbase partitions.

If you use replicated partitions, you can create the partition definition at replication time, run the partition, and drop the partition definition. You need not remove the partition definition during refresh because it is created when the partition is run.

Sorting Members Using a Utility

You can use the Planning Sort Member utility to sort dimension members. `SortMember.cmd` is similar to sorting on the Dimensions tab in Planning. You can sort Entity, Account, Scenario, Versions, and user-defined custom dimensions. You cannot sort Period, Year, or Currency dimension members. The utility is useful for sorting dimension members after loading members into Planning. The `SortMember.cmd` utility uses a command line interface. Only administrators can run it.

► To launch the `SortMember.cmd` utility:

- 1 From the `bin` directory on the server Planning is installed, enter this syntax:

```
SortMember [-f:passwordFile] servername username application member  
children|descendants ascend|descend
```

Table 69 SortMember Utility Parameters

Parameter	Description
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37.

Parameter	Description
<i>servername</i>	The server name where the Planning application resides
<i>username</i>	The name of the Planning administrator
<i>application</i>	The name of the Planning application containing the dimension members to sort
<i>member</i>	The parent member whose children or descendants to sort
children descendants	Whether to sort by children or descendants: sorting by children affects only members in the level immediately below the specified member; sorting by descendants affects all descendants of the specified member
ascend descend	Whether to sort by ascending order or by descending order

For example:

```
SortMember localhost admin BUDGET account200 descendants ascend
```

2 If prompted, enter your password.

If the application server, HBR Service, or RMI service is running when the SortMember utility is launched, you might see some java.rmi or “Port already in use” errors. They do not affect the functionality of this utility.

Deleting Shared Descendants Using a Utility

Use the DeleteSharedDescendant utility to delete shared dimension members that are descendants of a given member. You can delete shared Entity, Account, and user-defined dimension members. All shared descendant members are deleted, not just immediate children of the specified member.

Administrators run this utility using a command line interface. If the application server, HBR Service, or Oracle RMI service is running when the utility is launched, you may see java.rmi or “Port already in use” errors. They do not affect the functionality of the utility.

► To use the DeleteSharedDescendants utility:

1 Launch the DeleteSharedDescendants.cmd file from the bin directory on the server where Planning is installed, using this syntax:

```
DeleteSharedDescendants [-f:passwordFile] servername username  
application member
```

If you installed Planning in the default location, the bin directory is in: *HYPERION_HOME*/products/Planning/bin.

Table 70 DeleteSharedDescendants Utility Parameters

Variable	Description
<code>[-f:passwordFile]</code>	Optional: If an encrypted password file is set up, use as the first parameter in the command line to read the password from the full file path and name specified in <code>passwordFile</code> . See “Suppressing Password Prompts in Planning Utilities” on page 37 .
<code>servername</code>	The server name on which the Planning application resides.
<code>username</code>	The name of the Planning administrator
<code>application</code>	The name of the Planning application containing the shared dimension members to delete.
<code>member</code>	The member whose shared descendants to delete. If a member name contains a space, surround it by quotation marks (for example, “Member One”). If the shared member itself is specified in the command line, the message “No shared descendants of <code>member_namewere</code> found” is displayed.

For example:

```
DeleteSharedDescendants localhost admin BUDGET account200
```

- 2 If prompted, enter your password.
- 3 To see the results of running the utility, check the log files that are generated in the same directory as the utility:
 - `DeleteSharedDescendants.log`: contains status messages.
 - `DeleteSharedDescendantsExceptions.log`: contains error messages.

If you run this utility while the Manage Database task is open, no error message is displayed, but the member is not deleted. The LOG and CMD files in the `bin` directory show that one shared member was found, but 0 shared members were deleted.

Using Smart Lists, UDAs, and Member Formulas

For Classic applications, you can create and update Smart Lists, UDAs, and member formulas.

Working with Smart Lists

Administrators use Smart Lists to create custom drop-down lists that users access from data form cells. When clicking into cells whose members are associated with a Smart List (as a member property), users select items from drop-down lists instead of entering data. Users cannot type in cells that contain Smart Lists. Smart Lists display in cells as down arrows that expand when users click into the cells.

Perform these tasks to create and administer Smart Lists:

- Define Smart Lists, described here.
- Associate Smart Lists with members.
- Select dimensions for which Smart Lists are displayed.

- Optionally:
 - Turn Smart Lists on or off for data forms.
 - Use Smart List values in member formulas.
 - Set how #MISSING cells associated with Smart Lists display in data forms.

➤ To create or work with Smart Lists:

1 Select Administration, then Manage Smart Lists.

2 Perform one action:

- To create a Smart List, click **Create**, enter the name, and click **OK**.
- To change a Smart List, select it and click **Edit**.
- To delete Smart Lists, select them, click **Delete** and **OK**.

Data cells can display only one Smart List. If multiple Smart Lists intersect at cells, set which one takes precedence.

Adding or Changing Smart List Properties

Use the Edit Smart List Properties tab to set Smart List properties.

➤ To set Smart List properties:

1 Select Administration, then Manage Smart Lists.

2 Select a Smart List and click Edit.

3 Define Smart List properties on Properties:

Table 71 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. Smart List labels can reference a resource, which can be translated into different languages. See “About Customizing Text, Color, and Images” on page 248.
Display Order	How Smart Lists are sorted in the drop-down list: by ID, Name, or Label
#MISSING Drop-Down Label	Enter a label (for example, “No Justification”) to be displayed as an entry in the Smart List whose value is #MISSING. Notes: <ul style="list-style-type: none"> ● It displays as the first selection in the Smart List drop-down, allowing #MISSING as a selection in the data form. ● When the cell is not in focus, this label displays only if Drop-Down Setting is selected in the next option. Otherwise, #MISSING or a blank cell is displayed, depending on the Display Missing Values As Blank selection for the data form.

Property	Description
	<ul style="list-style-type: none"> ● #MISSING labels determine only the display of cells with #MISSING data; #MISSING remains the stored value.
#MISSING Data Form Label	<p>Determines how #MISSING values are represented in cells associated with Smart Lists. Options:</p> <ul style="list-style-type: none"> ● Drop-Down Setting: Displays the label set in #MISSING Drop-Down Label. ● Data Form Setting: Displays #MISSING or leaves cells blank, depending on the Display Missing Values As Blank selection for the data form. This selection determines what is displayed in the cell when it is not the focus. When the cell is in focus, the Smart List item that is selected from the drop-down is displayed.
Automatically Generate ID	Generate a numeric ID for each Smart List entry. If you do not select this option, you can customize Smart List ID values.

4 Click **Save**.

5 Select **Entries**.

Use the Entries tab to define selections on Smart Lists.

Adding or Changing Smart List Entries

Use the Edit /Add Smart Lists Entries tab to define the selections in the Smart List.

► To define Smart List entries:

1 Select **Administration**, then **Manage Smart Lists**.

2 Select a Smart List and click **Edit**.

3 On **Entries**, define drop-down list items:

- For first items only: enter information into the first row.
- To add an item, click **Add** and enter the information.
- To delete an item, select it and click **Delete**.
- To edit an item, change the information in its row:

Table 72 Smart List Entries

Entry Property	Description
ID	Unique number that sets the order for the displayed entry. Customizable only if Automatically Generate ID is not selected on the Properties tab.
Name	Unique alphanumeric name containing alphanumeric and underscore characters (for example: Customer_Feedback) and no special characters or spaces
Label	Displayed text for the Smart List entry on the drop-down list (for example: Customer Feedback).

Items highlighted in red are duplicates.

4 Perform one action:

- Click **Save**.

- Select **Preview**.

Previewing Smart Lists

Preview the defined Smart List on the Preview tab. The tab shows the Smart List as displayed in a drop-down list or a table.

Displaying #MISSING with Smart Lists

Administrators set values displayed in Smart Lists and data cells, including the display when no data is in the cell. Cells can display no value, #MISSING, or (for cells associated with Smart Lists) a specified value.

Use these options to control the display of #MISSING when cells are not in focus:

Option	Guideline
Blank	When designing data forms, select Display Missing Values as Blank. When setting Smart List properties, select Data Form Setting.
#MISSING	When designing data forms, do not select Display Missing Values as Blank. When setting Smart List properties, select Data Form Setting.
A custom label, such as "No Change"	When setting Smart List properties, enter the custom label in the #MISSING Drop-Down Label field (for example, No Change). Select Drop-Down Setting.

Working with UDAs




You can use user-defined attributes (UDAs), descriptive words or phrases, within calc scripts, member formulas, and reports. UDAs return lists of members associated with the UDA. For example:

- You can use the HSP_UDF UDA to prevent a formula from being overwritten when the application is refreshed. You must log on to each database associated with the Planning application and create the member formula with a UDA. The syntax for this UDA is:
(UDAs: HSP_UDF).
- If you use the @XREF function to look up a data value in another database to calculate a value from the current database, you can add the HSP_NOLINK UDA to members to prevent the @XREF function from being created on all plan types that are not the source plan type selected for that member.
- For a Product dimension with several product members, you can create a UDA called New Products and assign this UDA to the new products in the Product dimension hierarchy. Then you can base certain calculations on the designation New Products.

For more information on creating and using UDAs, see [“Considerations for Working with Essbase” on page 28](#) and *Oracle Essbase Database Administrator's Online Help*.

UDAs are specific to dimensions. For example, creating a UDA for an Account member makes it available for non-shared Account members. Deleting it removes it for all Account members. To make UDAs available for multiple dimensions, create the same UDA for multiple dimensions. For example, create a UDA named New for Account and Entity dimensions to make the UDA named New available for Account and Entity members.

➤ To select UDAs for members:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the dimension for whose members to associate the UDA.
- 3 From the dimension hierarchy, select a member and click **Edit**.
- 4 Select **UDA**.
- 5 **Optional:** To create a UDA, click **Add**.
- 6 Select UDAs for the member by moving them to **Selected UDA** and clicking **Save**:
 -  move selected UDAs
 -  remove selected UDAs
 -  remove all UDAs
- 7 **Optional:** To undo changes, click **Refresh**.

Creating UDAs

➤ To create UDAs:

- 1 Navigate to the UDA tab.
- 2 On **UDA**, click **Add**.
- 3 Enter a name and click **Save**.

Changing UDAs

➤ To change UDAs:

- 1 Navigate to the UDA tab.
- 2 On **UDA**, select a UDA and click **Edit**.
- 3 Change the name and click **Save**.

Deleting UDAs

Deleting a UDA removes it for the whole dimension.

➤ To delete UDAs:

- 1 Navigate to the **UDA** tab.
- 2 Select the UDA and click **Delete**.

If you delete UDAs, you must update all member formulas, calc scripts, and reports that reference them.

Working with Member Formulas

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

- Operator type, function, value, member name, UDA, and so on allowed in Essbase formulas (see the *Oracle Essbase Database Administrator's Guide*).
- Predefined Planning formula expressions, including Smart List values, that expand into a formula or value upon database refresh.

➤ To define member formulas:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the dimension for whose member to add or change a formula.
- 3 Select the member.
- 4 Select **Member Formula**.
- 5 In the text box, define formulas for the member.

See the *Oracle Essbase Database Administrator's Guide* for syntax, rules, and usage on Essbase formulas.

- 6 **Optional:** To check the validity of the member formula, click **Validate Member Formula** and **OK**.
- 7 Click **Save**.

Before you click Save, clicking Refresh restores the previous member formula information.

Viewing Details of Formula Validation

➤ To view details of the member formula validation:

- 1 On **Member Formulas**, click **Validate Member Formula**.
- 2 If the member formula is not valid, click **Show Details**.

If the member formula is valid, Show Details is not selectable.

- 3 Click **OK**.

Working with Formula Expressions

Planning member formulas support Essbase native formulas and Planning formula expressions that are evaluated and expanded into Essbase code blocks when the database is refreshed. In these expressions, you can address Smart Lists by name, which Planning replaces with their numerical values in calculations.

In the text box on the Member Formula tab, you can include predefined formula expressions in member formulas, and test them with the Validate Member Formula button. You can also load them.

You can update the dimension outline without updating the business rules and calc scripts that depend on the outline. Calculations become more independent of specifics in the outline. You can use Smart Lists as objects in calculations. Performance is not decreased when you use formula expressions because they are run only when you refresh the database.

► To use a formula expression in a member formula:

- 1 Select **Administration**, then **Dimensions**.
- 2 Select the dimension for whose member to add or change a formula.
- 3 Select the member and click **Edit**.
- 4 Select **Member Formula**.
- 5 In the text box, define formulas for the member.

You can include Planning formula expressions and Essbase native formulas in the member formula. See the *Oracle Essbase Database Administrator's Guide* for syntax, rules, and usage on Essbase native formulas.

Planning provides predefined formula expressions that you can use in member formulas. You cannot edit or create your own formula expressions.

- 6 **Optional:** To check the validity of the member formula, click **Validate Member Formula** and **OK**.
- 7 **Optional:** If there are errors in the member formula, click **Show Details** to view a description.
- 8 **Optional:** Click **Refresh** to restore the previous member formula if you do not want to save the changes you made to the member formula.
- 9 Click **Save**.

Prerequisites

Before using formula expressions in member formulas, you should understand Essbase formulas and calculation and the application outline. See the *Oracle Essbase Database Administrator's Guide*.

Syntax

Member formula expressions support functions and variables. Follow these syntax rules for functions and variables when you create formula expressions:

- Enclose variables or properties with square brackets, []. If you omit square brackets, the variable is treated as a native variable.
- Enclose member names with quotation marks.
- Characters in variables are case-insensitive, but cannot have extra spaces or characters such as underscore (_).
- You can include subcalls to other functions within an expression.
- Do not enter text where a number is required.
- The order of the outline is important in a member formula. For example, do not reference a value that has not been calculated yet.

Including Smart List Values as Variables

You can include a Smart List as a variable in a formula expression, such as the formula expression, “Status”= [Status.Departed]

“Status” is the member name, Status is the Smart List name, and Departed is a Smart List entry. If the Smart List ID for Departed is 2, Essbase replaces Status.Departed with 2 in the member formula (Planning treats Smart Lists as numbers). If the Smart List ID for Departed is 2, Essbase puts 2 in the calculation and stores 2 in the database.

Write Smart Lists in this format: [SmartListName.SmartListEntry]

Formula Expressions

Planning formula expressions can include these predefined variables and functions.

Table 73 Variables in Formula Expressions

Variable	Description
OpenInputValueBlock	Generates an IF statement if the Planning application is a multicurrency application, or an empty string if it is one currency application. Used with ClosedInputValueBlock.
CloseInputValueBlock	Generates an End IF statement if the Planning application is a multicurrency application, or an empty string if it is one currency application. Used with OpenInputValueBlock.
NumberOfPeriodsInYear	Returns the number of time periods in the year
NumberOfYears	Returns the number of years in the application

Table 74 Functions in Formula Expressions

Function	Description
Dimension(dimTag)	Returns the name of a predefined dimension. The dimtags are: <ul style="list-style-type: none"> ● DIM_NAME_PERIOD ● DIM_NAME_YEAR] ● DIM_NAME_ACCOUNT ● DIM_NAME_ENTITY

Function	Description
	<ul style="list-style-type: none"> ● DIM_NAME_SCENARIO ● DIM_NAME_VERSION ● DIM_NAME_CURRENCY
Period(periodName)	Returns the specified period. The periodName options are: <ul style="list-style-type: none"> ● FIRST_QTR_PERIOD ● SECOND_QTR_PERIOD ● THIRD_QTR_PERIOD ● FOURTH_QTR_PERIOD ● FIRST_PERIOD ● LAST_PERIOD
CrossRef(accountName)	Generates a cross reference for the account
CrossRef(accountName, prefix)	Generates a cross-reference for the account. The account name contains a prefix that you define. The default prefix is No, followed by a blank space and the account name, for example, No Salary.
getCalendarTPIndex()	Generates a member formula that returns an index for the time period; the index is based on the calendar year.
getFiscalTPIndex()	Generates a member formula that returns an index for the time period; the index is based on the fiscal year.
CYTD(memberName)	Generates a calendar year-to-date formula for the member
CYTD(memberName, calTpIndexName, fiscalTpIndexName)	Generates a calendar year-to-date formula for the member, and the time period index based on the calendar year and fiscal year. Use when members are renamed. The default member names are “Cal TP-Index” and “Fiscal TP-Index”

Understanding Common Errors

Follow the rules of syntax carefully. If formula expression syntax contains errors, error messages are returned after you validate the member formula. To get information about error messages, click Show Details on the Member Formula tab. The most common error message is “Failed to execute.” This occurs when you use parameters in the expression incorrectly. These actions cause “Failed to execute” error messages:

- Entering the wrong number of parameters in the formula expression
- Misspelling member names, functions, or variable names
- Not surrounding member names with quotation marks
- Including numbers where strings are required



Customizing Planning Web Client

In This Chapter

Customizing Reports	241
Customizing the Planning Tools Page	243
Customizing Cascading Style Sheets	243
Customizing the Style of Row and Column Members in Data Forms	247
About Customizing Text, Color, and Images	248
Customizing Colors, States, and Actions for Process Status	251
Creating Spreading Patterns	253
Examples of Spreading Patterns	254

Customizing Reports

Planning includes templates that control the layout and content of PDF reports of data forms, data form definitions, task lists, and planning units. You can use the templates as is. You can also customize the templates to add company logos, and special formatting for features such as shading, page size, orientation, font, font size, headers, percentage of page used for headers, number of data columns per page, and precision.

To customize reports, you must install and configure Microsoft Office Word 2000 or later and Oracle Business Intelligence Publisher Desktop. You can then use Word's BI Publisher menu to update information in the report using a sample .XML file. You can also use Word features to customize formatting. You make the template available by saving the .RTF file with the appropriate name, and placing it in the classpath or in the `HspJS.jar` file.

You can customize four report types, using the corresponding sample and template files. For information on creating the reports, see the related topics.

Report Type	Sample File Name	Template Name	Related Topics
Data Form	PlanningFormSample.xml	PlanningFormTemplate.rtf	See “Creating Data Forms” on page 110 .
Data Form Definition	PlanningFormDefSample.xml	PlanningFormDefTemplate.rtf	See “Printing Data Form Definitions” on page 137 .
Task List	PlanningTaskListSample.xml	PlanningTaskListTemplate.rtf	See the <i>Oracle Hyperion Planning User's Online Help</i> .

Report Type	Sample File Name	Template Name	Related Topics
Planning Unit Annotations	PlanningPUAnnotationSample.xml	PlanningPUAnnotationTemplate.rtf	See “Printing Planning Unit Annotations” on page 149 .

This topic gives general customization instructions. For detailed procedures, see the documentation installed with Word and BI Publisher. The BI Publisher install guide and user's guide are also available here:

http://download.oracle.com/docs/cd/E10091_01/welcome.html

► To install BI Publisher Desktop:

- 1 Download the most recent version of BI Publisher Desktop from this location:

<http://www.oracle.com/technology/software/products/publishing/index.html>

To customize reports, you only need to install BI Publisher Desktop. BI Publisher is not required.

- 2 Save the zip file to your drive, and extract it with Use Folder Names selected.
- 3 Close all Microsoft Office applications.
- 4 Navigate to the directory where you extracted the zip file, and double-click the `setup.exe` application.
- 5 Follow the instructions in the BI Publisher installation wizard to perform a basic installation, accepting the default settings.

► To customize reports:

- 1 Open the Planning `HspJS.jar` file, installed by default in `WEB-INF/lib`, and extract the sample file and corresponding template file.

For example, to customize the task list report, extract these files:
`PlanningTaskListSample.xml` and `PlanningTaskListTemplate.rtf`.

- 2 Save the sample and template files to a location where you can open them later.
- 3 In Microsoft Word, open the .RTF template file for the report to customize.

For example, for task list reports, open the `PlanningTaskListTemplate.rtf` template file.

- 4 From Microsoft Word's **Oracle BI Publisher** menu, select **Data**, then **Load Sample XML Data**, and open the sample file for the report.

For example, for task list reports, open the `PlanningTaskListSample.xml` sample file.

If Word does not display the BI Publisher menu, select **View**, then **Toolbars**, and then **Template Builder**. See the *Oracle Business Intelligence Publisher User's Guide*.

- 5 Use the Word menus to customize the template.

For example, you can insert graphics and update fonts. For assistance with these tasks, see the Word documentation.

- 6 **Optional:** To update fields in the report, use Word's BI Publisher menus to add fields from the sample file.

For example, select Oracle BI Publisher, then Insert, and then Field, click a field in the Field dialog box, and drag the field into the template.

For assistance with these tasks, see the BI Publisher documentation. (In Word, select Oracle BI Publisher, then Help. In Oracle Business Intelligence Publisher, click Help to view online help.)

- 7 When you finish customizing, select **Oracle BI Publisher**, then **Preview Template**, and select a format for previewing your changes. You can preview in any format. Click the close box to close the preview file.
- 8 In the template file, select **Save As**, and save the template as an .RTF file with the appropriate file name from the table.

For example, if you are customizing task list reports, save the file as `PlanningTaskListTemplate_Custom.rtf`.

Type of Template	File Name
All data forms	<code>PlanningFormTemplate_Custom.rtf</code>
Specific data form	<code>PlanningFormTemplate_Data_Form_Name</code> For example, to apply the template to a data form called Income, save the template as <code>PlanningFormTemplate_Income.rtf</code> .
Data form definition	<code>PlanningFormDefTemplate_Custom.rtf</code>
Task list status reports	<code>PlanningTaskListTemplate_Custom.rtf</code>
Planning unit annotation reports	<code>PlanningPUAnnotationTemplate_Custom.rtf</code>

You must save the template file in a location on the classpath so it is available to the Web application server. For example, if you are using Tomcat, you could save the files to this location: `WEB-INF/classes`. You can also insert the template file in the `HspJS.jar` file at the root level (the same level as the template files).

- 9 Restart the application server to make the custom template available when users create PDF reports in Planning.

Customizing the Planning Tools Page

Administrators can customize the Planning Tools page by adding links to commonly used tools for analyzing, tracking, and reporting on planning data. Planning users can then open Planning links from the Tools menu and click links to open pages in secondary windows in their browser. See [“Specifying Custom Tools” on page 169](#).

Customizing Cascading Style Sheets

In Planning, cascading style sheets help define user interface (UI) themes, which control the appearance of Planning. You can customize a Planning cascading style sheet to meet your needs. Some common style sheet customizations include:

- Changing font colors to add emphasis or create a color-coding scheme

- Adjusting the background color of certain UI elements
- Inserting your organization's logo

Before customizing cascading style sheets, ensure that you have basic knowledge of cascading style sheets and are familiar with style sheet editor tools. Some settings are necessary for displaying portions of the product.

Changes to cascading style sheets are made on the Web application server, so they affect all connected users. Style sheets are in the *DeploymentDirectory* for the Web application server. For information on where to find this directory for various Web application servers, see [“Default Location of Cascading Style Sheet Files” on page 245](#).

Planning primarily uses two cascading style sheets: a global sheet that defines Web UI elements common to multiple Oracle products, and one specific to Planning. The global cascading style sheet is called `global.css`. This table lists the major sections in the `global.css` file, and shows the sections that Oracle strongly recommends against customizing.

Table 75 Major Sections in the `global.css` File

Customizable Sections	Sections Not Recommended for Customizing
Non-Navigation Styles	Tadpole Menubar Tadpole MenuItem Menu Styles (order is important)
Tadpole Masthead	Tadpole Minimized Masthead
Tadpole Content Area	Tadpole Toolbar
Tadpole Logon Styles	Tadpole Viewpane
Tabs Tab Anchor Tags Tab Body	Tadpole Tree
TaskTabs Task Tab Body	
Groupbox Groupbox Anchor Tabs	
Buttons When Drawing Buttons in HTML When Using Button Tag When Using Input Tag	

► To customize a cascading style sheet:

- 1 **Locate the `HspCustom.css` file in `DeploymentDirectory/custom`.**

The path for *DeploymentDirectory* depends on your Web application server. See [“Default Location of Cascading Style Sheet Files” on page 245](#).

- 2 **Customize the `HspCustom.css` file, and save the updated file in the same directory, `DeploymentDirectory/custom`.**

All Planning Web pages refer to the `HspCustom.css` file, and settings made here override those in `global.css` and `planning.css`.

To customize styles in data forms to reflect members' hierarchical positions within the row or column axis, see [“Customizing the Style of Row and Column Members in Data Forms” on page 247](#).

Default Location of Cascading Style Sheet Files

The cascading style sheets for Planning are located by default in: *DeploymentDirectory/ui_themes/tadpole*. The path for *DeploymentDirectory* depends on your Web application server. For example, for WebLogic: *HYPERION_HOME/products/Planning/AppServer/InstallableApps/Common/HyperionPlanning.ear*. For detailed information, see the *Oracle Hyperion Enterprise Performance Management System Manual Deployment Guide*.

Modifying Cascading Style Sheets

When working with Oracle cascading style sheets:

- Before making changes, copy the original version and make it easily accessible.
- Styles common across multiple products are in `global.css`.
- Styles for products requiring unique styles are in `productname.css`.
- Some user interface controls are components of third-party technologies. Changing styles of third-party controls requires changing all associated files.
- Avoid making changes to styles that affect layout, including Padding, Borders, Text and Vertical Align, Position, and White-space.
- Changing `color` affects font color. Changing `background` affects the color of elements such as buttons.
- To change hyperlink text color, use the hyperlink style (`a:link`, `a:hover`, and `a:visited`).
- Toolbar button changes may require changes to all buttons in the product.
- Toolbar buttons rely on a dark-blue background to display properly.

Customization Examples

This section provides topics on common customizations:

- [“Including Your Logo” on page 246](#)
- [“Changing the Color of the Content Area Background” on page 246](#)

- “Changing the Color of Hyperlinks and Visited Hyperlinks” on page 247

Including Your Logo

You can update the `global.css` file to use your organization’s logo or image.

➤ To include your logo:

- 1 Place the image in this directory:

`WebAppServer/ApplicationDirectory/ui_themes/tadpole/images_global`

- 2 Open the `global.css` file using a style sheet editor tool.

- 3 In the Tadpole Masthead section of the `global.css` file, change the reference to the image:

```
.masthead TD.co_brand {
background-image: URL("images_global/image_name.gif");
```

- 4 Open the `BannerTop.jin` file, and make the same change to the `<td>` tag that has this comment:

Insert Co-Brand Logo onto this TD.

```
<!--Insert Co-Brand Logo onto this TD -->
```

```
<td class="co_brand logo_padding">&nbsp;
```

```

```

```
</td>
```

- 5 To change the image on the Logon screen to your organization’s logo or image, replace the `logon_panel.jpg` file with the image in this directory:

`WebAppServer/ApplicationDirectory/ui_themes/tadpole/images_global`

Changing the Color of the Content Area Background

You can change the background color of the content area on the right side of the page by editing the Tadpole Logon Styles section of the `global.css` file. This table shows how to edit files to change the default color to white.

Table 76 Changing the Color of the Content Area Background

Default Content Area Background	White Content Area Background
<pre>.content table.content { background: #e5eaeef; }</pre>	<pre>.content table.content { background: #ffffff; }</pre>

Changing the Color of Hyperlinks and Visited Hyperlinks

You can change hyperlink color by editing the Non-Navigation Styles section of the `global.css` file. This table shows how to edit the file to change the default color to red.

Table 77 Changing the Color of Hyperlinks and Visited Hyperlinks

Black Hyperlinks and Visited Hyperlinks	Red Hyperlinks and Visited Hyperlinks
<pre>a:link, a:visited { color: #000000; }</pre>	<pre>.content table.content { background: #ff0000; }</pre>

Customizing the Style of Row and Column Members in Data Forms

You can customize Planning cascading style sheets to display data forms with different styles for the levels of members in rows and columns, depending on their hierarchical position in the data form. Note:

- Customized styles are applied regardless of the dimension displayed in data forms.
 - Text alignment in rows is not supported.
 - Styles are based on members' displayed hierarchical level, not necessarily Essbase relationships.
 - Setting large font sizes on a row impacts the row's alignment.
 - Customized styles are not reflected when data forms are printed to PDF files.
- To customize the style of row and column members:
- 1 Modify the `planning.css` file using the instructions in [“Customizing Cascading Style Sheets” on page 243](#).
 - 2 In the `planning.css` file, customize these header tags:

Table 78 Header Tags in Planning.css File

Rows	Columns
rowHeader_0	columnHeader_0
rowHeader_1	columnHeader_1
rowHeader_2	columnHeader_2
rowHeader_3	columnHeader_3
rowHeader_4	columnHeader_4

The tags `rowHeader_0` and `columnHeader_0` affect the lowest-level class. The tags `rowHeader_4` and `columnHeader_4` affect the highest-level class. Members above level 4 display with the level 4 style.

About Customizing Text, Color, and Images

You can customize text, colors, and images in the Planning Web interface, and you can localize text.

- [“Customizing Text, Color, and Images” on page 248](#)
- [“Localizing Text for Planning Applications” on page 249](#)
- [“Restricted Characters in Customized Messages” on page 250](#)

Customizing Text, Color, and Images

You can customize text, color, and images for these items in the Planning Web interface:

- Labels and messages, using `HspCustomMsgs_en.template`
- Customizable colors and images, using `HspCustomImgs_en.template`

Whenever you add labels, you must add them to the `HspCustomMsgs` resource file. For example, when you add Smart List entries or menu items, include labels in the resource file. When your application is localized, update the corresponding resource file. For example, to localize an application into three languages, add labels to all three localized `HspCustomMsg` files (each with their language code in the filename) and include the translated words. Note:

- You must avoid certain characters when customizing text or messages. See [“Restricted Characters in Customized Messages” on page 250](#).
- Some colors are named, and others are given as hex or RGB values.
- Image file names and the `Images` directory are case-sensitive on some Web application servers and operating systems.
- You must store images added to the Web interface using `HspCustomImgs_en.properties` in the `Images` directory.
- The silent deployment implemented by WebLogic 8.1 does not extract files from EAR and WAR archives. You must manually extract, modify, and archive the customization files.
- To customize other types of images, see [“Customizing Cascading Style Sheets” on page 243](#).
- The procedure describes customizing English messages in the `HspCustomMsgs_en` file. To localize the application, update the file for the appropriate language.

► To customize text, color, and images:

- 1 **Locate these files:** `HspCustomMsgs_en.template` and `HspCustomImgs_en.template`.

If Planning was installed in the default directory, the files are in this path:

DeploymentDirectory/custom

- 2 **Copy and paste** `HspCustomMsgs_en.template` **and** `HspCustomImgs_en.template` **to a temporary location.**
- 3 **Rename the files** `HspCustomMsgs_en.properties` **and** `HspCustomImgs_en.properties`.
- 4 **Update the content of** `HspCustomMsgs_en.properties` **and** `HspCustomImgs_en.properties`.

Each line in `HspCustomMsgs_en.properties` represents a customizable text resource. Each line in `HspCustomImgs_en.properties` represents a customizable color or image resource.

The lines start with the name of the resource, an equal sign, and the customizable resource. You need not include punctuation at the end of lines. Carriage returns signal the end of the resource.

For example, you can change the Tools menu label from “Tools: Analyze and Report” to “Tools: Additional Resources” by changing this line in the `HspCustomMsgs_en.properties` file:

```
LABEL_HOME_TOOLS=Tools: Analyze and Report
```

to:

```
LABEL_HOME_TOOLS=Tools: Additional Resources
```

- 5 **Save the updated files to** `DeploymentDirectory/custom`.
- 6 **Copy the properties file to a directory that is in the classpath. The recommended directory is the** `WEB-INF\classes` **folder of your application server because this directory is always in the classpath. For example, if you are customizing** `HspCustomMsgs_ru.properties` **on WebLogic, copy the file to this path:**

```
HYPERION_HOME\deployments\WebLogic9\servers\HyperionPlanning\webapps  
\HyperionPlanning\WEB-INF\classes
```

- 7 **Stop and restart the Web application server.**

Localizing Text for Planning Applications

You can localize text for languages supported by Planning applications by customizing labels in the Planning Web interface. This topic provides an example of customizing labels in the `HspCustomMsgs_en.template` file for the Russian language.

► To customize `HspCustomMsgs_en.template`:

- 1 **Locate** `HspCustomMsgs_en.template` **in the** `/custom` **directory.**
- 2 **If you are customizing the labels for a specific language, such as Russian, rename the template file to use a** `.source` **extension, such as** `HspCustomMsgs_ru.source`.
- 3 **Review the source file for the language, such as** `HspCustomMsgs_ru.source`, **to determine which labels in the file need to be customized. Add the translation for the labels to the** `.source` **file, remove all other labels, and save the file. (You need only customize the labels in this file.)**
- 4 **Perform one action:**
 - For languages such as Russian, Japanese, Korean, Turkish, Simplified Chinese, and Traditional Chinese, continue to [step 5](#) to create the Java property file.

- For Latin1 languages, such as Danish, German, Spanish, French, Italian, Brazilian Portuguese, and Swedish, create the property file manually by saving the `.source` file (such as `HspCustomMsgs_ru.source`) with a `.properties` extension, such as `HspCustomMsgs_ru.properties`. Then skip to [step 6](#).

- 5 **Complete this step to use the `native2ascii` program to convert source strings to Unicode strings to create a custom property file for the language, such as `HspCustomMsgs_ru.properties`. To use this program, you must enter the encoding for the language, such as `Cp1251` for Russian. The usage is as follows:**

```
native2ascii [-encoding language_encoding]
[inputfile_name[outputfile_name]]
```

Table 79 Examples of Language Encoding Parameters

Language	Encoding Parameter
Russian	Cp1251
Turkish	Cp1254
Japanese	SJIS
Simplified Chinese	EUC_CN
Traditional Chinese	Big5
Korean	EUC_KR

- a. Ensure that the `native2ascii` program included with the Sun JDK is installed, and note the installation path to this program. Some application servers install this program for you. If it is not installed, you can install the JDK.
- b. Open a command prompt.
- c. Type the path to the `native2ascii` program, followed by the language encoding parameter and the names of the source and target files to be created. For example:

```
C:\jdk1.4.2_15\bin\native2ascii -encoding Cp1251
HspCustomMsgs_ru.source HspCustomMsgs_ru.properties
```

- 6 **Copy the properties file (such as `HspCustomMsgs_ru.properties`) to a directory that is in the classpath. The recommended directory is the `WEB-INF\classes` folder of your application server because this directory is always in the classpath. For example, if you are customizing `HspCustomMsgs_ru.properties` on WebLogic, copy the file to this path:**

```
HYPERION_HOME\deployments\WebLogic9\servers\HyperionPlanning\webapps
\HyperionPlanning\WEB-INF\classes
```

- 7 **Restart the application server.**

Restricted Characters in Customized Messages

You can modify text strings to meet your business needs. Avoid using characters that can cause errors in custom messages, depending on how the text string is used and whether the string is generated by Java or JavaScript. (In most cases, text messages are generated by Java, and pop-

up boxes and some buttons are generated by JavaScript.) For example, errors occur if you add this string inside a JavaScript call because JavaScript cannot parse double quotation marks:

```
someJavaScript ("<%= HspMsgs.SOME_RESOURCE %>");
```

The best practice is to avoid using these characters in custom messages:

- Single quotation marks
- Double quotation marks
- Back slashes
- Forward slashes

Customizing Colors, States, and Actions for Process Status

These procedures require knowledge of how to maintain and manipulate your relational database:

- [“Customizing Colors” on page 251](#)
- [“Customizing States” on page 252](#)
- [“Customizing Actions” on page 252](#)

Customizing Colors

By default, each process status in the Workflow Tasks page is black. You can customize each state to display in another color. Valid colors include all named colors that browsers recognize. See the Microsoft Web site for an list of supported named colors.

The information for process status state colors is stored in the `HSP_PM_STATES` table in the `COLOR` column. You must run statements for your relational database to change the color value in the database.

After you customize the colors, restart the Web application server.

Example: SQL Relational Database

To get a list of all the available process status states and their current color settings, run this query: `select * from hsp_pm_states`

This returns the `state_id`, the name of the process status state, and the color. By default, the value of the color is `<NULL>`, which converts to black.

Note the `state_id` of the process status state you want to change, and run this query:

```
UPDATE HSP_PM_STATES SET COLOR = 'new color' WHERE STATE_ID = state_id
```

To change the First Pass process status state color to green, run this SQL statement:

```
UPDATE HSP_PM_STATES SET COLOR='GREEN' WHERE STATE_ID=1
```

Note: For Oracle relational databases, issue a `COMMIT;` command to commit the transaction.

Customizing States

You can customize these preset process status states:

- Not Started
- First Pass
- Approved
- Not Signed Off
- Under Review
- Signed Off

The process status state information is stored in the `HSP_PM_STATES` table in the `NAME` column. You must run statements for your relational database to change the state value in the database.

Example: SQL Relational Database

Get a list of all the available process status states and their current names by running this query:

```
select * from hsp_pm_states
```

This returns the `state_id`, the name of the process status state, and the color.

Note the `state_id` of the process status state you want to change, and run this query:

```
UPDATE HSP_PM_STATES SET NAME = 'NewName' WHERE STATE_ID = state_id
```

To change the Approved process status state name to Promoted, run this SQL statement:

```
UPDATE HSP_PM_STATES SET NAME='PROMOTED' WHERE STATE_ID=1
```

Note: For Oracle relational databases, issue a `COMMIT;` command to commit the transaction.

Customizing Actions

You can customize these preset process status actions:

- Start
- Exclude
- Reject
- Approve
- Sign Off
- Promote

The process status action information is stored in the `HSP_PM_ACTIONS` table in the `NAME` column. You must run statements for your relational database to change the action value in the database.

Example: SQL Relational Database

To get a list of all the available process status actions and their current names, run this query:

```
select * from hsp_pm_actions
```

This returns the `action_id` and the name of the process status action.

Note the `state_id` of the process status action you want to change, and run this query:

```
UPDATE HSP_PM_ACTIONS SET NAME = 'NewName' WHERE ACTION_ID = action_id
```

To change the Approve process status action name to Accept, run this SQL statement:

```
UPDATE HSP_PM_ACTIONS SET NAME='ACCEPT' WHERE ACTION_ID=1
```

Note: For Oracle relational databases, issue a `COMMIT;` command to commit the transaction.

Creating Spreading Patterns

Administrators who understand SQL server can create custom spreading patterns, accessible from the Grid Spread and Mass Allocate dialog boxes, by adding them to the `HSP_SPREAD_PATTERN` database table.

► To add a custom spreading pattern:

- 1 Open the `HSP_SPREAD_PATTERN` database table.
- 2 Enter row values to name and represent how values are spread from a parent cell:

Table 80 HSP_SPREAD_PATTERN Table

Column	Description
NAME – VARCHAR (80) UNIQUE NOT NULL	Internal name (not displayed)
LABEL – VARCHAR (80) NOT NULL	Reference to a string ID in the resource file, which displays in the user interface. If a string resource is not created, the LABEL string identifier displays instead (for example “Label_Spread_445”).
PATTERN VARCHAR (255) NOT NULL	Space delimited; one or more elements, up to 255 characters: <ul style="list-style-type: none">● A number: For example, to specify that the parent value splits evenly across its children, specify 1.● !: Lock; do not change the value of a cell that has “!” as its pattern element.● P: Parent value being spread. A child receives the exact value the parent receives.● *: Repeat the element that follows. Use 1* to repeat 1 for every cell, to take up the rest of the space in the applied pattern. If no * exists as part of an element within a pattern, the entire pattern repeats itself instead of a specific element. A pattern cannot contain only the * character, and can have only one element with the * character. For example, 1 2* 3 is allowed, but 1 2* 3* is not.

- 3 Save and close the table.

The new patterns display as spreading options on the Grid Spread and Mass Allocate dialog boxes.

See also the *Oracle Hyperion Planning User's Online Help* and “[Examples of Spreading Patterns](#)” on page 254.

Examples of Spreading Patterns

Assume a data form cell having three children cells is affected by a spread. Examples of how patterns are applied to the three child values:

Table 81 Examples of Applied Spreading Patterns

Pattern	Cells			Explanation
	Cell 1	Cell 2	Cell 3	
4 4 5	4	4	5	The new value is spread to level zero members using the 4 4 5 pattern. For example, if you set Q1 to 13, its value is spread as: Jan = 4 Feb = 4 Mar = 5
1	1	1	1	1 is the same as 1*. This FLOW spread type shows how FLOW works if no values exist to make the spread proportional. The parent's value is divided by 3, the number of children cells, and spread equally to each of its three children.
P	P	P	P	Each child cell receives the parent's value. This is more a copy pattern than a spread.
! * P	!	!	P	Equivalent to a BALANCE type of spread.
P ! *	P	!	!	Equivalent to a FIRST type of spread.
1 2 1	1	2	1	Similar to a bell-curve type of spread.

		Cells		
0 1 0	0	1	0	The beginning and ending children cells receive no spread values, and the middle child member receives the parent's value.

In This Chapter

Calculation Script is Too Long or Complex	257
Cannot Find Member	258
Cannot Process Request	258
Conflicting Change by Users	259
Copy Version Error	259
Currency Conversion Calc Script Failed	260
Data Form Error Message	260
Database Locked by Another Administrator	260
Exiting Planning	260
500 Error Message	261
Invalid Value When Entering Date Value.....	262
Manage Database Errors	262
Maximum Number of Applications Already Connected or Database is Locked	263
Registering a Restored Application with Planning	264
Session Timeout.....	265
Slow Performance When Opening Data Forms Using a Dial-Up Connection	265
Slow Performance When Opening Large Data Forms.....	266
Unable to Create Application Error	267
Unable to Refresh Application Error.....	267
Unsuccessful Log On	268

For detailed troubleshooting information, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.

Calculation Script is Too Long or Complex

Scenario:

Use the Manage Currency Conversion page to generate application-wide calc scripts that you can launch from the Web to convert currencies. When generating calc scripts, if this error displays, “Calculation script is too long or complex,” the calc script exceeds the 64K limit. This

can occur if an application has many periods (such as 500) included in a currency conversion calc script.

Solution:

➤ To resolve this issue:

1 Limit the calc script to one scenario.

If the calc script generates successfully, skip to the last step.

2 If the error still displays, limit the calc script to one reporting currency.

If the calc script generates successfully, try selecting two reporting currencies. If that works, continue adding reporting currencies until the error displays. Then skip to the last step. If the calc script does not generate with one reporting currency, remove some currencies from the application.

3 Use the **Manage Currency Conversion page to generate as many other, smaller currency conversion calc scripts as necessary to include your scenarios, reporting currencies, and versions. You can also give periods very short names.**

Cannot Find Member

Scenario:

During database refresh, this error displays: “Cannot Find Member x.”

Solution:

Perform a full database refresh (instead of an incremental refresh) from the Manage Database page.

Cannot Process Request

Scenario:

This error displays to application users: “Cannot process your request because the application is being refreshed.” While application databases are created or refreshed, aspects of the application are locked, and users cannot change:

- Dimensions or members
- Users
- Groups
- Security
- Aliases
- Alias tables

- Exchange rates
- Year
- Period

Solution:

Wait until the database is created or refreshed.

Conflicting Change by Users

Scenario:

When changing data, this error displays: “You are trying to change data that has been changed by a user on another server.”

Solution:

The data is currently being changed by a user on another server. Wait a few seconds and try again. If the message continues to display, try closing the page and reopening it. If the error continues, restart the Web application server.

Copy Version Error

Scenario:

After using Copy Version with a large number of members selected, this error displays: "An error occurred when running specified calc script, please check logs for details." The log for the Web application server contains this message: “com.hyperion.planning.olap.HspOlapException: The length of the calc script exceeds the maximum allowed length.”

Solution:

Copy Version uses a calculation script, and Essbase limits the size of calculation scripts to 64K. If the limit is reached while copying a large number of members, Copy Version can fail. To avoid this, select fewer members to copy with Copy Version. See the *Oracle Hyperion Planning User's Online Help* and *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.

To prevent a failed Copy Version, estimate how many members you can successfully copy. (The script uses member names, so you must consider the number of members and the length of member names. If each member is an average of 10 characters, fewer than 6,400 members can be copied. The script also includes syntax characters, such as the calc script command and a comma for each member name. Assuming a calc script command length of 500 characters, you can use this formula:

(Number of members * Average length of member names) + Number of members + 500 <= 64,000

Currency Conversion Calc Script Failed

Scenario:

When verifying calculation scripts, this error displays if scenarios, versions, or currencies in the calculation script are Dynamic Calc members or all account members are dynamic: “The FIX statement cannot contain a Dynamic Calc member.”

Solution:

When selecting scenarios, versions, and currencies for currency conversion calc scripts on the Manage Database page, do not select Dynamic Calc members. In addition, at least one account must be set to Store.

Data Form Error Message

Scenario:

The first person to use Planning after an Essbase timeout may receive an error that the data form failed to open.

Solution:

The user should click Refresh to restore the connection between Planning and Essbase.

Database Locked by Another Administrator

Scenario:

Occasionally, Planning applications may become locked. This can be caused by events such as abnormally exiting the application.

Solution:

See “[Unlocking Applications](#)” on page 65.

Exiting Planning

Scenario:

In some cases, multiple instances of Planning processes may occur on the Planning server, especially if processes are not completed between logon sessions, or the Web and NT client are

opened simultaneously. This could result in the inability to refresh data and reduce available memory.

Solution:

After all Planning users are logged off, the administrator must manually remove the processes from the Planning server.

- To stop Planning processes manually:
 - 1 Open Task Manager by right-clicking the Windows task bar and selecting Task Manager.
 - 2 Select Processes.
 - 3 Sort by clicking the Image Name column heading.
 - 4 End all instances of these processes:
 - HspDS.exe
 - HSXSER~1.EXE (you may see another number)

500 Error Message

Scenario:

You may receive a 500 ERROR message on a data form when the number of members on the page (the cross-product combination of dimensions after security filter is applied) causes Java to run out of memory.

Solution:

Oracle recommends reducing the number of cross-product dimension combinations on the page as much as possible. By default, Java is allocated 128 MB of memory, but if your server has more memory available, you should increase the amount that Java can use. As a general rule, if the server is dedicated to Planning, allocate 75% of the server's RAM to Java; if the server is not dedicated, allocate 25 to 50% of the RAM to Java. The minimum memory setting should be 128 MB.

- To change the Java memory settings for WebSphere:
 - 1 Highlight the default server in the administrator console.
 - 2 Enter the amount of RAM in the Initial and Maximum java heap size fields.

Note: You can set both values to 512 or 1024, depending on your system's RAM. If it has 1 GB of physical RAM, use 512 MB as the java heap size.

- To change the Java memory settings for WebLogic:
 - 1 Open the CMD file used to start the application server (for example, `startHPServer.cmd`).
 - 2 Locate the section for the `%JAVA_HOME%` variables and make the changes shown below.

```
echo on

"%JAVA_HOME%\bin\java" -hotspot -ms256m -mx256m -classpath %CLASSPATH% -
Dweblogic.management.discover=false -Dweblogic.Domain=Hyperion -
Dweblogic.Name=HPServer "-Dbea.home=%BEA_HOME%" -
Dweblogic.management.password=%WLS_PW% -
Dweblogic.ProductionModeEnabled=%STARTMODE% "-Djava.security.policy==
%WEBLOGIC_HOME%/lib/weblogic.policy" weblogic.Server

goto finish
```

Invalid Value When Entering Date Value

Scenario:

When trying to enter a value into a data form cell, you get the error, “You have entered an invalid value. Please try again.” For example, you try to enter a formatted date value (for example, 12/8/2008) into a cell whose Data Type is Date and get this error.

Solution:

Depending on the type of data users want to enter in a cell, in Oracle Hyperion EPM Architect, Fusion Edition, set the dimension of the member with this data type as the first in the Data Type Evaluation Order.

Manage Database Errors

Scenario:

Manage database errors can occur when the 8.3 naming convention is not used. When the application database is created or refreshed, the error may display as: “Manage Database Errors detected (Start 1060044): Could not open outline -2147218904.”

Solution:

To resolve Manage Database errors, you may need to redirect the operating system environment variables. Essbase requires an 8.3 character-naming convention for the Temp and Tmp environment variables in the operating system. Use this procedure to check the naming convention and reassign these variables if needed.

File Naming Convention

Ensure that you are logged on to the Planning server before making changes.

- To check the environment variables for Windows 2000:
 - 1 Close Planning.
 - 2 Close Essbase.

- 3 On the Windows desktop, right-click **My Computer**.
- 4 Select **Properties**.
- 5 Select **Advanced**.
- 6 Select **Environment Variables**.
- 7 Change the settings for Temp and Tmp in the user variable to the 8.3 naming convention.
For example, C:\Temp
- 8 Open, and open the outline.
- 9 Close, and restart Planning.
- 10 Try creating or refreshing the database on the Manage Database page. If the error recurs, repeat the previous steps, but this time also create a folder called C:\tmp and set the System Environment variables Temp and Tmp to C:\tmp.

If the error persists, create a Temp folder under C:\temp. Repeat the previous steps, redirecting first the user variables then the system variables to C:\temp\temp. If the error continues, redirect the environment variables to C:\temp.

Note: The settings for Temp and Tmp in the user and system variables must be unique. Do not set the user and system variables to the same folder.

Maximum Number of Applications Already Connected or Database is Locked

Scenario:

You are using DB2 as your relational database, and you receive either of these messages when you try to create or refresh the database:

- ... maximum number of applications already connected
- Database is locked ...

By default, the MAXAPPLS parameter is set to 40.

Solution:

Increase the number of applications allowed by increasing the MAXAPPLS parameter.

- To increase the MAXAPPLS parameter:

- 1 In **Control Center**, right-click the database and select **Configure Parameters**.

Alternately, you can set the parameter from the DB2 window.

- 2 Set the MAXAPPLS parameter, using this format:

```
db2 -v update db cfg for database name using MAXAPPLS n
```

```
db2 -v terminate
```

where *database name* is the name of the relational database and *n* is the number of applications that can be connected simultaneously. For example, to increase the number of maximum number of applications that can be connected simultaneously to 100 for a database named Business, specify:

```
db2 -v update db cfg for Business using MAXAPPLS 100
```

```
db2 -v terminate
```

Registering a Restored Application with Planning

Scenario:

You have restored a Planning application, but it is not available for selecting in EPM Workspace.

Solution:

Register the application with Shared Services from within Planning.

► To register a restored Planning application with Shared Services from Planning:

1 Enter the Planning URL.

For example:

```
http://hostname:8300/HyperionPlanning/LogOn.jsp
```

2 Select Administration, then Application Settings.

3 From Show, select Advanced Settings, then click Go.

4 Select System Settings, and under Shared Services Setting, enter the URL to the Shared Services server in this format:

```
http://hostname:portnumber/interop
```

The *hostname* can include a domain, but the domain is not required if the server can resolve the hostname to the Shared Services server using other means. For example:

```
http://product:58080
```

or

```
http://finserv.hyperion.com:58080/interop
```

Note: Even if Shared Services and Planning run on the same server, do not register the Shared Services URL as *localhost:port number*. Instead, use the DNS-listed hostname for that server, such as *sharedserver:58080*.

5 Click Register Shared Services.

Session Timeout

Scenario:

If a user quits a session by closing the browser instead of logging off, the session times out after a period of time, and the user name is removed from the Statistics page. You can change the number of minutes before the session times out by modifying timeout settings for your Web application server. For example, for Apache Tomcat 5.0.28, modify the `web.xml` file in the `Web-inf` directory.

Solution:

- To change the session timeout setting:

- 1 **Open the timeout setting file for your Web application server.**

For example, in Apache Tomcat 5.0.28, open the `web.xml` file in the `driveletter:\Tomcat\conf` directory.

- 2 **Modify the timeout setting for the Web application server, and save your changes.**

For example, for Apache Tomcat 5.0.28, change the number 60 in the `session-timeout` setting to the number of minutes to use.

To modify timeout settings, see the documentation for your Web application server.

Slow Performance When Opening Data Forms Using a Dial-Up Connection

Scenario:

Opening a data form using a slow network connection (for example, with a modem) is slow.

Solution:

You can increase the network bandwidth when opening data forms by modifying the `web.xml` file. This compresses by approximately 90% the data stream sent from the Planning server to the client.

Note: If you use a WebLogic (all supported versions) Web application server, complete the second procedure, specific to WebLogic. If you use another Web application server, complete the first procedure.

- To improve performance for a Web application server other than WebLogic:

- 1 **With a text editor, open the `web.xml` file in `HyperionPlanning.ear` or `HyperionPlanning.war`.**

- 2 **After the tag `</description>` and before the tag `<listener>`, insert these lines:**

```

<filter>
    <filter-name>HspCompressionFilter</filter-name> <filter-
class>com.hyperion.planning.HspCompressionFilter</filter-class>
    <init-param>
        <param-name>compressionThreshold</param-name>
        <param-value>2048</param-value>
    </init-param>
    <init-param>
        <param-name>debug</param-name>        <param-value>1</param-value>
    </init-param>
</filter>
<filter-mapping>
    <filter-name>HspCompressionFilter</filter-name>
    <url-pattern>/EnterData.jsp</url-pattern>
</filter-mapping>

```

3 Save the web.xml file.

If you use WebLogic, you must manually modify the .ear file and redeploy it for the Web application server.

► To improve performance with a WebLogic application server:

- 1 Unzip the HyperionPlanning.ear file to /ear, for example.**
- 2 Unzip Hyperion.war under /ear to /war.**
- 3 With a text editor, open /war/WEB-INF/web.xml and modify it using the instructions in step 2 in the preceding procedure.**
- 4 Compress the content in /war to /ear/HyperionPlanning.war.**
- 5 Compress the content in /ear into /ear/HyperionPlanning.ear.**
- 6 Deploy the new HyperionPlanning.ear for the WebLogic Web application server.**

Slow Performance When Opening Large Data Forms

Scenario:

Unusually large data forms with many cells may require significant time to open.

Solution:

You can display a warning when users try to open data forms that are larger than a specified number of cells. The default value for the number of cells that trigger the warning is 5,000. Administrators can change the default value

► To set the number of data form cells that trigger a warning message upon opening:

- 1 In **Planning**, select **File**, then **Preferences**.
- 2 Select **Display Options**.
- 3 In **Warn If Data Form Larger Than Cells Specified**, enter a value.

See also [“Clearing Options when Creating or Refreshing an Application Database”](#) on page 74.

Unable to Create Application Error

Scenario:

When you try to create an application in Planning, this error might display: “Unable to create application *application name*. Error number: -2147160060.”

Solution:

There may be several causes. To resolve this, ensure that these conditions are met:

- Essbase must be running.
- Advanced User Rights must be configured on the local machine.
- The Administrator user must be configured for HsxServer and HspDataSource.
- If the Essbase supervisor user name matches the Planning administrator user name, the passwords must be identical.
- If you use local authentication, the machine name must be different from the user name.
- The Planning application name cannot match Essbase application names.
- If you are using DB2, a error message might state that the database is not configured correctly. This can result from various causes, such as incorrect configuration for auditing tables.

Unable to Refresh Application Error

Scenario:

When using a DB2 database and refreshing an application in Planning, an error displays.

Solution:

DB2 could not refresh or correctly generate its transaction log because insufficient space was allocated for log files. (See the *Oracle Hyperion Enterprise Performance Management System*

Installation and Configuration Troubleshooting Guide.) To resolve this issue, use these procedures.

- Allocate more space for the DB2 transaction log:
 - 1 In DB2 Command Center, select **Tools**, then **Wizard**, and then **Configure Database Logging Wizard**.
 - 2 Connect to the database and ensure that **Circular Logging** is selected.
 - 3 On the next screen, increase the number of primary (for example, 20) and secondary log files (for example, 10).
 - 4 Increase the size of each log file (for example, 2000).
 - 5 Click **Next** to accept all defaults.
 - 6 At **Summary**, enter the user name and password.
 - 7 Click **Finish**.
 - 8 Restart DB2.
 - 9 Refresh the database.

- Perform a database refresh more frequently, after a few changes instead of waiting for all changes to the application.

Unsuccessful Log On

Scenario:

When logging on to Planning Web client, this error displays: “Unsuccessful login. Please try again.”

Solutions:

Ensure that you are using a valid username and password.

Ensure that Essbase server and Oracle's Hyperion® Shared Services are running.

Review the error log. (See *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.)

If the Log On button does not appear at logon, adjust the security settings in your Web browser, and add the Planning server name as a Trusted Site. For example, in Internet Explorer, select Tools, then Internet Options, then Security, then Trusted Sites, and then Sites and add `http://servername` to the trusted sites zone.



Customizing Data Forms with JavaScript

In This Appendix

About Customizing Data Forms	269
Modifying JavaScript	269
Deploying Custom JavaScript	274

About Customizing Data Forms

This section describes using JavaScript to customize data forms. If you are a programmer familiar with JavaScript, you can customize the look and behavior of Planning data forms by writing custom JavaScript and including it in applications.

You can customize data forms in these ways:

- Add buttons to data forms that link to Web pages or run custom JavaScript code.
- Modify application behavior when users save data forms.
- Modify application behavior when the focus enters or leaves a cell.

There are no additional requirements for users when including custom JavaScript in your Planning application. Administrators need not take special steps to enable custom JavaScript for applications.

Modifying JavaScript

To add custom JavaScript to a Planning application, modify the code in `ValidateData.js`, installed in `/default web application directory/Planning/web/WEB-INF`. For examples of customizing data forms, see the sample code and comments in `SampleValidateData.js`.

`customCellEnterPre`

Description

Use `customCellEnterPre` to change the behavior when users click in cells, before default Planning logic runs. For example, when users click in cells, a message can indicate the acceptable

range of values to enter. When users click in certain types of cells, such as accounts, a message can prompt users to enter supporting detail.

After `customCellEnterPre` runs, if it returns `True`, the default Planning logic runs. If it returns `False`, no additional logic follows the custom code. `SampleValidateData.js` includes an example that displays a message prompting users to enter supporting detail when clicking in certain cells.

Arguments

Argument Description

<i>Row</i>	An integer that defines the row for the cell.
<i>Col</i>	An integer that defines the column for the cell.
<i>Cell</i>	A handle for the HTML input artifact.

Returns

Return Value Description

<i>True</i>	The Planning default logic runs after this code completes.
<i>False</i>	No further logic (Planning default logic or <code>customCellEnterPost</code>) runs after this completes.

See Also

[customCellEnterPost](#)

customCellEnterPost

Description

Use `customCellEnterPost` for custom behavior when focus comes into a cell, after the default Planning logic runs. This code runs when users click in a cell, after the Planning default logic if `customCellEnterPre` and the default logic return `True`. If they return `False`, this function is disabled.

`SampleValidateData.js` includes an example for `customCellEnterPre` that you can use for `customCellEnterPost`. The example displays a message prompting the user to enter supporting detail clicking in specific cells.

Arguments

Argument Description

<i>Row</i>	An integer that defines the row for the cell.
<i>Col</i>	An integer that defines the column for the cell.

Argument Description

Cell A handle for the HTML input artifact.

Returns**Return Value Description**

True The Planning default logic runs after this code completes.

False No further logic runs after this completes.

See Also

[customCellEnterPre](#)

customCellValidatePre

Description

Use `customCellValidatePre` for custom behavior when focus leaves a cell, before default Planning logic runs. For example, you can compare the value a user entered for the cell to another value and display an alert if the variance is too great or out of range. You can define a spreading algorithm that occurs when the focus leaves a cell, replacing the default spreading algorithm. For example, Planning usually spreads by time period, but you can write custom JavaScript that pre-aggregates the data on the client side before writing values back to Essbase.

After this function runs, if `customCellValidatePre` returns `True`, Planning default logic runs. If it returns `False`, no additional logic follows the custom code. Data forms use your custom code and skip the default Planning logic, which could cause users to submit invalid data.

`SampleValidateData.js` includes an example that executes additional validation when the focus leaves a cell. When the focus leaves a cell, the code compares the value the user entered for the current year to last year's value. If the current year value is over 10% larger than the previous year, a message is displayed.

Arguments**Argument Description**

Row An integer that defines the row for the cell.

Col An integer that defines the column for the cell.

Cell A handle for the HTML input artifact.

Returns

Argument	Description
----------	-------------

<i>True</i>	The Planning default logic runs after this code completes.
-------------	--

<i>False</i>	No further logic runs after this code completes.
--------------	--

See Also

[customCellValidatePost](#)

customCellValidatePost

Description

Use `customCellValidatePost` for custom behavior when focus leaves a cell, after the default Planning logic runs. This is similar to `customCellValidatePre`, which runs when the focus leaves a cell, after the Planning default logic if this function and the Planning default logic return `True`. If they return `False`, this function is disabled. Return values have no effect.

`SampleValidateData.js` includes an example for `customCellValidatePre` that you can use for this code. The example executes additional validation when the focus leaves a cell. The JavaScript code compares the value entered for the current year to the value for last year. If the current year value is over 10% larger than the previous year value, a message is displayed.

Arguments

Argument	Description
----------	-------------

<i>Row</i>	An integer that defines the row for the cell.
------------	---

<i>Col</i>	An integer that defines the column for the cell.
------------	--

<i>Cell</i>	A handle for the HTML input artifact.
-------------	---------------------------------------

Returns

Return Value	Description
--------------	-------------

<i>True</i>	The Planning default logic runs after this code completes.
-------------	--

<i>False</i>	No further logic runs after this completes.
--------------	---

See Also

[customCellValidatePre](#)

customOnLoad

Description

Use `customOnLoad` for custom behavior when data forms are loaded or reloaded. A JavaScript variable called `savedData` indicates whether Save has been completed. For example, when users open data forms, you could display instructions or determine if they match tolerances and inform users of any corrective actions.

Arguments

No arguments.

Returns

No return values.

drawCustomButtons

Description

Use `drawCustomButton` to add custom buttons to data forms. Buttons can link to any data entry page or launch custom JavaScript code included in `ValidateData.js`. For example, you can add buttons to validate data forms or run reports.

`SampleValidateData.js` includes an example that adds a button labeled **Validate** to a data form. When a user clicks the button, the JavaScript code compares the values entered for the current year to last year's values. If the current year's values are more than 10% larger than the previous year's values, a message is displayed.

Arguments

No arguments.

Returns

No return values.

validateForm

Description

Use `validateForm` to provide data form-level behavior that is launched when users click Save. With default Planning behavior, when users click Save on data forms, the `validateForm` function submits the grid. For example, you can calculate a variance between budget and actuals by comparing values in two columns, displaying an alert when users click Save if the variance is too high.

`SampleValidateData.js` includes an example that executes additional validation when users click Save. The values entered for the current year are compared to last year's values. If the current year values are more than 10% larger than previous year values, a message displays.

Arguments

No arguments.

Returns

Return Value	Description
--------------	-------------

<i>True</i>	Saves the grid.
-------------	-----------------

<i>False</i>	Cancels the save.
--------------	-------------------

Deploying Custom JavaScript

When using custom JavaScript in Planning applications:

- Specify which data forms the code applies to, as shown in `SampleValidateData.js`.
- Maintain custom JavaScripts when upgrading Planning. When upgrading modifying standard Planning JavaScript files, you must merge your code with the updated version of `ValidateData.js`. (If there are no changes to `ValidateData.js`, you can back up the file before upgrading Planning and copy your version of the file.)
- The default Planning calc scripts can run after data forms are saved, so default calc scripts could overwrite the actions of custom JavaScript. If custom calc scripts run automatically after data forms are saved, calc scripts could overwrite actions of custom JavaScripts. If custom JavaScript determines how values are propagated throughout the hierarchy, be aware of possible conflicts with calc scripts.
- Custom JavaScript can modify cells that are not visible on data forms. For example, if a quarter is collapsed, custom JavaScript can still affect values for individual months.
- Custom JavaScript cannot affect suppressed rows or columns.
- Depending on the complexity of code, cell-level JavaScript functions can have an adverse effect on application performance. For complex JavaScript, consider using the data form-level function (`validateForm`) or custom buttons to launch JavaScript code.
- Utility functions are included in `validateDataHelper.js`.
- If custom JavaScript overrides default Planning behavior, you are responsible for ensuring that data is valid. You can verify data after running custom JavaScript using `enterData.js`. For examples, see `LeaveCell`.



Naming Restrictions for Essbase

In This Appendix

Naming Restrictions for Applications.....	275
Naming Restrictions for Dimensions, Members, Aliases, and Data Forms	276
Dimension and Member Names in Calc Scripts, Report Scripts, Formulas, Filters, and Substitution Variables	277
Naming Restrictions for User Names.....	278

Naming Restrictions for Applications

When naming applications, follow these rules:

- For non-Unicode mode applications, use no more than 8 bytes; for Unicode-mode applications, use no more than 30 characters.
- Do not use spaces.
- Do not use these special characters:
 - asterisks
 - backslashes
 - brackets
 - colons
 - commas
 - equal signs
 - greater than signs
 - less than signs
 - periods
 - plus signs
 - question marks
 - quotation marks (double and single)
 - semicolons
 - slashes
 - tabs
 - vertical bars

- For Planning application names in relational database environments, do not use extended characters (except for underscores).
- For aggregate storage databases, do not use DEFAULT, LOG, METADATA, or TEMP as application names.

Enter names in the preferred case. Application names are created exactly as entered.

For detailed information on creating Essbase applications and databases, see the Essbase product documentation.

Naming Restrictions for Dimensions, Members, Aliases, and Data Forms

When naming dimensions, members, and aliases, follow these rules:

- For non-Unicode mode dimensions, members, or aliases, use no more than 80 bytes. For Unicode-mode dimensions, members, or aliases, use no more than 80 characters.
- Distinguish between upper and lower case only if case sensitivity is enabled.
- Do not use HTML tags in member names, dimension names, aliases, and descriptions.
- Do not use quotation marks, brackets, backslashes, or tabs. Brackets are permitted but not recommended in block storage outlines. They cause errors when converting to aggregate storage outlines.
- To begin dimension or member names, do not use these characters:
 - at signs
 - backslashes
 - brackets
 - commas
 - dashes, hyphens, or minus signs
 - equal signs
 - less than signs
 - parentheses
 - periods
 - plus signs
 - quotation marks
 - underscores
 - vertical bars
- Do not place spaces at the beginning or end of names. Essbase ignores such spaces.
- Do not use forward slashes in member names.
- For time periods in custom calendars, do not use spaces in prefixes.

- Do not use these words as dimension or member names:
 - Calculation script commands, operators, and keywords; for a list of commands, see the Essbase product documentation.
 - Report writer commands.
 - Function names and function arguments.
 - Names of other dimensions and members (unless the member is shared), and generation names, level names, and aliases in the database.
 - These words:

ALL AND ASSIGN AVERAGE CALC	GENRANGE GROUP GT ID IDERROR	RELOP SET SKIPBOTH
CALCMBR COPYFORWARD	INTEGER LE LEVELRANGE	SKIPMISSING SKIPNONE
CROSSDIM CURMBRNAME DIM	LOOPBLOCK LOOPPARMS LT MBR	SKIPZERO TO LOCALRATE
DIMNAME DIV DYNAMIC	MBRNAME MBRONLY MINUS	TRAILMISSING TRAILSUM UMINUS
EMPTYPARM EQ EQOP EXCEPT EXP	MISSING MUL MULOP NE NON	UPPER VARORXMBR XMBRONLY \$
EXPERROR FLOAT FUNCTION GE	NONINPUT NOT OR PAREN	\$UNIVERSE\$\$ #MISSING #MI
GEN	PARENPARM PERCENT PLUS	

- If Dynamic Time Series is enabled, do not use History, Year, Season, Period, Quarter, Month, Week, or Day.

Dimension and Member Names in Calc Scripts, Report Scripts, Formulas, Filters, and Substitution Variables

In substitution variable values, calc scripts, report scripts, filter definitions, partition definitions, or formulas, you must enclose member names in brackets ([]) when used within MDX statements and in quotation marks (") for block storage databases, in these situations:

- The name starts with one or more numerals (for example, 100).
- The name contains spaces or these characters:

&	ampersand	>	greater than sign
*	asterisk	<	less than sign
@	at sign	()	parentheses
\	backslash	%	percent sign
{ }	brackets	.	period
:	colon	+	plus sign
,	comma	;	semicolon
-	dash, hyphen, or minus	/	slash
!	exclamation point	~	tilde
=	equal sign		

In calculation scripts and formulas, enclose member names that are also 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 quotes names that contain these characters in calc scripts, report scripts or formulas, and names that start with these characters: Spaces + - * / () : , @ ;) { } [] <

Naming Restrictions for User Names

User names in Essbase must be under 30 characters.



Sample Application

In This Appendix

About the Application	279
Predefined Structure	280
Creating Sample Applications	283
Loading Data	284

About the Application

The Sample Application is a Planning application, included with installation, that provides sample application designs and objects: dimensions, composite data forms, business rules, menus, task lists, user variables, substitution variables, and data. It provides a basic application framework that can be used to ensure that a required software system is in place before you create a fully functional Planning application. The Sample Application does not contain all of the functionality available in the current release, and should not be construed as a typical Planning application. You can work with the Sample Application in Classic Planning.

The Sample Application:

- Confirms completeness of installation and configuration when you initialize this application.
- Demonstrates the relationships among application elements.
- Provides sample data forms whose definitions you can export for use in your own applications.

For information on:

- Installing Planning, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.
- Managing Planning, see this guide.
- Creating and initializing the Sample Application, see [“Creating Sample Applications” on page 283](#).
- Loading data into the Sample Application, see [“Loading Data” on page 284](#).

The Sample Application models a company specializing in producing, selling, and installing audio systems, digital video, televisions, and VCRs.

When you create the Sample Application, a Planning database is created.

Predefined Structure

This section describes the predefined elements.

Dimension Members

Table 82 Predefined Dimension Members

Dimension	Members
Period	19
Account	314
Entity	230
Scenario	4
Version	5
Currency	17 (4 reporting currencies)
Year	4
Segments	35

Note: To convert values from one currency to another, set up exchange rates.

Data Forms

Some of the predefined data forms contain data. You can manipulate the loaded data and enter new data to get familiar with Planning functionality.

Financials folder:

- Balance Sheet
- Cash Flow
- Financials Summary
- Income Statement
- Revenue Impact

Revenue folder:

- Plan Revenue – All Years
- Plan Revenue – Assumptions
- Plan Revenue – COS
- Plan Revenue – Detail by Year

Expenses folder:

- Plan Dept Expenses
- Plan Facilities Expenses
- Plan Operating Expenses
- Plan G&A Expenses (hidden data form)

Allocation folder:

- Allocations – Expense
- Allocations – IT
- Allocations – Marketing
- Allocations – Facilities Expense
- Allocations – Segment

Business Rules

The Sample Application includes these business rules:

- AggAll
- AggPlan
- CalcRev
- ClrBS
- ClrFinal
- ClrTrgts

In your own application, you typically associate business rules with the appropriate data forms. Because the Sample Application business rules are not yet associated with the data forms, launch them by selecting Tools, then Business Rules. Select the business rule and click Launch.

User Variables

With user variables enabled, rows, columns, or pages in data forms are dynamically filtered, based on how each person has set user variables as a preference. The Sample Application's data forms provide several examples of how dynamic user variables can be used.

➤ To enable dynamic user variables:

- 1 Select **Administration**, then **Manage Data Forms**.
- 2 Select a data form, then click **Edit**.
- 3 Select the **Page/Point of View Layout** tab, then select the **Enable Dynamic User Variables** check box.

When you log on to the Sample Application as an administrator, these users variables and members are set:

Table 83 User Variables

Dimension	User Variable Name	Selected Member
Scenario	Compare Scenario	Plan
Scenario	Current Scenario	Forecast
Account	Allocation Expense	312100
Account	Revenue Measure	400000
Entity	My Region	E01_101_1110
Period	Allocation Quarter	Q4
Version	Compare Version	Final
Version	Current Version	Working
Segments	My Segment	Seg01

After you add planners to an application, users must set their own user variables by selecting File, then Preferences, and then User Variable Options, and selecting members.

Substitution Variables

Substitution variables act as global placeholders for information that changes regularly. Each variable has a value assigned to it that can be set and changed centrally on the Essbase server. Substitution variables are especially useful when you are developing and reporting on rolling forecasts. When you select substitution variables as members on the data form, the values for those members are based on information that is dynamically generated. For example, you could set the current Period member to the substitution variable CurrPrd so that when the month changes, you need not update the month value manually in the data form or the report script.

Predefined substitution variables:

- PriorYr = FY07
- CurrYr = FY08
- NextYr = FY09
- NextYr1 = FY10
- PriorPrd = Jan
- CurrPrd = Feb
- NxtPrd = Mar

Menus

Menus enable you to right-click in data forms and launch another data form. The members selected for the member, page, and POV are retained in the launched data form. If it contains these members on the page, its page is set to match the context.

Table 84 Predefined Menu Items for Plan Revenue Menu

Menu	Associated Data Form
Revenue Assumptions	Accounts in <i>Plan Revenue – Assumptions</i> data form
Cost of Sales Rate	Accounts in <i>Plan Revenue – COS</i> data form

Task Lists

Task lists guide users through the planning process by listing tasks, instructions, and due dates. From a task list, planners can launch data forms, launch business rules, or promote planning units.

Database

The Sample Application's plan type, or database, is named *Consol*. It consists of expenses for departments and sales, and data for financial statements and capital-related expenses.

Users

The Sample Application includes only one user: the administrator who creates and initializes the Sample Application.

Creating Sample Applications

In order to initialize the sample application, you must create an application using Classic Application Administration with the Sample Application checkbox selected. See [“Selecting Data Sources” on page 181](#). You cannot initialize the sample application in an existing application.

After creating your Classic Sample Application, you must initialize it.

► To initialize a Classic Sample Application:

- 1 Start Oracle Enterprise Performance Management Workspace, Fusion Edition and log on to Planning.
- 2 In Oracle Hyperion Planning, Fusion Edition, select **Edit, Sample Application**, then **Initialize Sample Application**.

A message confirms when initialization is successful; the Initialize item no longer displays.

- 3 Select **Administration**, then **Manage Database** and run **Cube Create** to create the outline in Oracle Essbase.

You can now load data into the Sample Application. See [“Loading Data” on page 284](#).

Loading Data

You load Sample Application data directly through Administration Services console.

► To load data into the Sample Application:

1 In `HYPERION_HOME/products/Planning/bin/sampleapp`, find the `sampApp_data.zip` file.

This file contains the `SampleApp_data.txt` data file.

2 Extract `sampApp_data.txt` to any location on your drive.

3 Start Oracle Essbase Administration Services console and log on.

4 In the left panel, expand to: .

`Essbase Servers/Server Name/Applications/Sample_app_name/Consol`

5 Right-click and select **Load data**.

6 Select **Find Data File**.

7 Select the **File System** tab.

8 Browse to the file where you extracted the sample data, and click **OK**.

9 Click **OK**, then click **OK** again to begin loading the data.

A success message displays when the data is loaded.

Glossary

! See *bang character (!)*.

#MISSING See *missing data (#MISSING)*.

access permissions A set of operations that a user can perform on a resource.

accessor Input and output data specifications for data mining algorithms.

account blocking The process by which accounts accept input data in the consolidated file. Blocked accounts do not receive their value through the additive consolidation process.

account eliminations Accounts which have their values set to zero in the consolidated file during consolidation.

account type How an account's value flows over time, and its sign behavior. Account type options can include expense, income, asset, liability, and equity.

accountability map A visual, hierarchical representation of the responsibility, reporting, and dependency structure of the accountability teams (also known as critical business areas) in an organization.

accounts dimension A dimension type that makes accounting intelligence available. Only one dimension can be defined as Accounts.

active service A service whose Run Type is set to Start rather than Hold.

activity-level authorization Defines user access to applications and the types of activities they can perform on applications, independent of the data that will be operated on.

ad hoc report An online analytical query created on-the-fly by an end user.

adapter Software that enables a program to integrate with data and metadata from target and source systems.

adaptive states Interactive Reporting Web Client level of permission.

adjustment See *journal entry (JE)*.

Advanced Relational Access The integration of a relational database with an Essbase multidimensional database so that all data remains in the relational database and is mapped to summary-level data residing in the Essbase database.

agent An Essbase server process that starts and stops applications and databases, manages connections from users, and handles user-access security. The agent is referred to as ESSBASE.EXE.

aggregate cell A cell comprising several cells. For example, a data cell that uses Children(Year) expands to four cells containing Quarter 1, Quarter 2, Quarter 3, and Quarter 4 data.

aggregate function A type of function, such as sum or calculation of an average, that summarizes or performs analysis on data.

aggregate limit A limit placed on an aggregated request line item or aggregated metatopic item.

aggregate storage database The database storage model designed to support large-scale, sparsely distributed data which is categorized into many, potentially large dimensions. Upper level members and formulas are dynamically calculated, and selected data values are aggregated and stored, typically with improvements in overall aggregation time.

aggregate view A collection of aggregate cells based on the levels of the members within each dimension. To reduce calculation time, values are pre-aggregated and stored as aggregate views. Retrievals start from aggregate view totals and add up from there.

aggregation The process of rolling up and storing values in an aggregate storage database; the stored result of the aggregation process.

aggregation script In aggregate storage databases only, a file that defines a selection of aggregate views to be built into an aggregation.

alias An alternative name. For example, for a more easily identifiable column descriptor you can display the alias instead of the member name.

alias table A table that contains alternate names for members.

alternate hierarchy A hierarchy of shared members. An alternate hierarchy is based upon an existing hierarchy in a database outline, but has alternate levels in the dimension. An alternate hierarchy allows the same data to be seen from different points of view.

ancestor A branch member that has members below it. For example, the members Qtr2 and 2006 are ancestors of the member April.

appender A Log4j term for destination.

application (1) A software program designed to run a specific task or group of tasks such as a spreadsheet program or database management system. (2) A related set of dimensions and dimension members that are used to meet a specific set of analytical and/or reporting requirements.

application currency The default reporting currency for the application.

area A predefined set of members and values that makes up a partition.

arithmetic data load A data load that performs operations on values in the database, such as adding 10 to each value.

artifact An individual application or repository item; for example, scripts, forms, rules files, Interactive Reporting documents, and financial reports. Also known as an object.

assemblies Installation files for EPM System products or components.

asset account An account type that stores values that represent a company's assets.

assignment The association of a source and destination in the allocation model that controls the direction of allocated costs or revenue flow within Profitability and Cost Management.

attribute Characteristic of a dimension member. For example, Employee dimension members may have attributes of Name, Age, or Address. Product dimension members can have several attributes, such as a size and flavor.

attribute association A relationship in a database outline whereby a member in an attribute dimension describes a characteristic of a member of its base dimension. For example, if product 100-10 has a grape flavor, the product 100-10 has the Flavor attribute association of grape. Thus, the 100-10 member of the Product dimension is associated with the Grape member of the Flavor attribute dimension.

Attribute Calculations dimension A system-defined dimension that performs these calculation operations on groups of members: Sum, Count, Avg, Min, and Max. This dimension is calculated dynamically and is not visible in the database outline. For example, using the Avg member, you can calculate the average sales value for Red products in New York in January.

attribute dimension A type of dimension that enables analysis based on the attributes or qualities of dimension members.

attribute reporting A reporting process based on the attributes of the base dimension members. *See also* [base dimension](#).

attribute type A text, numeric, Boolean, date, or linked-attribute type that enables different functions for grouping, selecting, or calculating data. For example, because the Ounces attribute dimension has the type numeric, the number of ounces specified as the attribute of each product can be used to calculate the profit per ounce for that product.

authentication Verification of identity as a security measure. Authentication is typically based on a user name and password. Passwords and digital signatures are forms of authentication.

authentication service A core service that manages one authentication system.

auto-reversing journal A journal for entering adjustments that you want to reverse in the next period.

automated stage A stage that does not require human intervention, for example, a data load.

axis (1) A straight line that passes through a graphic used for measurement and categorization. (2) A report aspect used to arrange and relate multidimensional data, such as filters, pages, rows, and columns. For example, for a data query in Simple Basic, an axis can define columns for values for Qtr1, Qtr2, Qtr3, and Qtr4. Row data would be retrieved with totals in the following hierarchy: Market, Product.

backup A duplicate copy of an application instance.

balance account An account type that stores unsigned values that relate to a particular point in time.

balanced journal A journal in which the total debits equal the total credits.

bang character (!) A character that terminates a series of report commands and requests information from the database. A report script must be terminated with a bang character; several bang characters can be used within a report script.

bar chart A chart that can consist of one to 50 data sets, with any number of values assigned to each data set. Data sets are displayed as groups of corresponding bars, stacked bars, or individual bars in separate rows.

base currency The currency in which daily business transactions are performed.

base dimension A standard dimension that is associated with one or more attribute dimensions. For example, assuming products have flavors, the Product dimension is the base dimension for the Flavors attribute dimension.

base entity An entity at the bottom of the organization structure that does not own other entities.

batch calculation Any calculation on a database that is done in batch; for example, a calculation script or a full database calculation. Dynamic calculations are not considered to be batch calculations.

batch file An operating system file that can call multiple ESSCMD scripts and run multiple sessions of ESSCMD. On Windows-based systems, batch files have BAT file extensions. On UNIX, batch files are written as a shell script.

batch loader An FDM component that enables the processing of multiple files.

batch POV A collection of all dimensions on the user POV of every report and book in the batch. While scheduling the batch, you can set the members selected on the batch POV.

batch processing mode A method of using ESSCMD to write a batch or script file that can be used to automate routine server maintenance and diagnostic tasks. ESSCMD script files can execute multiple commands and can be run from the operating system command line or from within operating system batch files. Batch files can be used to call multiple ESSCMD scripts or run multiple instances of ESSCMD.

block The primary storage unit which is a multidimensional array representing the cells of all dense dimensions.

block storage database The Essbase database storage model categorizing and storing data based on the sparsity of data values defined in sparse dimensions. Data values are stored in blocks, which exist only for sparse dimension members for which there are values.

Blocked Account An account that you do not want calculated in the consolidated file because you want to enter it manually.

book A container that holds a group of similar Financial Reporting documents. Books may specify dimension sections or dimension changes.

book POV The dimension members for which a book is run.

bookmark A link to a reporting document or a Web site, displayed on a personal page of a user. The two types of bookmarks are My Bookmarks and image bookmarks.

bounding rectangle The required perimeter that encapsulates the Interactive Reporting document content when embedding Interactive Reporting document sections in a personal page, specified in pixels for height and width or row per page.

broadcast message A simple text message sent by an administrator to a user who is logged on to a Planning application. The message displays information to the user such as system availability, notification of application refresh, or application backups.

budget administrator A person responsible for setting up, configuring, maintaining, and controlling an application. Has all application privileges and data access permissions.

build method A method used to modify database outlines. Choice of a build method is based on the format of data in data source files.

business process A set of activities that collectively accomplish a business objective.

business rules Logical expressions or formulas that are created within an application to produce a desired set of resulting values.

cache A buffer in memory that holds data temporarily.

calc script A set of commands that define how a database is consolidated or aggregated. A calculation script may also contain commands that specify allocation and other calculation rules separate from the consolidation process.

calculated member in MaxL DML A member designed for analytical purposes and defined in the optional WITH section of a MaxL DML query.

calculated member in MaxL DML A member designed for analytical purposes and defined in the optional WITH section of a MaxL DML query.

calculation The process of aggregating data, or of running a calculation script on a database.

Calculation Manager A calculation module with which Planning, Financial Management, and Essbase users can design, validate, and administer business rules in a graphical environment.

calculation status A consolidation status that indicates that some values or formula calculations have changed. You must reconsolidate to get the correct values for the affected entity.

calendar User-defined time periods and their relationship to each other. Q1, Q2, Q3, and Q4 comprise a calendar or fiscal year.

cascade The process of creating multiple reports for a subset of member values.

Catalog pane Displays a list of elements available to the active section. If Query is the active section, a list of database tables is displayed. If Pivot is the active section, a list of results columns is displayed. If Dashboard is the active section, a list of embeddable sections, graphic tools, and control tools are displayed.

categories Groupings by which data is organized. For example, Month.

cause and effect map Depicts how the elements that form your corporate strategy relate and how they work together to meet your organization's strategic goals. A Cause and Effect map tab is automatically created for each Strategy map.

CDF See *custom-defined function (CDF)*.

CDM See *custom-defined macro (CDM)*.

cell (1) The data value at the intersection of dimensions in a multidimensional database; the intersection of a row and a column in a worksheet. (2) A logical group of nodes belonging to one administrative domain.

cell note A text annotation for a cell in an Essbase database. Cell notes are a type of LRO.

CHANGED status Consolidation status that indicates data for an entity has changed.

chart A graphical representation of spreadsheet data. The visual nature expedites analysis, color-coding, and visual cues that aid comparisons.

chart template A template that defines the metrics to display in Workspace charts.

child A member with a parent above it in the database outline.

choice list A list of members that a report designer can specify for each dimension when defining the report's point of view. A user who wants to change the point of view for a dimension that uses a choice list can select only the members specified in that defined member list or those members that meet the criteria defined in the function for the dynamic list.

clean block A data block that where the database is fully calculated, if a calculation script calculates all dimensions at once, or if the SET CLEARUPDATESTATUS command is used in a calculation script.

cluster An array of servers or databases that behave as a single resource which share task loads and provide failover support; eliminates one server or database as a single point of failure in a system.

clustered bar charts Charts in which categories are viewed side-by-side; useful for side-by-side category analysis; used only with vertical bar charts.

code page A mapping of bit combinations to a set of text characters. Different code pages support different sets of characters. Each computer contains a code page setting for the character set requirements of the language of the computer user. In the context of this document, code pages map characters to bit combinations for non-Unicode encodings. *See also [encoding](#).*

column A vertical display of information in a grid or table. A column can contain data from one field, derived data from a calculation, or textual information.

committed access An Essbase Kernel Isolation Level setting that affects how Essbase handles transactions. Under committed access, concurrent transactions hold long-term write locks and yield predictable results.

computed item A virtual column (as opposed to a column that is physically stored in the database or cube) that can be calculated by the database during a query, or by Interactive Reporting Studio in the Results section. Computed items are calculations of data based on functions, data items, and operators provided in the dialog box and can be included in reports or reused to calculate other data.

configuration file The security platform relies on XML documents to be configured by the product administrator or software installer. The XML document must be modified to indicate meaningful values for properties, specifying locations and attributes pertaining to the corporate authentication scenario.

connection file *See [Interactive Reporting connection file \(.oce\)](#).*

consolidated file (Parent) A file into which all of the business unit files are consolidated; contains the definition of the consolidation.

consolidation The process of aggregating data from dependent entities to parent entities. For example, if the dimension Year consists of the members Qtr1, Qtr2, Qtr3, and Qtr4, its consolidation is Year.

consolidation file (*.cns) The consolidation file is a graphical interface that enables you to add, delete or move Strategic Finance files in the consolidation process using either a Chart or Tree view. It also enables you to define and modify the consolidation.

consolidation rule Identifies the rule that is executed during the consolidation of the node of the hierarchy. This rule can contain customer specific formulas appropriate for the correct consolidation of parent balances. Elimination processing can be controlled within these rules.

content Information stored in the repository for any type of file.

content browser A Component that allows users to Browse and select content to be placed in a Workspace Page .

context variable A variable that is defined for a particular task flow to identify the context of the taskflow instance.

contribution The value added to a parent from a child entity. Each child has a contribution to its parent.

controls group Used in FDM to maintain and organize certification and assessment information, especially helpful for meeting Sarbanes-Oxley requirements.

conversion rate *See [exchange rate](#).*

cookie A segment of data placed on your computer by a Web site.

correlated subqueries Subqueries that are evaluated once for every row in the parent query; created by joining a topic item in the subquery with a topic in the parent query.

critical business area (CBA) An individual or a group organized into a division, region, plant, cost center, profit center, project team, or process; also called accountability team or business area.

critical success factor (CSF) A capability that must be established and sustained to achieve a strategic objective; owned by a strategic objective or a critical process and is a parent to one or more actions.

crosstab reporting Categorizes and summarizes data in table format. The table cells contain summaries of the data that fit within the intersecting categories. For example, a crosstab report of product sales information could show size attributes, such as Small and Large, as column headings and color attributes, such as Blue and Yellow, as row headings. The cell in the table where Large and Blue intersect could contain the total sales of all Blue products that are sized Large.

cube A block of data that contains three or more dimensions. An Essbase database is a cube.

cube deployment In Essbase Studio, the process of setting load options for a model to build an outline and load data into an Essbase application and database.

cube schema In Essbase Studio, the metadata elements, such as measures and hierarchies, representing the logical model of a cube.

currency conversion A process that converts currency values in a database from one currency into another. For example, to convert one U. S. dollar into the European euro, the exchange rate (for example, 0.923702) is multiplied with the dollar ($1 * 0.923702$). After conversion, the European euro amount is .92.

Currency Overrides In any input period, the selected input method can be overridden to enable input of that period's value as Default Currency/Items. To override the input method, enter a pound sign (#) either before or after the number.

currency partition A dimension type that separates local currency members from a base currency, as defined in an application. Identifies currency types, such as Actual, Budget, and Forecast.

custom calendar Any calendar created by an administrator.

custom dimension A dimension created and defined by users. Channel, product, department, project, or region could be custom dimensions.

custom property A property of a dimension or dimension member that is created by a user.

custom report A complex report from the Design Report module, composed of any combination of components.

custom-defined function (CDF) Essbase calculation functions developed in Java and added to the standard Essbase calculation scripting language using MaxL. *See also* [custom-defined macro \(CDM\)](#).

custom-defined macro (CDM) Essbase macros written with Essbase calculator functions and special macro functions. Custom-defined macros use an internal Essbase macro language that enables the combination of calculation functions and they operate on multiple input parameters. *See also* [custom-defined function \(CDF\)](#).

cycle through To perform multiple passes through a database while calculating it.

dashboard A collection of metrics and indicators that provide an interactive summary of your business. Dashboards enable you to build and deploy analytic applications.

data cache A buffer in memory that holds uncompressed data blocks.

data cell *See* [cell](#).

data file cache A buffer in memory that holds compressed data (PAG) files.

data form A grid display that enables users to enter data into the database from an interface such as a Web browser, and to view and analyze data or related text. Certain dimension member values are fixed, giving users a specific view into the data.

data function That computes aggregate values, including averages, maximums, counts, and other statistics, that summarize groupings of data.

data load location In FDM, a reporting unit responsible for submitting source data into the target system. Typically, there is one FDM data load location for each source file loaded to the target system.

data load rules A set of criteria that determines how to load data from a text-based file, a spreadsheet, or a relational data set into a database.

data lock Prevents changes to data according to specified criteria, such as period or scenario.

data mining The process of searching through an Essbase database for hidden relationships and patterns in a large amount of data.

data model A representation of a subset of database tables.

data value *See* [cell](#).

database connection File that stores definitions and properties used to connect to data sources and enables database references to be portable and widely used.

date measure In Essbase, a member tagged as "Date" in the dimension where measures are represented. The cell values are displayed as formatted dates. Dates as measures can be useful for types of analysis that are difficult to represent using the Time dimension. For example, an application may need to track acquisition dates for a series of capital assets, but the acquisition dates span too large a period to allow for feasible Time dimension modeling. *See also [typed measure](#).*

Default Currency Units Define the unit scale of data. For example, if you select to define your analysis in Thousands, and enter "10", this is interpreted as "10,000".

dense dimension In block storage databases, a dimension likely to contain data for every combination of dimension members. For example, time dimensions are often dense because they can contain all combinations of all members. *Contrast with [sparse dimension](#).*

dependent entity An entity that is owned by another entity in the organization.

derived text measure In Essbase Studio, a text measure whose values are governed by a predefined rule expressed as a range. For example, a derived text measure, called "Sales Performance Index," based on a measure Sales, could consist of the values "High," "Medium," and "Low." This derived text measure is defined to display "High," "Medium," and "Low" depending on the range in which the corresponding sales values fall. *See also [text measure](#).*

descendant Any member below a parent in the database outline. In a dimension that includes years, quarters, and months, the members Qtr2 and April are descendants of the member Year.

Design Report An interface in Web Analysis Studio for designing custom reports, from a library of components.

destination (1) For Business Rules and Calculation Manager, an intersection within the database where allocated values are stored. (2) Within a Profitability and Cost Management assignment, the receiving point for allocated values.

destination currency The currency to which balances are converted. You enter exchange rates and convert from the source currency to the destination currency. For example, when you convert from EUR to USD, the destination currency is USD.

detail chart A chart that provides the detailed information that you see in a Summary chart. Detail charts appear in the Investigate Section in columns below the Summary charts. If the Summary chart shows a Pie chart, then the Detail charts below represent each piece of the pie.

dimension A data category used to organize business data for retrieval and preservation of values. Dimensions usually contain hierarchies of related members grouped within them. For example, a Year dimension often includes members for each time period, such as quarters and months.

dimension build The process of adding dimensions and members to an Essbase outline.

dimension build rules Specifications, similar to data load rules, that Essbase uses to modify an outline. The modification is based on data in an external data source file.

dimension tab In the Pivot section, the tab that enables you to pivot data between rows and columns.

dimension table (1) A table that includes numerous attributes about a specific business process. (2) In Essbase Integration Services, a container in the OLAP model for one or more relational tables that define a potential dimension in Essbase.

dimension type A dimension property that enables the use of predefined functionality. Dimensions tagged as time have a predefined calendar functionality.

dimensionality In MaxL DML, the represented dimensions (and the order in which they are represented) in a set. For example, the following set consists of two tuples of the same dimensionality because they both reflect the dimensions (Region, Year): { (West, Feb), (East, Mar) }

direct rate A currency rate that you enter in the exchange rate table. The direct rate is used for currency conversion. For example, to convert balances from JPY to USD, In the exchange rate table, enter a rate for the period/scenario where the source currency is JPY and the destination currency is USD.

dirty block A data block containing cells that have been changed since the last calculation. Upper level blocks are marked as dirty if their child blocks are dirty (that is, they have been updated).

display type One of three Web Analysis formats saved to the repository: spreadsheet, chart, and pinboard.

dog-ear The flipped page corner in the upper right corner of the chart header area.

domain In data mining, a variable representing a range of navigation within data.

drill-down Navigation through the query result set using the dimensional hierarchy. Drilling down moves the user perspective from aggregated data to detail. For example, drilling down can reveal hierarchical relationships between years and quarters or quarters and months.

drill-through The navigation from a value in one data source to corresponding data in another source.

driver A driver is an allocation method that describes the mathematical relationship between the sources that utilize the driver, and the destinations to which those sources allocate cost or revenue.

duplicate alias name A name that occurs more than once in an alias table and that can be associated with more than one member in a database outline. Duplicate alias names can be used with duplicate member outlines only.

duplicate member name The multiple occurrence of a member name in a database, with each occurrence representing a different member. For example, a database has two members named “New York.” One member represents New York state and the other member represents New York city.

duplicate member outline A database outline containing duplicate member names.

Dynamic Calc and Store members A member in a block storage outline that Essbase calculates only upon the first retrieval of the value. Essbase then stores the calculated value in the database. Subsequent retrievals do not require calculating.

Dynamic Calc members A member in a block storage outline that Essbase calculates only at retrieval time. Essbase discards calculated values after completing the retrieval request.

dynamic calculation In Essbase, a calculation that occurs only when you retrieve data on a member that is tagged as Dynamic Calc or Dynamic Calc and Store. The member's values are calculated at retrieval time instead of being precalculated during batch calculation.

dynamic hierarchy In aggregate storage database outlines only, a hierarchy in which members are calculated at retrieval time.

dynamic member list A system-created named member set that is based on user-defined criteria. The list is refreshed automatically whenever it is referenced in the application. As dimension members are added and deleted, the list automatically reapplies the criteria to reflect the changes.

dynamic reference A pointer in the rules file to header records in a data source.

dynamic report A report containing data that is updated when you run the report.

Dynamic Time Series A process that performs period-to-date reporting in block storage databases.

dynamic view account An account type indicating that account values are calculated dynamically from the data that is displayed.

Eliminated Account An account that does not appear in the consolidated file.

elimination The process of zeroing out (eliminating) transactions between entities within an organization.

employee A user responsible for, or associated with, specific business objects. Employees need not work for an organization; for example, they can be consultants. Employees must be associated with user accounts for authorization purposes.

encoding A method for mapping bit combinations to characters for creating, storing, and displaying text. Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41. *See also code page and locale.*

ending period A period enabling you to adjust the date range in a chart. For example, an ending period of “month”, produces a chart showing information through the end of the current month.

Enterprise View An Administration Services feature that enables management of the Essbase environment from a graphical tree view. From Enterprise View, you can operate directly on Essbase artifacts.

entity A dimension representing organizational units. Examples: divisions, subsidiaries, plants, regions, products, or other financial reporting units.

Equity Beta The riskiness of a stock, measured by the variance between its return and the market return, indicated by an index called “beta”. For example, if a stock's return normally moves up or down 1.2% when the market moves up or down 1%, the stock has a beta of 1.2.

essbase.cfg An optional configuration file for Essbase. Administrators may edit this file to customize Essbase Server functionality. Some configuration settings may also be used with Essbase clients to override Essbase Server settings.

EssCell A function entered into a cell in Essbase Spreadsheet Add-in to retrieve a value representing an intersection of specific Essbase database members.

ESSCMD A command-line interface for performing Essbase operations interactively or through batch script files.

ESSLANG The Essbase environment variable that defines the encoding used to interpret text characters. *See also [encoding](#).*

ESSMSH *See [MaxL Shell](#).*

exceptions Values that satisfy predefined conditions. You can define formatting indicators or notify subscribing users when exceptions are generated.

exchange rate A numeric value for converting one currency to another. For example, to convert 1 USD into EUR, the exchange rate of 0.8936 is multiplied with the U.S. dollar. The European euro equivalent of \$1 is 0.8936.

exchange rate type An identifier for an exchange rate. Different rate types are used because there may be multiple rates for a period and year. Users traditionally define rates at period end for the average rate of the period and for the end of the period. Additional rate types are historical rates, budget rates, forecast rates, and so on. A rate type applies to one point in time.

expense account An account that stores periodic and year-to-date values that decrease net worth if they are positive.

Extensible Markup Language (XML) A language comprising a set of tags used to assign attributes to data that can be interpreted between applications according to a schema.

external authentication Logging on to Oracle's Hyperion applications with user information stored outside the applications, typically in a corporate directory such as MSAD or NTLM.

externally triggered events Non-time-based events for scheduling job runs.

Extract, Transform, and Load (ETL) Data source-specific programs for extracting data and migrating it to applications.

extraction command An Essbase reporting command that handles the selection, orientation, grouping, and ordering of raw data extracted from a database; begins with the less than (<) character.

fact table The central table in a star join schema, characterized by a foreign key and elements drawn from a dimension table. This table typically contains numeric data that can be related to all other tables in the schema.

Favorites gadget Contains links to Reporting and Analysis documents and URLs.

field An item in a data source file to be loaded into an Essbase database.

file delimiter Characters, such as commas or tabs, that separate fields in a data source.

filter A constraint on data sets that restricts values to specific criteria; for example, to exclude certain tables, metadata, or values, or to control access.

flow account An unsigned account that stores periodic and year-to-date values.

folder A file containing other files for the purpose of structuring a hierarchy.

footer Text or images at the bottom of report pages, containing dynamic functions or static text such as page numbers, dates, logos, titles or file names, and author names.

format Visual characteristics of documents or report objects.

format string In Essbase, a method for transforming the way cell values are displayed.

formula A combination of operators, functions, dimension and member names, and numeric constants calculating database members.

frame An area on the desktop. There are two main areas: the navigation and Workspace frames.

free-form grid An object for presenting, entering, and integrating data from different sources for dynamic calculations.

free-form reporting Creating reports by entering dimension members or report script commands in worksheets.

function A routine that returns values or database members.

gadget Simple, specialized, lightweight applications that provide easy viewing of EPM content and enable access to core Reporting and Analysis functionality.

genealogy data Additional data that is optionally generated after allocation calculations. This data enables reporting on all cost or revenue flows from start to finish through all allocation steps.

generation A layer in a hierarchical tree structure that defines member relationships in a database. Generations are ordered incrementally from the top member of the dimension (generation 1) down to the child members. Use the unique generation name to identify a layer in the hierarchical tree structure.

generic jobs Non-SQR Production Reporting or non-Interactive Reporting jobs.

global report command A command in a running report script that is effective until replaced by another global command or the file ends.

grid POV A means for specifying dimension members on a grid without placing dimensions in rows, columns, or page intersections. A report designer can set POV values at the grid level, preventing user POVs from affecting the grid. If a dimension has one grid value, you put the dimension into the grid POV instead of the row, column, or page.

group A container for assigning similar access permissions to multiple users.

GUI Graphical user interface

head up display A mode that shows your loaded Smart Space desktop including the background image above your Windows desktop.

highlighting Depending on your configuration, chart cells or ZoomChart details may be highlighted, indicating value status: red (bad), yellow (warning), or green (good).

Historical Average An average for an account over a number of historical periods.

holding company An entity that is part of a legal entity group, with direct or indirect investments in all entities in the group.

host A server on which applications and services are installed.

host properties Properties pertaining to a host, or if the host has multiple Install_Homes, to an Install_Home. The host properties are configured from the CMC.

Hybrid Analysis An analysis mapping low-level data stored in a relational database to summary-level data stored in Essbase, combining the mass scalability of relational systems with multidimensional data.

hyperlink A link to a file, Web page, or an intranet HTML page.

Hypertext Markup Language (HTML) A programming language specifying how Web browsers display data.

identity A unique identification for a user or group in external authentication.

image bookmarks Graphic links to Web pages or repository items.

IMPACTED status Indicates changes in child entities consolidating into parent entities.

implied share A member with one or more children, but only one is consolidated, so the parent and child share a value.

import format In FDM, defines the structure of the source file which enables the loading of a source data file to an FDM data load location.

inactive group A group for which an administrator has deactivated system access.

inactive service A service suspended from operating.

INACTIVE status Indicates entities deactivated from consolidation for the current period.

inactive user A user whose account has been deactivated by an administrator.

income account An account storing periodic and year-to-date values that, if positive, increase net worth.

index (1) A method where Essbase uses sparse-data combinations to retrieve data in block storage databases. (2) The index file.

index cache A buffer containing index pages.

index entry A pointer to an intersection of sparse dimensions. Index entries point to data blocks on disk and use offsets to locate cells.

index file An Essbase file storing block storage data retrieval information, residing on disk, and containing index pages.

index page A subdivision in an index file. Contains pointers to data blocks.

input data Data loaded from a source rather than calculated.

Install_Home A variable for the directory where EPM System products are installed. Refers to one instance of an EPM System product when multiple applications are installed on the same computer.

integration Process that is run to move data between EPM System products using Shared Services. Data integration definitions specify the data moving between a source application and a destination application, and enable the data movements to be grouped, ordered, and scheduled.

intelligent calculation A calculation method tracking updated data blocks since the last calculation.

Interactive Reporting connection file (.oce) Files encapsulating database connection information, including: the database API (ODBC, SQL*Net, etc.), database software, the database server network address, and database user name. Administrators create and publish Interactive Reporting connection files (.oce).

intercompany elimination See [elimination](#).

intercompany matching The process of comparing balances for pairs of intercompany accounts within an application. Intercompany receivables are compared to intercompany payables for matches. Matching accounts are used to eliminate intercompany transactions from an organization's consolidated totals.

intercompany matching report A report that compares intercompany account balances and indicates if the accounts are in, or out, of balance.

interdimensional irrelevance A situation in which a dimension does not intersect with other dimensions. Because the data in the dimension cannot be accessed from the non-intersecting dimensions, the non-intersecting dimensions are not relevant to that dimension.

intersection A unit of data representing the intersection of dimensions in a multidimensional database; also, a worksheet cell.

inrastage assignment Assignments in the financial flow that are assigned to objects within the same stage.

introspection A deep inspection of a data source to discover hierarchies based on the inherent relationships in the database. *Contrast with [scraping](#).*

Investigation See [drill-through](#).

isolation level An Essbase Kernel setting that determines the lock and commit behavior of database operations. Choices are: committed access and uncommitted access.

iteration A “pass” of the budget or planning cycle in which the same version of data is revised and promoted.

Java Database Connectivity (JDBC) A client-server communication protocol used by Java based clients and relational databases. The JDBC interface provides a call-level API for SQL-based database access.

job output Files or reports produced from running a job.

jobs Documents with special properties that can be launched to generate output. A job can contain Interactive Reporting, SQR Production Reporting, or generic documents.

join A link between two relational database tables or topics based on common content in a column or row. A join typically occurs between identical or similar items within different tables or topics. For example, a record in the Customer table is joined to a record in the Orders table because the Customer ID value is the same in each table.

journal entry (JE) A set of debit/credit adjustments to account balances for a scenario and period.

JSP Java Server Pages.

KeyContacts gadget Contains a group of Smart Space users and provides access to Smart Space Collaborator. For example, you can have a KeyContacts gadget for your marketing team and another for your development team.

latest A Spreadsheet key word used to extract data values from the member defined as the latest time period.

layer (1) The horizontal location of members in a hierarchical structure, specified by generation (top down) or level (bottom up). (2) Position of objects relative to other objects. For example, in the Sample Basic database, Qtr1 and Qtr4 are in the same layer, so they are also in the same generation, but in a database with a ragged hierarchy, Qtr1 and Qtr4 might not be in same layer, though they are in the same generation.

layout area Used to designate an area on a Workspace Page where content can be placed.

legend box A box containing labels that identify the data categories of a dimension.

level A layer in a hierarchical tree structure that defines database member relationships. Levels are ordered from the bottom dimension member (level 0) up to the parent members.

level 0 block A data block for combinations of sparse, level 0 members.

level 0 member A member that has no children.

liability account An account type that stores “point in time” balances of a company's liabilities. Examples of liability accounts include accrued expenses, accounts payable, and long term debt.

life cycle management The process of managing application information from inception to retirement.

Lifecycle Management Utility A command-line utility for migrating applications and artifacts.

line chart A chart that displays one to 50 data sets, each represented by a line. A line chart can display each line stacked on the preceding ones, as represented by an absolute value or a percent.

line item detail The lowest level of detail in an account.

lineage The relationship between different metadata elements showing how one metadata element is derived from one or more other metadata elements, ultimately tracing the metadata element to its physical source. In Essbase Studio, a lineage viewer displays the relationships graphically. *See also* [traceability](#).

link (1) A reference to a repository object. Links can reference folders, files, shortcuts, and other links. (2) In a task flow, the point where the activity in one stage ends and another begins.

link condition A logical expression evaluated by the taskflow engine to determine the sequence of launching taskflow stages.

linked data model Documents that are linked to a master copy in a repository.

linked partition A shared partition that enables you to use a data cell to link two databases. When a user clicks a linked cell in a worksheet, Essbase opens a new sheet displaying the dimensions in the linked database. The user can then drill down those dimensions.

linked reporting object (LRO) A cell-based link to an external file such as cell notes, URLs, or files with text, audio, video, or pictures. (Only cell notes are supported for Essbase LROs in Financial Reporting.) *Contrast with* [local report object](#).

local currency An input currency type. When an input currency type is not specified, the local currency matches the entity's base currency.

local report object A report object that is not linked to a Financial Reporting report object in Explorer. *Contrast with* [linked reporting object \(LRO\)](#).

local results A data model's query results. Results can be used in local joins by dragging them into the data model. Local results are displayed in the catalog when requested.

locale A computer setting that specifies a location's language, currency and date formatting, data sort order, and the character set encoding used on the computer. Essbase uses only the encoding portion. *See also* [encoding](#) and [ESSLANG](#).

locale header record A text record at the beginning of some non-Unicode-encoded text files, such as scripts, that identifies the encoding locale.

location alias A descriptor that identifies a data source. The location alias specifies a server, application, database, user name, and password. Location aliases are set by DBAs at the database level using Administration Services Console, ESSCMD, or the API.

locked A user-invoked process that prevents users and processes from modifying data.

locked data model Data models that cannot be modified by a user.

LOCKED status A consolidation status indicating that an entity contains data that cannot be modified.

Log Analyzer An Administration Services feature that enables filtering, searching, and analysis of Essbase logs.

logic group In FDM, contains one or more logic accounts that are generated after a source file is loaded into FDM. Logic accounts are calculated accounts that are derived from the source data.

LRO See *linked reporting object (LRO)*.

managed server An application server process running in its own Java Virtual Machine (JVM).

manual stage A stage that requires human intervention to complete.

Map File Used to store the definition for sending data to or retrieving data from an external database. Map files have different extensions (.mps to send data; .mpr to retrieve data).

Map Navigator A feature that displays your current position on a Strategy, Accountability, or Cause and Effect map, indicated by a red outline.

Marginal Tax Rate Used to calculate the after-tax cost of debt. Represents the tax rate applied to the last earned income dollar (the rate from the highest tax bracket into which income falls) and includes federal, state and local taxes. Based on current level of taxable income and tax bracket, you can predict marginal tax rate.

Market Risk Premium The additional rate of return paid over the risk-free rate to persuade investors to hold “riskier” investments than government securities. Calculated by subtracting the risk-free rate from the expected market return. These figures should closely model future market conditions.

master data model An independent data model that is referenced as a source by multiple queries. When used, “Locked Data Model” is displayed in the Query section's Content pane; the data model is linked to the master data model displayed in the Data Model section, which an administrator may hide.

mathematical operator A symbol that defines how data is calculated in formulas and outlines. Can be any of the standard mathematical or Boolean operators; for example, +, -, *, /, and %.

MaxL The multidimensional database access language for Essbase, consisting of a data definition language (MaxL DDL) and a data manipulation language (MaxL DML). See also *MaxL DDL*, *MaxL DML*, and *MaxL Shell*.

MaxL DDL Data definition language used by Essbase for batch or interactive system-administration tasks.

MaxL DML Data manipulation language used in Essbase for data query and extraction.

MaxL Perl Module A Perl module (essbase.pm) that is part of Essbase MaxL DDL. This module can be added to the Perl package to provide access to Essbase databases from Perl programs.

MaxL Script Editor A script-development environment in Administration Services Console. MaxL Script Editor is an alternative to using a text editor and the MaxL Shell for administering Essbase with MaxL scripts.

MaxL Shell An interface for passing MaxL statements to Essbase Server. The MaxL Shell executable file is located in the Essbase bin directory (UNIX: essmsh, Windows: essmsh.exe).

MDX (multidimensional expression) The language that give instructions to OLE DB for OLAP- compliant databases, as SQL is used for relational databases. When you build the OLAPQuery section's Outliner, Interactive Reporting Clients translate requests into MDX instructions. When you process the query, MDX is sent to the database server, which returns records that answer your query. See also *SQL spreadsheet*.

measures Numeric values in an OLAP database cube that are available for analysis. Measures are margin, cost of goods sold, unit sales, budget amount, and so on. See also *fact table*.

member A discrete component within a dimension. A member identifies and differentiates the organization of similar units. For example, a time dimension might include such members as Jan, Feb, and Qtr1.

member list A named group, system- or user-defined, that references members, functions, or member lists within a dimension.

member load In Integration Services, the process of adding dimensions and members (without data) to Essbase outlines.

member selection report command A type of Report Writer command that selects member ranges based on outline relationships, such as sibling, generation, and level.

member-specific report command A type of Report Writer formatting command that is executed as it is encountered in a report script. The command affects only its associated member and executes the format command before processing the member.

merge A data load option that clears values only from the accounts specified in the data load file and replaces them with values in the data load file.

metadata A set of data that defines and describes the properties and attributes of the data stored in a database or used by an application. Examples of metadata are dimension names, member names, properties, time periods, and security.

metadata elements Metadata derived from data sources and other metadata that is stored and cataloged for Essbase Studio use.

metadata sampling The process of retrieving a sample of members in a dimension in a drill-down operation.

metadata security Security set at the member level to restrict users from accessing certain outline members.

metaoutline In Integration Services, a template containing the structure and rules for creating an Essbase outline from an OLAP model.

metric A numeric measurement computed from business data to help assess business performance and analyze company trends.

migration The process of copying applications, artifacts, or users from one environment or computer to another; for example, from a testing environment to a production environment.

migration audit report A report generated from the migration log that provides tracking information for an application migration.

migration definition file (.mdf) A file that contains migration parameters for an application migration, enabling batch script processing.

migration log A log file that captures all application migration actions and messages.

migration snapshot A snapshot of an application migration that is captured in the migration log.

MIME Type (Multipurpose Internet Mail Extension) An attribute that describes the data format of an item, so that the system knows which application should open the object. A file's mime type is determined by the file extension or HTTP header. Plug-ins tell browsers what mime types they support and what file extensions correspond to each mime type.

mining attribute In data mining, a class of values used as a factor in analysis of a set of data.

minireport A report component that includes layout, content, hyperlinks, and the query or queries to load the report. Each report can include one or more minireports.

minischema A graphical representation of a subset of tables from a data source that represents a data modeling context.

missing data (#MISSING) A marker indicating that data in the labeled location does not exist, contains no value, or was never entered or loaded. For example, missing data exists when an account contains data for a previous or future period but not for the current period.

model (1) In data mining, a collection of an algorithm's findings about examined data. A model can be applied against a wider data set to generate useful information about that data. (2) A file or content string containing an application-specific representation of data. Models are the basic data managed by Shared Services, of two major types: dimensional and non-dimensional application objects. (3) In Business Modeling, a network of boxes connected to represent and calculate the operational and financial flow through the area being examined.

monetary A money-related value.

multidimensional database A method of organizing, storing, and referencing data through three or more dimensions. An individual value is the intersection point for a set of dimensions. *Contrast with relational database.*

multiload An FDM feature that allows the simultaneous loading of multiple periods, categories, and locations.

My Workspace Page A page created with content from multiple sources including documents, URL, and other content types. Enables a user to aggregate content from Oracle and non-Oracle sources.

named set In MaxL DML, a set with its logic defined in the optional WITH section of a MaxL DML query. The named set can be referenced multiple times in the query.

native authentication The process of authenticating a user name and password from within the server or application.

nested column headings A report column heading format that displays data from multiple dimensions. For example, a column heading that contains Year and Scenario members is a nested column. The nested column heading shows Q1 (from the Year dimension) in the top line of the heading, qualified by Actual and Budget (from the Scenario dimension) in the bottom line of the heading.

NO DATA status A consolidation status indicating that this entity contains no data for the specified period and account.

non-dimensional model A Shared Services model type that includes application objects such as security files, member lists, calculation scripts, and Web forms.

non-unique member name See *duplicate member name*.

note Additional information associated with a box, measure, scorecard or map element.

Notifications gadget Shows notification message history received from other users or systems.

null value A value that is absent of data. Null values are not equal to zero.

numeric attribute range A feature used to associate a base dimension member that has a discrete numeric value with an attribute that represents a value range. For example, to classify customers by age, an Age Group attribute dimension can contain members for the following age ranges: 0-20, 21-40, 41-60, and 61-80. Each Customer dimension member can be associated with an Age Group range. Data can be retrieved based on the age ranges rather than on individual age values.

ODBC Open Database Connectivity. A database access method used from any application regardless of how the database management system (DBMS) processes the information.

OK status A consolidation status indicating that an entity has already been consolidated, and that data has not changed below it in the organization structure.

OLAP Metadata Catalog In Integration Services, a relational database containing metadata describing the nature, source, location, and type of data that is pulled from the relational data source.

OLAP model In Integration Services, a logical model (star schema) that is created from tables and columns in a relational database. The OLAP model is then used to generate the structure of a multidimensional database.

online analytical processing (OLAP) A multidimensional, multiuser, client-server computing environment for users who analyze consolidated enterprise data in real time. OLAP systems feature drill-down, data pivoting, complex calculations, trend analysis, and modeling.

Open Database Connectivity (ODBC) Standardized application programming interface (API) technology that allows applications to access multiple third-party databases.

organization An entity hierarchy that defines each entity and their relationship to others in the hierarchy.

origin The intersection of two axes.

outline The database structure of a multidimensional database, including all dimensions, members, tags, types, consolidations, and mathematical relationships. Data is stored in the database according to the structure defined in the outline.

outline synchronization For partitioned databases, the process of propagating outline changes from one database to another database.

P&L accounts (P&L) Profit and loss accounts. Refers to a typical grouping of expense and income accounts that comprise a company's income statement.

page A display of information in a grid or table often represented by the Z-axis. A page can contain data from one field, derived data from a calculation, or text.

page file Essbase data file.

page heading A report heading type that lists members represented on the current page of the report. All data values on the page have the members in the page heading as a common attribute.

page member A member that determines the page axis.

palette A JASC compliant file with a .PAL extension. Each palette contains 16 colors that complement each other and can be used to set the dashboard color elements.

parallel calculation A calculation option. Essbase divides a calculation into tasks and calculates some tasks simultaneously.

parallel data load In Essbase, the concurrent execution of data load stages by multiple process threads.

parallel export The ability to export Essbase data to multiple files. This may be faster than exporting to a single file, and it may resolve problems caused by a single data file becoming too large for the operating system to handle.

parent adjustments The journal entries that are posted to a child in relation to its parent.

parents The entities that contain one or more dependent entities that report directly to them. Because parents are both entities and associated with at least one node, they have entity, node, and parent information associated with them.

partition area A sub cube within a database. A partition is composed of one or more areas of cells from a portion of the database. For replicated and transparent partitions, the number of cells within an area must be the same for the data source and target to ensure that the two partitions have the same shape. If the data source area contains 18 cells, the data target area must also contain 18 cells to accommodate the number of values.

partitioning The process of defining areas of data that are shared or linked between data models. Partitioning can affect the performance and scalability of Essbase applications.

pattern matching The ability to match a value with any or all characters of an item entered as a criterion. Missing characters may be represented by wild card values such as a question mark (?) or an asterisk (*). For example, "Find all instances of apple" returns apple, but "Find all instances of apple*" returns apple, applesauce, applecranberry, and so on.

percent consolidation The portion of a child's values that is consolidated to its parent.

percent control Identifies the extent to which an entity is controlled within the context of its group.

percent ownership Identifies the extent to which an entity is owned by its parent.

performance indicator An image file used to represent measure and scorecard performance based on a range you specify; also called a status symbol. You can use the default performance indicators or create an unlimited number of your own.

periodic value method (PVA) A process of currency conversion that applies the periodic exchange rate values over time to derive converted results.

permission A level of access granted to users and groups for managing data or other users and groups.

persistence The continuance or longevity of effect for any Essbase operation or setting. For example, an Essbase administrator may limit the persistence of user name and password validity.

personal pages A personal window to repository information. You select what information to display and its layout and colors.

personal recurring time events Reusable time events that are accessible only to the user who created them.

personal variable A named selection statement of complex member selections.

perspective A category used to group measures on a scorecard or strategic objectives within an application. A perspective can represent a key stakeholder (such as a customer, employee, or shareholder/financial) or a key competency area (such as time, cost, or quality).

pie chart A chart that shows one data set segmented in a pie formation.

pinboard One of the three data object display types. Pinboards are graphics, composed of backgrounds and interactive icons called pins. Pinboards require traffic lighting definitions.

pins Interactive icons placed on graphic reports called pinboards. Pins are dynamic. They can change images and traffic lighting color based on the underlying data values and analysis tools criteria.

pivot The ability to alter the perspective of retrieved data. When Essbase first retrieves a dimension, it expands data into rows. You can then pivot or rearrange the data to obtain a different viewpoint.

planner Planners, who comprise the majority of users, can input and submit data, use reports that others create, execute business rules, use task lists, enable e-mail notification for themselves, and use Smart View.

planning unit A data slice at the intersection of a scenario, version, and entity; the basic unit for preparing, reviewing, annotating, and approving plan data.

plot area The area bounded by X, Y, and Z axes; for pie charts, the rectangular area surrounding the pie.

plug account An account in which the system stores any out of balance differences between intercompany account pairs during the elimination process.

post stage assignment Assignments in the allocation model that are assigned to locations in a subsequent model stage.

POV (point of view) A feature for setting data focus by selecting members that are not already assigned to row, column, or page axes. For example, selectable POVs in FDM could include location, period, category, and target category. In another example, using POV as a filter in Smart View, you could assign the Currency dimension to the POV and select the Euro member. Selecting this POV in data forms displays data in Euro values.

precalculation Calculating the database prior to user retrieval.

precision Number of decimal places displayed in numbers.

predefined drill paths Paths used to drill to the next level of detail, as defined in the data model.

presentation A playlist of Web Analysis documents, enabling reports to be grouped, organized, ordered, distributed, and reviewed. Includes pointers referencing reports in the repository.

preserve formulas User-created formulas kept within a worksheet while retrieving data.

primary measure A high-priority measure important to your company and business needs. Displayed in the Contents frame.

process monitor report Displays a list of locations and their positions within the FDM data conversion process. You can use the process monitor report to monitor the status of the closing process. The report is time-stamped. Therefore, it can be used to determine to which locations at which time data was loaded.

product In Shared Services, an application type, such as Planning or Performance Scorecard.

Production Reporting See [SQR Production Reporting](#).

project An instance of EPM System products grouped together in an implementation. For example, a Planning project may consist of a Planning application, an Essbase cube, and a Financial Reporting server instance.

property A characteristic of an artifact, such as size, type, or processing instructions.

provisioning The process of granting users and groups specific access permissions to resources.

proxy server A server acting as an intermediary between workstation users and the Internet to ensure security.

public job parameters Reusable, named job parameters created by administrators and accessible to users with requisite access privileges.

public recurring time events Reusable time events created by administrators and accessible through the access control system.

PVA See [periodic value method \(PVA\)](#).

qualified name A member name in a qualified format that differentiates duplicate member names in a duplicate member outline. For example, [Market].[East].[State].[New York] or [Market].[East].[City].[New York]

query Information requests from data providers. For example, used to access relational data sources.

query governor An Essbase Integration server parameter or Essbase server configuration setting that controls the duration and size of queries made to data sources.

range A set of values including upper and lower limits, and values falling between limits. Can contain numbers, amounts, or dates.

reciprocal assignment An assignment in the financial flow that also has the source as one of its destinations.

reconfigure URL URL used to reload servlet configuration settings dynamically when users are already logged on to the Workspace.

record In a database, a group of fields making up one complete entry. For example, a customer record may contain fields for name, address, telephone number, and sales data.

recurring template A journal template for making identical adjustments in every period.

recurring time event An event specifying a starting point and the frequency for running a job.

redundant data Duplicate data blocks that Essbase retains during transactions until Essbase commits updated blocks.

regular journal A feature for entering one-time adjustments for a period. Can be balanced, balanced by entity, or unbalanced.

Related Accounts The account structure groups all main and related accounts under the same main account number. The main account is distinguished from related accounts by the first suffix of the account number.

relational database A type of database that stores data in related two-dimensional tables. *Contrast with [multidimensional database](#).*

replace A data load option that clears existing values from all accounts for periods specified in the data load file, and loads values from the data load file. If an account is not specified in the load file, its values for the specified periods are cleared.

replicated partition A portion of a database, defined through Partition Manager, used to propagate an update to data mastered at one site to a copy of data stored at another site. Users can access the data as though it were part of their local database.

Report Extractor An Essbase component that retrieves report data from the Essbase database when report scripts are run.

report object In report designs, a basic element with properties defining behavior or appearance, such as text boxes, grids, images, and charts.

report script A text file containing Essbase Report Writer commands that generate one or more production reports.

Report Viewer An Essbase component that displays complete reports after report scripts are run.

reporting currency The currency used to prepare financial statements, and converted from local currencies to reporting currencies.

repository Stores metadata, formatting, and annotation information for views and queries.

resources Objects or services managed by the system, such as roles, users, groups, files, and jobs.

restore An operation to reload data and structural information after a database has been damaged or destroyed, typically performed after shutting down and restarting the database.

restructure An operation to regenerate or rebuild the database index and, in some cases, data files.

result frequency The algorithm used to create a set of dates to collect and display results.

review level A Process Management review status indicator representing the process unit level, such as Not Started, First Pass, Submitted, Approved, and Published.

Risk Free Rate The rate of return expected from “safer” investments such as long-term U.S. government securities.

role The means by which access permissions are granted to users and groups for resources.

roll-up See [consolidation](#).

root member The highest member in a dimension branch.

RSC services Services that are configured with Remote Service Configurator, including Repository Service, Service Broker, Name Service, Event Service, and Job Service.

runtime prompt A variable that users enter or select before a business rule is run.

sampling The process of selecting a representative portion of an entity to determine the entity's characteristics. See also [metadata sampling](#).

saved assumptions User-defined Planning assumptions that drive key business calculations (for example, the cost per square foot of office floor space).

scaling Scaling determines the display of values in whole numbers, tens, hundreds, thousands, millions, and so on.

scenario A dimension for classifying data (for example, Actuals, Budget, Forecast1, and Forecast2).

scope The area of data encompassed by any Essbase operation or setting; for example, the area of data affected by a security setting. Most commonly, scope refers to three levels of granularity, where higher levels encompass lower levels. From highest to lowest, these levels are as follows: the entire system (Essbase Server), applications on Essbase servers, or databases within Essbase server applications. See also [persistence](#).

score The level at which targets are achieved, usually expressed as a percentage of the target.

scorecard Business object that represents the progress of an employee, strategy element, or accountability element toward goals. Scorecards ascertain this progress based on data collected for each measure and child scorecard added to the scorecard.

scrapping An inspection of a data source to derive the most basic metadata elements from it. Contrast with [introspection](#).

Search gadget Searches the Reporting and Analysis repository. The Search gadget looks for a match in the document keywords and description, which are set when you import a document.

secondary measure A low-priority measure, less important than primary measures. Secondary measures do not have Performance reports but can be used on scorecards and to create dimension measure templates.

security agent A Web access management provider (for example, Netegrity SiteMinder) that protects corporate Web resources.

security platform A framework enabling EPM System products to use external authentication and single sign-on.

serial calculation The default calculation setting. Divides a calculation pass into tasks and calculates one task at a time.

services Resources that enable business items to be retrieved, changed, added, or deleted. Examples: Authorization and Authentication.

servlet A piece of compiled code executable by a Web server.

shared member A member that shares storage space with another member of the same name, preventing duplicate calculation of members that occur multiple times in an Essbase outline.

Shared Services Registry Part of the Shared Services database, the Shared Services Registry stores and re-uses information for most installed EPM System products, including installation directories, database settings, deployment settings, computer names, ports, servers, URLs, and dependent service data.

Shared Workspace Page Workspace Pages shared across an organization which are stored in a special System folder and can be accessed by authorized users from the Shared Workspace Pages Navigate menu.

sibling A child member at the same generation as another child member and having the same immediate parent. For example, the members Florida and New York are children of East and each other's siblings.

single sign-on Ability to access multiple EPM System products after a single login using external credentials.

smart slice In Smart View, a reusable perspective of a data source that contains a restricted set of dimensions or dimension members.

Smart Space client software Runs on the client's computer and provides gadgets, instant collaboration and access to the Reporting and Analysis repository. It is composed of the Smart Space framework and gadgets.

Smart Space Collaborator A service that enables users or systems to send messages and share Reporting and Analysis repository content. The message can take many forms, including instant message style discussions, meetings, and toast messages.

smart tags Keywords in Microsoft Office applications that are associated with predefined actions available from the Smart Tag menu. In EPM System products, smart tags can also be used to import Reporting and Analysis content, and access Financial Management and Essbase functions.

SmartBook gadget Contains documents from the Reporting and Analysis repository or URLs. All documents are loaded when the SmartBook is opened so you can access all content immediately.

SmartCut A link to a repository item, in URL form.

snapshot Read-only data from a specific time.

source currency The currency from which values originate and are converted through exchange rates to the destination currency.

sparse dimension In block storage databases, a dimension unlikely to contain data for all member combinations when compared to other dimensions. For example, not all customers have data for all products. *Contrast with [dense dimension](#).*

SPF files Printer-independent files created by an SQR Production Reporting server, containing a representation of the actual formatted report output, including fonts, spacing, headers, footers, and so on.

Spotlighter A tool that enables color coding based on selected conditions.

SQL spreadsheet A data object that displays the result set of a SQL query.

SQR Production Reporting A specialized programming language for data access, data manipulation, and creating SQR Production Reporting documents.

stage A task description that forms one logical step within a taskflow, usually performed by an individual. A stage can be manual or automated.

stage action For automated stages, the invoked action that executes the stage.

staging area A database that you create to meet the needs of a specific application. A staging area is a snapshot or restructured version of one or more RDBMSs.

standard dimension A dimension that is not an attribute dimension.

standard journal template A journal function used to post adjustments that have common adjustment information for each period. For example, you can create a standard template that contains the common account IDs, entity IDs, or amounts, then use the template as the basis for many regular journals.

Status bar The status bar at the bottom of the screen displays helpful information about commands, accounts, and the current status of your data file.

stored hierarchy In aggregate storage databases outlines only. A hierarchy in which the members are aggregated according to the outline structure. Stored hierarchy members have certain restrictions, for example, they cannot contain formulas.

strategic objective (SO) A long-term goal defined by measurable results. Each strategic objective is associated with one perspective in the application, has one parent, the entity, and is a parent to critical success factors or other strategic objectives.

Strategy map Represents how the organization implements high-level mission and vision statements into lower-level, constituent strategic goals and objectives.

structure view Displays a topic as a simple list of component data items.

Structured Query Language A language used to process instructions to relational databases.

Subaccount Numbering A system for numbering subaccounts using non-sequential, whole numbers.

subscribe Flags an item or folder to receive automatic notification whenever the item or folder is updated.

Summary chart In the Investigates Section, rolls up detail charts shown below in the same column, plotting metrics at the summary level at the top of each chart column.

super service A special service used by the startCommonServices script to start the RSC services.

supervisor A user with full access to all applications, databases, related files, and security mechanisms for a server.

supporting detail Calculations and assumptions from which the values of cells are derived.

suppress rows Excludes rows containing missing values, and underscores characters from spreadsheet reports.

symmetric multiprocessing (SMP) A server architecture that enables multiprocessing and multithreading. Performance is not significantly degraded when a large number of users connect to a single instance simultaneously.

sync Synchronizes Shared Services and application models.

synchronized The condition that exists when the latest version of a model resides in both the application and in Shared Services. *See also* [model](#).

system extract Transfers data from an application's metadata into an ASCII file.

tabs Navigable views of accounts and reports in Strategic Finance.

target Expected results of a measure for a specified period of time (day, quarter, and so on).

task list A detailed status list of tasks for a particular user.

taskflow The automation of a business process in which tasks are passed from one taskflow participant to another according to procedural rules.

taskflow definition Represents business processes in the taskflow management system. Consists of a network of stages and their relationships; criteria indicating the start and end of the taskflow; and information about individual stages, such as participants, associated applications, associated activities, and so on.

taskflow instance Represents a single instance of a taskflow including its state and associated data.

taskflow management system Defines, creates, and manages the execution of a taskflow including: definitions, user or application interactions, and application executables.

taskflow participant The resource who performs the task associated with the taskflow stage instance for both manual and automated stages.

Taxes - Initial Balances Strategic Finance assumes that the Initial Loss Balance, Initial Gain Balance and the Initial Balance of Taxes Paid entries have taken place in the period before the first Strategic Finance time period.

TCP/IP *See* [Transmission Control Protocol/Internet Protocol \(TCP/IP\)](#).

template A predefined format designed to retrieve particular data consistently.

text list In Essbase, an object that stores text values mapped to numeric identifiers. Text Lists enable the use of text measures.

text measure A data type that allows measure values to be expressed as text. In Essbase, a member tagged as "Text" in the dimension where measures are represented. The cell values are displayed as predefined text. For example, the text measure "Satisfaction Index" may have the values Low, Medium, and High. *See also* [typed measure](#), [text list](#), [derived text measure](#).

time dimension Defines the time period that the data represents, such as fiscal or calendar periods.

time events Triggers for execution of jobs.

time line viewer An FDM feature that allows a user to view dates and times of completed process flow steps for specific locations.

time scale Displays metrics by a specific period in time, such as monthly or quarterly.

time series reporting A process for reporting data based on a calendar date (for example, year, quarter, month, or week).

Title bar Displays the Strategic Finance name, the file name, and the scenario name Version box.

toast message Messages that appear in the lower right corner of the screen and fade in and out.

token An encrypted identification of one valid user or group on an external authentication system.

top and side labels Column and row headings on the top and sides of a Pivot report.

top-level member A dimension member at the top of the tree in a dimension outline hierarchy, or the first member of the dimension in sort order if there is no hierarchical relationship among dimension members. The top-level member name is generally the same as the dimension name if a hierarchical relationship exists.

trace allocations A feature of Profitability and Cost Management that enables you to visually follow the flow of financial data, either forwards or backwards, from a single intersection throughout the model.

trace level Defines the level of detail captured in the log file.

traceability The ability to track a metadata element to its physical source. For example, in Essbase Studio, a cube schema can be traced from its hierarchies and measure hierarchies, to its dimension elements, date/time elements, and measures, and ultimately, to its physical source elements.

traffic lighting Color-coding of report cells, or pins based on a comparison of two dimension members, or on fixed limits.

transformation (1) Transforms artifacts so that they function properly in the destination environment after application migration. (2) In data mining, modifies data (bidirectionally) flowing between the cells in the cube and the algorithm.

translation See [currency conversion](#).

Transmission Control Protocol/Internet Protocol (TCP/IP) A standard set of communication protocols linking computers with different operating systems and internal architectures. TCP/IP utilities are used to exchange files, send mail, and store data to various computers that are connected to local and wide area networks.

transparent login Logs in authenticated users without launching the login screen.

transparent partition A shared partition that enables users to access and change data in a remote database as though it is part of a local database

triangulation A means of converting balances from one currency to another via a third common currency. In Europe, this is the euro for member countries. For example, to convert from French franc to Italian lira, the common currency is defined as European euro. Therefore, in order to convert balances from French franc to Italian lira, balances are converted from French franc to European euro and from European euro to Italian lira.

triggers An Essbase feature whereby data is monitored according to user-specified criteria which when met cause Essbase to alert the user or system administrator.

trusted password A password that enables users authenticated for one product to access other products without reentering their passwords.

trusted user Authenticated user.

tuple MDX syntax element that references a cell as an intersection of a member from each dimension. If a dimension is omitted, its top member is implied. Examples: (Jan); (Jan, Sales); ([Jan], [Sales], [Cola], [Texas], [Actual])

two-pass An Essbase property that is used to recalculate members that are dependent on the calculated values of other members. Two-pass members are calculated during a second pass through the outline.

typed measure In Essbase, a member tagged as “Text” or “Date” in the dimension where measures are represented. The cell values are displayed as predefined text or dates.

unary operator A mathematical indicator (+, -, *, /, %) associated with an outline member. The unary operator defines how the member is calculated during a database roll-up.

Unicode-mode application An Essbase application wherein character text is encoded in UTF-8, enabling users with computers set up for different languages to share application data.

Uniform Resource Locator The address of a resource on the Internet or an intranet.

unique member name A non-shared member name that exists only once in a database outline.

unique member outline A database outline that is not enabled for duplicate member names.

upgrade The process of replacing an earlier software release with a current release or replacing one product with another.

upper-level block A type of data block wherein at least one of the sparse members is a parent-level member.

user directory A centralized location for user and group information. Also known as a repository or provider.

user variable Dynamically renders data forms based on a user's member selection, displaying only the specified entity. For example, user variable named Department displays specific departments and employees.

user-defined attribute (UDA) User-defined attribute, associated with members of an outline to describe a characteristic of the members. Users can use UDAs to return lists of members that have the specified UDA associated with them.

user-defined member list A named, static set of members within a dimension defined by the user.

validation A process of checking a business rule, report script, or partition definition against the outline to make sure that the object being checked is valid. For example, in FDM, validation rules ensure that certain conditions are met after data is loaded from FDM to the target application.

value dimension Used to define input value, translated value, and consolidation detail.

variance Difference between two values (for example, planned and actual value).

varying attribute An attribute association that changes over one or more dimensions. It can be used to track a value in relation to these dimensions; for example, the varying attribute Sales Representative, associated with the Product dimension, can be used to track the value Customer Sales of several different sales representatives in relation to the Time dimension. Varying attributes can also be used for member selection, such as finding the Products that a Sales Representative was responsible for in May.

version Possible outcome used within the context of a scenario of data. For example, Budget - Best Case and Budget - Worst Case where Budget is scenario and Best Case and Worst Case are versions.

view Representation of either a year-to-date or periodic display of data.

visual cue A formatted style, such as a font or a color, that highlights specific types of data values. Data values may be dimension members; parent, child, or shared members; dynamic calculations; members containing a formula; read only data cells; read and write data cells; or linked objects.

Web server Software or hardware hosting intranet or Internet Web pages or Web applications.

weight Value assigned to an item on a scorecard that indicates the relative importance of that item in the calculation of the overall scorecard score. The weighting of all items on a scorecard accumulates to 100%. For example, to recognize the importance of developing new features for a product, the measure for New Features Coded on a developer's scorecard would be assigned a higher weighting than a measure for Number of Minor Defect Fixes.

wild card Character that represents any single character or group of characters (*) in a search string.

WITH section In MaxL DML, an optional section of the query used for creating re-usable logic to define sets or members. Sets or custom members can be defined once in the WITH section, and then referenced multiple times during a query.

work flow The steps required to process data from start to finish in FDM. The workflow consists of Import (loading data from the GL file), Validate (ensures all members are mapped to a valid account), Export (loads the mapped members to the target application), and Check (verifies accuracy of data by processing data with user-defined validation rules).

workbook An entire spreadsheet file with many worksheets.

Workspace Page A page created with content from multiple sources including documents, URL, and other content types. Enables a user to aggregate content from Oracle and non-Oracle sources.

write-back The ability for a retrieval client, such as a spreadsheet, to update a database value.

ws.conf A configuration file for Windows platforms.

wsconf_platform A configuration file for UNIX platforms.

XML See *Extensible Markup Language (XML)*.

XOLAP An Essbase multidimensional database that stores only the outline metadata and retrieves all data from a relational database at query time. XOLAP supports aggregate storage databases and applications that contain duplicate member names.

Y axis scale Range of values on Y axis of charts displayed in Investigate Section. For example, use a unique Y axis scale for each chart, the same Y axis scale for all Detail charts, or the same Y axis scale for all charts in the column. Often, using a common Y axis improves your ability to compare charts at a glance.

Zero Administration Software tool that identifies version number of the most up-to-date plug-in on the server.

zoom Sets the magnification of a report. For example, magnify a report to fit whole page, page width, or percentage of magnification based on 100%.

ZoomChart Used to view detailed information by enlarging a chart. Enables you to see detailed numeric information on the metric that is displayed in the chart.

Index

Symbols

#MISSING data
 displaying as empty cells, 126
 in accounts calculations, 197
 representing with Smart Lists, 235
 suppressing in columns, 113
 suppressing in rows, 114
 #MISSING Data Form Label option, 234
 #MISSING Drop-Down Label option, 233
 .RTF files, for custom reports, 241
 .XML files, for custom reports, 241
 @XREF, 235

Numbers

500 error message, 261
 56K dialup connections, using, 37

A

access permissions
 and data forms, 108
 assigning launch access to business rules, 42
 assigning to data forms, 46
 assigning to folders, 46
 assigning to members, 42
 assigning to task lists, 157
 clearing all, 48
 deleting for members or business rules, 44
 elements that can be secured, 40
 exporting, 51
 for shared members, 45
 importing, 48
 inheritance options, 41
 modifying for business rules, 43
 modifying for members, 43
 overview, 20, 39
 precedence, 41
 pushing to Essbase, 58

reporting on, 54
 setting up in Financial Reporting, 63
 types, 40
 access rights, defining for scenarios, 221
 access statistics, 67
 access to application, withdrawing, 168
 accessing
 Performance Management Architect, 36
 Planning, 36
 Workspace, 36
 account annotations
 clearing, 161
 deleting with SQL, 170
 Account dimension, 195
 Account members
 account types, 195
 data type and exchange rate type, 198
 plan types, 199
 saved assumptions, 198
 account types
 and variance reporting, 197
 default time balance property, 196
 described, 195
 examples of use, 195
 summary of, 196
 Actual_365 weighted average time balance, 197
 Actual_Actual weighted average time balance, 196
 ad hoc analysis, in Smart View, 127
 Add Access page, setting the number of users and groups, 169
 Add Smart List Entries tab, 234
 administrators
 and application ownership, 20, 167
 creating applications, 21
 creating business rules, 22
 creating metadata, 20
 designing data forms, 21
 populating applications, 22

- setting up access permissions, [39](#)
- Advanced Settings option
 - described, [165](#)
 - using, [167](#)
- aggregation options, [192](#)
- alias names
 - about, [185](#)
 - displaying on data form rows, [115](#)
 - for attributes, [206](#)
 - for DTS members, [227](#)
- alias tables
 - clearing, [186](#)
 - copying, [187](#)
 - creating, [186](#)
 - deleting, [186](#)
 - editing, [186](#)
 - for attributes, [206](#)
 - for DTS members, [227](#)
 - setting the application default, [187](#)
- Allow Multiple Currencies Per Entity option, [126](#)
- Allow Search When Number of Pages Exceeds option, [125](#)
- Allow Users to Dynamically Add Rows option, [114](#)
- alternate hierarchies for dimensions, [227](#)
- Analytic Services. *See* Essbase
- ancestors, determining for a member, [192](#)
- annotating
 - accounts, [125](#)
 - cells, [127](#)
- application databases
 - about, [67](#)
 - refresh, [228](#)
- application defaults
 - overriding, [166](#)
 - setting, [166](#)
- application information, reviewing Classic, [184](#)
- Application Maintenance Mode, [168](#)
- applications, [279](#). *See also* Sample Application
 - assigning to Shared Service projects, [168](#)
 - backing up, [74](#)
 - creating, [16](#), [65](#)
 - creating and updating with Classic Application Wizard, [177](#)
 - creating, considerations for, [67](#)
 - forcing users off, [168](#)
 - importing data and metadata, [83](#)
 - naming rules, [275](#)
 - opening, [65](#)
 - optimizing performance, [73](#)
 - overview to creating and refreshing, [21](#), [27](#)
 - overview to populating with data, [22](#)
 - overview to setting up, [19](#)
 - ownership, [20](#), [167](#)
 - problem creating, [267](#)
 - problem refreshing, [267](#)
 - refreshing, considerations for, [67](#)
 - registering restored applications with Shared Services, [264](#)
 - reviewing Classic, [184](#)
 - Sample Application, [279](#)
 - selecting, [36](#)
 - Unicode mode, [179](#)
 - unlocking, [65](#)
- Apply Precision option, [127](#)
- Apply Security check box, [42](#), [201](#)
- asset account type
 - example, [195](#)
 - setting, [196](#)
- Assign Application to Project options, [168](#)
- Assigned Access check box, [55](#)
- asymmetric rows and columns, [124](#)
- Attribute Dimension Date Format, setting, [206](#)
- attribute values
 - as members on data forms, [119](#)
 - assigning to members, [207](#)
 - defining, [207](#)
 - deleting, [208](#)
 - effect of changing dimension from sparse to dense, [205](#)
 - hierarchies and aliases for, [228](#)
 - modifying, [208](#)
 - overview, [207](#)
- attributes
 - aliases and alias tables, [206](#)
 - creating, [205](#)
 - data types, [205](#), [206](#)
 - deleting, [206](#)
 - displaying, [116](#)
 - effect of changing dimension from sparse to dense, [205](#)
 - functions, [205](#)
 - hierarchies and aliases for, [228](#)
 - on data forms, [109](#)
 - overview, [205](#)

audit trails
 clearing, [58](#)
 setting up, [56](#)

Automatically Generate ID option, [234](#)

average time balance property, [196](#)

Avg option for exchange rates, [71](#)

B

background processing, setting for jobs, [34](#)

backing up
 application databases, [74](#)
 applications, [74](#)
 before refreshing, [28](#)

balance property, [196](#)

bandwidth, increasing, [265](#)

base currency for Entity members, [195](#)

base currency, setting in Performance Management Architect, [69](#)

BegBalance member. *See* Beginning Balance time period

Beginning Balance time period
 and exchange rates, [215](#)
 and scenarios, [220](#)
 considerations, [228](#)
 described, [71](#)
 editing, [212](#)

BI Publisher Desktop
 customizing reports with, [242](#)
 documentation for, [241](#)
 extracting sample and template files, [242](#)
 installing, [242](#)
 making reports available, [243](#)
 report types, [241](#)
 sample file names, [241](#)
 template names, [243](#)

Boolean data type, [205](#), [206](#)

bottom-up versions
 described, [223](#)
 entering data into, [109](#)

broadcast messaging, [66](#)

budget process
 cycle, [24](#)
 impact of the entities hierarchy, [148](#)
 initializing, [24](#)
 overview, [15](#), [147](#)
 post-planning activities, [26](#)
 starting the review cycle, [24](#)

business rule folders. *See* folders

Business Rules
 overview of, [18](#)
 selecting for Classic applications, [181](#)
 selecting, as the calculation module, [167](#), [182](#)
 user roles, [22](#)

business rules
 assigning launch access to, [42](#)
 Classic View, [175](#)
 configuring background processing, [34](#)
 created in Calculation Manager, [16](#)
 creating, [22](#)
 deleting launch access permissions, [44](#)
 launch upon Load, [129](#)
 launch upon Save, [129](#)
 launch with a utility, [134](#)
 migrating launch access permissions, [62](#)
 modifying launch access permissions, [43](#)
 offline considerations, [145](#)
 overview to launching, [24](#)
 overview to runtime prompts, [130](#)
 overview to using with data forms, [128](#)
 selecting members for runtime prompts, [117](#), [129](#)
 Streamline View, [175](#)
 using, [128](#)
 using with offline data forms, [144](#)
 when launch permissions conflict, [41](#)

C

CalcMgrCmdLineLauncher.cmd utility, [134](#)

Calculate Currencies business rule
 about, [214](#)
 clearing, [129](#)
 importance of launching order, [129](#)
 with multiple currencies, [108](#)

Calculate Data Form business rule
 and Smart View, [214](#)
 creation, [129](#)
 description, [214](#)
 importance of launching order, [129](#)
 Run on Save option, [24](#)

Calculation Manager
 launching business rules with a utility, [134](#)
 overview, [16](#)
 selecting, as the calculation module, [167](#), [182](#)
 setting background processing for business rules, [35](#)

- using with Classic applications, [181](#), [182](#)
- calculation method of currency conversion, [70](#), [219](#)
- Calculation Module option, [167](#), [181](#), [182](#)
- calendars
 - adding years to, [211](#)
 - changing the current year or time period, [211](#)
 - creating summary time periods, [210](#)
 - defining roll ups, [209](#)
 - deleting summary time periods, [210](#)
 - practical limits, [209](#)
 - setting up, [209](#)
 - setting up for Classic applications, [182](#)
- CAPTURE_RTP_ON_JOB_CONSOLE property, [130](#)
- cascading style sheets, customizing, [243](#)
- cell details, clearing, [161](#)
- cell text
 - clearing, [161](#)
 - setting maximum length, [33](#)
 - showing, [127](#)
- cell-level documents, clearing, [161](#)
- cells, read-only, [113](#), [114](#)
- change-tracking, [56](#)
- characters, reserved, [275](#)
- Classic application administration, [177](#)
 - Dimension Editor role, [40](#)
 - Planning Application Creator role, [40](#)
- Classic Application Wizard, [177](#), [184](#), [185](#)
- Classic applications
 - creating and refreshing, [21](#)
 - creating or updating, [178](#), [184](#)
 - deleting, [185](#)
 - importing data and metadata, [83](#)
 - registering, [184](#)
 - reviewing, [184](#)
 - roles required for creating and updating, [40](#)
 - selecting instances (clusters) and data sources, [181](#)
 - selecting the calculation module, [167](#), [182](#)
 - setting up currencies, [183](#)
 - setting up the calendar, [182](#)
 - specifying plan types, [183](#)
- Classic View option , [175](#)
- Clear Cell Details
 - configuring background processing, [34](#)
 - using, [161](#)
- ClosedInputValueBlock variable, [239](#)
- closing Planning, [38](#)
- clusters
 - selecting, [181](#)
 - updating, [38](#), [181](#)
- collapsed member list, setting, [115](#)
- colors, customizing in the UI, [248](#)
- column definitions
 - adding, [116](#)
 - changing, [117](#)
- column properties, setting, [115](#)
- Column Width option, [113](#)
- columns
 - read-only, [113](#)
 - separators for, [113](#)
 - suppress hierarchy, [113](#)
- columns, in reports, [241](#)
- composite data forms, [111](#), [114](#)
- compression filter, [265](#)
- compression settings for offline applications, [33](#)
- conflicting changes to data, [259](#)
- connections, checking, [180](#)
- Consolidation operator, considerations, [228](#)
- context menus, [128](#)
- Copy Data
 - configuring background processing, [34](#)
 - using, [160](#)
- Copy Document Link feature, [156](#)
- Copy Version, [259](#)
- copying
 - data, [160](#)
 - supporting detail, [160](#)
- creating applications, considerations for, [21](#)
- creating databases, considerations for, [21](#), [68](#), [229](#)
- Cross Dimension runtime prompts
 - selecting members, [117](#)
 - with Hide Prompt option, [132](#)
 - with Use Members on Data Form option, [132](#)
- CrossRef(accountName) function, [240](#)
- CrossRef(accountName, prefix) function, [240](#)
- CSS files, customizing, [243](#)
- CubeRefresh utility, [228](#)
- cubes. *See* databases
- currencies
 - calculating conversions, [70](#), [219](#)
 - calculating mixed currencies, [214](#)
 - considerations for currency conversion calc scripts and data storage, [228](#)
 - creating, [216](#)

- data forms, [108](#)
 - deleting, [218](#)
 - editing, [217](#)
 - enabling multiple, [213](#)
 - managing, [69](#)
 - reporting, [215](#)
 - setting up for Classic applications, [183](#)
 - viewing information about, [216](#)
 - currency conversion calc script
 - and Smart View, [214](#)
 - considerations for, [228](#)
 - considerations for data storage, [228](#)
 - failed, [260](#)
 - optimizing performance, [72](#)
 - too long, [257](#)
 - Currency option, [71](#)
 - Current Application Defaults option
 - described, [165](#)
 - using, [166](#)
 - custom attribute display, enabling, [201](#)
 - custom tools, setting, [169](#)
 - customCellEnterPost, [270](#)
 - customCellEnterPre, [269](#)
 - customCellValidatePost, [272](#)
 - customCellValidatePre, [271](#)
 - customizing reports
 - data form definitions, [243](#)
 - data forms, [243](#)
 - extracting sample and template files, [242](#)
 - file names, [243](#)
 - making available, [243](#)
 - planning unit annotations, [243](#)
 - report types, [241](#)
 - sample and template files, [241](#)
 - task lists, [243](#)
 - template names, [243](#)
 - using Microsoft Word, [241](#)
 - using Oracle BI Publisher Desktop, [241](#)
 - customizing text
 - about, [248](#)
 - in non-Latin languages, [249](#)
 - customizing the UI
 - data spreading patterns, examples, [254](#)
 - data spreading patterns, updating, [253](#)
 - of data forms, [269](#)
 - of row and column members, [247](#)
 - of the Tools page, [243](#)
 - text, color, and images, [248](#)
 - with a cascading style sheet, [243](#)
 - customOnLoad, [273](#)
 - CYTD(memberName) function, [240](#)
 - CYTD(memberName, calTpIndexName, fiscalTpIndexName) function, [240](#)
- ## D
- D-T-D, Day-to-date, [226](#)
 - data
 - copying, [160](#)
 - drilling through to source, [108](#)
 - importing with Outline Load, [83](#)
 - invalid value error, [262](#)
 - load files, [103](#)
 - loading, [78](#), [80](#), [105](#)
 - loading into Sample Application, [284](#)
 - read-only, [114](#)
 - suppress missing in columns, [113](#)
 - suppressing missing in rows, [114](#)
 - data columns, in reports, [241](#)
 - data entry forms. *See* data forms
 - data files
 - loading with, [105](#)
 - sample load file, [106](#)
 - data form definitions, customizing reports for, [243](#)
 - data form folders. *See* folders
 - Data Form menu item
 - properties, [175](#)
 - using, [175](#)
 - data forms
 - about, [107](#)
 - access permissions, assigning, [46](#)
 - access permissions, described, [108](#)
 - and versions, [109](#)
 - asymmetric rows and columns, [124](#)
 - attribute values as members, [119](#)
 - attributes, [109](#)
 - business rules for, [128](#)
 - composite, [111](#), [114](#)
 - creating, [110](#)
 - currencies, [108](#)
 - custom attribute display, [116](#)
 - customizing reports for, [243](#)
 - customizing with JavaScript, [269](#)
 - deleting dynamically added rows, [117](#)
 - designing for drill-through, [108](#)

- designing with multiple currencies, 108
- documents in cells, 127
- enabling grid spread, 126
- enabling mass allocation, 126
- exporting definitions, 143
- failure to open, troubleshooting, 260
- hiding, 127
- hiding dimensions, 116
- importing data form definitions, 139
- in Smart View, 127
- large, improving performance, 266
- layout, 112
- members, 117
- menus, associating with, 128
- message for rows with no data, 127
- moving, 139
- offline considerations, 144
- optimizing calculations, 109
- options, 125
- overview to setting up, 21
- page axis, 110
- plan types, 107
- point of view, 110
- previewing design, 136
- printing form definitions, 137
- printing options, 127
- properties, 111
- read-only, 113, 114, 127
- rows and columns, 110
- searching for, 138
- segments, 112
- selecting, 111
- separators, 113, 114
- shared members, 109
- Data Integration Management Adapter for Planning, 105
- Data Load Cube Name, described, 80
- data sources
 - checking connections, 180
 - creating, 179
 - deleting, 181
 - editing, 180
 - selecting for Classic applications, 181
 - working with in Classic applications, 178
- Data Storage property
 - about, 193
 - considerations, 228
 - considerations for currency conversion calc scripts, 228
 - setting, 201
- data types, described, 205, 206
- data types, text, date, Boolean, and numeric, 205
- databases, 67
 - creating and refreshing, 68, 229
 - Essbase, 67
 - unlocking, 260
- Date data type, 205, 206
- DB2. *See* IBM DB2
- decimal places on data forms, 113
- defaults
 - about setting, 165
 - alias table, 187
 - overriding with personal preferences, 166
 - setting for applications, 166
 - setting system, 166
- DeleteSharedDescendants utility, 231
- deleting
 - account annotations with SQL, 170
 - applications, 185
 - cell details, 161
 - supporting detail with SQL, 170
 - users, cleaning database records, 63
- dense dimensions
 - assigning, 73
 - changing to sparse, 205
 - described, 188
- Dependency check box, 153, 155
- deploying Performance Management Architect
 - applications to create Planning applications, 16
- deployment scenario for Planning, 19
- deprovisioning users, cleaning database records, 63
- dialup connections, 37
- DIM, overview of, 105
- dimension hierarchy
 - expanding and collapsing, 189
 - finding members in, 190
 - moving members, 190
 - navigating, 189
 - setting the number of members displayed, 190
 - sorting members, 190
- Dimension(dimtag) function, 239
- dimensions, 192
 - Account, 195
 - aggregation options, 192

- alternate hierarchy for, [227](#)
 - collapsing, [189](#)
 - creating and updating, [16](#)
 - dense, [188](#)
 - Entity, [194](#)
 - expanding, [189](#)
 - finding, [190](#)
 - hiding, [116](#)
 - hierarchies, [188](#)
 - overview, [187](#)
 - relationship to members, [188](#)
 - sparse, [188](#)
 - storage options, [193](#)
 - user-defined, [199](#)
 - Display Member Formula option, [113](#), [114](#), [125](#)
 - Display Missing Values as Blank option, [126](#)
 - Display Order option for Smart Lists, [233](#)
 - Display Users' Full Names option
 - and importing access permissions, [51](#)
 - setting, [167](#)
 - Divide or Multiply option for exchange rates, [71](#)
 - documents, adding to cells in data forms, [127](#)
 - down arrows in data cells, [232](#)
 - drawCustomButtons, [273](#)
 - drilling through to source data, [108](#)
 - driver dimensions, [105](#)
 - Driver Member, described, [80](#)
 - DTS members
 - alias tables and alias names, [227](#)
 - avoiding time balance properties with, [227](#)
 - enabling, [227](#)
 - generations, [227](#)
 - predefined, [226](#)
 - renaming Year and Period, [227](#)
 - setting up, [226](#)
 - Due Date check box, [152](#), [154](#)
 - Dynamic Calc and Store option, setting, [193](#)
 - Dynamic Calc members
 - about, [193](#)
 - and currency conversion calc scripts, [72](#)
 - database refresh and member formulas, [230](#)
 - in currency conversion calc script, [260](#)
 - setting, [193](#)
 - dynamic calc versus dynamic calc and store, [193](#)
 - Dynamic Time Series (DTS)
 - alias tables and alias names, [227](#)
 - avoiding time balance properties with, [227](#)
 - enabling, [227](#)
 - generations, [227](#)
 - predefined, [226](#)
 - setting up, [226](#)
 - dynamic user variables, enabling, [125](#)
 - dynamically adding rows
 - about, [124](#)
 - deleting after users add them, [117](#)
 - enabling, [114](#)
- ## E
- e-mail notifications
 - broadcasting, [66](#)
 - setting up, [148](#)
 - Edit Smart List Entries tab, [234](#)
 - Edit Smart List Properties tab, [233](#)
 - Effective Access check box, [56](#)
 - Enable cell-level document option, [127](#)
 - Enable custom attribute display option, [201](#)
 - Enable Dynamic User Variables option, [125](#)
 - Enable for Smart Slice option, [127](#)
 - Enable Grid Spread option, [126](#)
 - Enable Mass Allocate option, [126](#)
 - End option for exchange rates, [71](#)
 - Entity dimension, [194](#)
 - Entity members
 - alias table, [201](#)
 - and plan types, [199](#)
 - base currency, [195](#)
 - EPMA. *See* Performance Management Architect
 - Equal attribute function, [118](#)
 - equity account type, example, [195](#)
 - ERP Integrator, drilling through to source data, [108](#)
 - error messages
 - for formula expressions, [240](#)
 - troubleshooting, [257](#)
 - Essbase
 - and access permissions, [58](#)
 - closing, [38](#)
 - connecting to multiple servers, [29](#)
 - considerations for, [28](#)
 - multidimensional databases, [67](#)
 - naming restrictions, [275](#)
 - overview of, [17](#)
 - starting, [28](#)
 - working with partitions, [230](#)
 - Essbase Write Access role, [59](#)

Essbase.sec file, [69](#)

examples

- access permissions, [45](#)
- asymmetric rows and columns, [124](#)
- broadcast messaging, [66](#)
- customizing reports, [243](#)
- data precision, [126](#)
- deleting account annotations, [170](#)
- deleting supporting detail associated with a scenario, [170](#)
- importing data form definitions, [142](#)
- loading data, [106](#)
- refreshing databases, [230](#)
- runtime prompts, [130](#)

exchange rate tables

- defining for scenarios, [220](#)
- Fill Year and Fill Table options, [71](#)
- overview of, [69](#), [219](#)
- setting up, [71](#)
- spreading values, [71](#)

exchange rates

- Fill Year and Fill Table options, [71](#)
- setting options for, [71](#)
- setting type, [69](#)
- setting up, [69](#), [218](#)
- spreading values, [71](#)
- types, [215](#)
- types for Account members, [198](#)

expanded member list, setting for rows, [115](#)

expense account type, [196](#)

- and variance reporting, [197](#)
- examples, [195](#)

exporting

- access permissions, [51](#)
- data forms, [143](#)
- task lists, [159](#)

ExportSecurity utility, [51](#)

expressions. *See* formula expressions

F

Failed to execute error message, [240](#)

Favorites menu, documentation for, [16](#)

FDM, drilling through to source data, [108](#)

Fill Table and Fill Year, selecting in exchange rate tables, [71](#)

fill time balance property, [196](#)

Financial Reporting

access permissions, [63](#)

and usage statistics, [67](#)

overview of, [18](#)

user roles, [23](#)

Finish tab, for Classic applications, [184](#)

First time balance property, [196](#)

FIX statement, troubleshooting, [260](#)

Flow time balance property, [196](#)

folders

- about assigning access, [46](#)
- assigning access, [47](#)
- creating, [163](#)
- deleting, [164](#)
- moving, [164](#)
- renaming, [165](#)

fonts and font size, in reports, [241](#)

Format Data option, [127](#)

FormDefUtil utility, [143](#)

forms. *See* data forms

formula expressions

- described, [239](#)
- error messages for, [240](#)
- prerequisites for using, [238](#)
- syntax for, [238](#)
- using in member formulas, [238](#)
- using Smart Lists in, [239](#)
- working with, [238](#)

functions. *See* formula expressions

G

generations, setting for DTS members, [227](#)

getCalendarTPIndex() function, [240](#)

getFiscalTPIndex() function, [240](#)

global settings

- about, [165](#)
- specifying, [166](#)

global.css style sheet, [244](#)

Greater attribute function, [118](#)

GreaterOrEqual attribute function, [119](#)

Grid Spread, enabling, [126](#)

Group the Results By option, [55](#)

H

H-T-D, History-to-date, [226](#)

HBRMigrateSecurity.cmd utility, [62](#)

header record fields, [104](#)

headers, in reports, [241](#)
 Hide Prompt option
 interaction with other settings, [131](#)
 setting, [130](#)
 Hide Runtime Prompt design-time property, [131](#)
 hiding data forms, [127](#)
 hiding dimensions, for rows, [116](#)
 hierarchies
 for attributes, [228](#)
 for periods, [228](#)
 for scenarios and versions, [228](#)
 Historical option for exchange rates, [71](#)
 HSP_Audit_Records table, [58](#)
 HSP_InputCurrency member, [69](#)
 Hsp_InputCurrency member, [219](#)
 HSP_InputValue member, [69](#)
 Hsp_InputValue member, [219](#)
 HSP_NOLINK, [29](#), [235](#)
 HSP_Rates dimension, [69](#)
 Hsp_Rates dimension, [219](#)
 HSP_SPREAD_PATTERN table, [253](#), [254](#)
 HSP_UDF, [29](#), [235](#)
 HspCustom.css, [244](#)
 HspCustomImgs_en.template file, [248](#)
 HspCustomMsgs_en.template file, [248](#)
 HSPSYS_DATASOURCE table, [31](#)

I

IBM DB2 database
 database is locked message, [263](#)
 enabling more connections, [263](#)
 unable to refresh, [267](#)
 images
 customizing with the cascading style sheet, [243](#)
 customizing with the HspCustomImgs_en file, [248](#)
 ImportFormDefinition utility, [139](#)
 importing
 access permissions, [48](#)
 Account dimensions, [91](#)
 attribute dimensions, [101](#)
 Currency dimensions, [100](#)
 data, [80](#)
 data form definitions, [139](#), [143](#)
 dimension properties, [89](#)
 Entity dimensions, [96](#)
 exchange rates, [102](#)

 member properties, [89](#)
 metadata, [82](#)
 overview, [78](#)
 Period dimensions, [97](#)
 Scenario dimensions, [99](#)
 task list definitions, [159](#)
 UDAs, [102](#)
 user-defined dimensions, [98](#)
 using load files, [103](#)
 Version dimensions, [100](#)
 with the Outline Load utility, [83](#)
 Year dimensions, [98](#)
 ImportSecurity utility, [48](#)
 Include Supporting Detail option, [127](#)
 inheritance options
 Children option, [41](#)
 Descendant option, [41](#)
 iChildren option, [41](#)
 iDescendant option, [41](#)
 Member option, [41](#)
 instances
 selecting for Classic applications, [181](#)
 updating, [38](#)
 invalid value error, troubleshooting, [262](#)

J

JavaScript, customizing data forms with, [269](#)
 JDBC connection pooling, [32](#)
 JDBC driver, changing, [31](#)

L

label only data storage
 about, [194](#)
 and data entry forms, [194](#)
 setting, [193](#)
 labels
 in menus, [174](#)
 in Smart Lists, [233](#)
 language in the UI, localizing, [248](#)
 leap year, accounting for, [196](#)
 Less attribute function, [119](#)
 LessOrEqual attribute function, [119](#)
 liability account type, example, [195](#)
 Lifecycle Management, [17](#)
 Linking tasks to Planning Web pages, [156](#)
 links to URLs

- from the Tools page, [169](#)
 - on menus, [175](#)
 - load files
 - creating, [103](#)
 - dimension properties, [89](#)
 - loading
 - Account dimensions, [91](#)
 - attribute dimensions, [101](#)
 - Currency dimensions, [100](#)
 - data, [80](#)
 - dimension properties, [89](#)
 - Entity dimensions, [96](#)
 - exchange rates, [102](#)
 - load files, [103](#)
 - member properties, [89](#)
 - metadata, [82](#)
 - overview, [78](#)
 - Period dimensions, [97](#)
 - Scenario dimensions, [99](#)
 - UDAs, [102](#)
 - user-defined dimensions, [98](#)
 - Version dimensions, [100](#)
 - with the Outline Load utility, [83](#)
 - Year dimensions, [98](#)
 - localizing, [248](#)
 - labels, [174](#)
 - UI text, [248](#)
 - locked application, [65](#)
 - locking users out of an application, [168](#)
 - logged-on users, [67](#)
 - logging on
 - to Performance Management Architect, [36](#)
 - to Planning, [36](#)
 - to Workspace, [36](#)
 - unsuccessful try, [268](#)
- M**
- M-T-D, Month-to-date, [226](#)
 - maintenance mode, [168](#)
 - Manage Database errors, [262](#)
 - Manage menu, Workspace documentation for, [16](#)
 - mass allocation, enabling for data forms, [126](#)
 - MAX_CELL_NOTE_SIZE setting, [33](#)
 - MAXAPPLS parameter, [263](#)
 - Member Alias option, [115](#)
 - Member Formula tab, [237](#)
 - member formulas
 - using formula expressions in, [238](#)
 - validating, [237](#)
 - viewing results of validation, [237](#)
 - working with, [237](#)
 - Member Name option, [115](#)
 - member properties, viewing from data forms, [204](#)
 - Member Range runtime prompts, selecting members, [117](#)
 - Member runtime prompts
 - with Hide Prompt option, [132](#)
 - with Use Members on Data Form option, [132](#)
 - member selection, [117](#)
 - member selection, setting for runtime prompts, [129](#)
 - members
 - adding, [201](#)
 - alias names, [185](#)
 - assigning access to, [42](#)
 - assigning attribute values to, [207](#)
 - collapsing, [189](#)
 - considerations for refreshing, [28](#)
 - deleting, [203](#)
 - deleting access permissions, [44](#)
 - deleting parents, [204](#)
 - determining ancestors, [192](#)
 - determining where used, [192](#)
 - editing, [201](#)
 - effect of moving on plan types, [190](#)
 - expanding, [189](#)
 - finding, [190](#)
 - importing using Data Integration Management Adapter, [105](#)
 - importing using Oracle Data Integrator, [104](#)
 - importing with Outline Load, [83](#)
 - modifying access permissions, [43](#)
 - moving within the dimension hierarchy, [190](#)
 - names, displaying on data form, [115](#)
 - of the Entity dimension, [199](#)
 - selecting, [117](#)
 - sorting, [190](#)
 - viewing properties from data forms, [204](#)
 - working with shared members, [204](#)
 - Menu Header menu item, [175](#)
 - menu items
 - Business Rule type, [175](#)
 - creating or changing, [174](#)
 - data forms, [175](#)
 - icons for, [174](#)

- label for, [174](#)
- Menu Header, [175](#)
- types, [174](#)
- URLs, [175](#)
- Workflow type, [175](#)
- menus
 - adding, [248](#)
 - adding items to, [173](#)
 - associating with data forms, [128](#)
 - creating, [173](#)
 - updating, [248](#)
- message announcements
 - creating, [66](#)
 - for scheduled application refresh, [229](#)
- Message for data forms with no data option, [127](#)
- metadata
 - Account dimensions, [91](#)
 - attribute dimensions, [101](#)
 - considerations for refreshing, [28](#)
 - creating load files, [103](#)
 - Currency dimensions, [100](#)
 - dimension properties, [89](#)
 - Entity dimensions, [96](#)
 - exchange rates, [102](#)
 - importing with Outline Load, [83](#)
 - loading, [78](#)
 - overview, [82](#)
 - overview to setting up, [20](#)
 - Period dimensions, [97](#)
 - Scenario dimensions, [99](#)
 - UDAs, [102](#)
 - user-defined dimensions, [98](#)
 - Version dimensions, [100](#)
 - Year dimensions, [98](#)
- Method for exchange rate tables, [71](#)
- Method for exchange rates, [71](#)
- Migrate Identities button, [42](#)
- migrating
 - launch permissions to business rules, [62](#)
 - user identities, [61](#)
- minimizing dimension columns, [191](#)
- missing data in accounts calculations, [197](#)
- multiple alias tables, [185](#)
- multiple currencies
 - and working offline, [144](#)
 - designing data forms with, [108](#)
 - enabling, [213](#)

- Multiply or Divide option for exchange rates, [71](#)

N

- naming restrictions
 - applications, [275](#)
 - Essbase user names, [278](#)
 - for applications, [275](#)
 - for dimensions, members, and aliases, [276](#)
 - in calculation scripts and report scripts, [277](#)
 - in Essbase, [275](#)
 - in formulas and filters, [277](#)
- native2ascii program, [249](#)
- Navigate menu, documentation for, [16](#)
- network bandwidth, increasing, [265](#)
- Never Share option
 - about, [194](#)
 - setting, [193](#)
- non-expense account types, [197](#)
- NotEqual attribute function, [118](#)
- number formatting, [215](#)
- NumberOfPeriodsInYear variable, [239](#)
- NumberOfYears variable, [239](#)
- numeric data type, [205](#), [206](#)

O

- ODI, overview of, [104](#)
- offline
 - business rule considerations, [145](#)
 - considerations for working with, [144](#)
- offline applications, setting compression for, [33](#)
- OFFLINE_COMPRESSION_THRESHOLD setting, [33](#)
- opening large data forms, [266](#)
- OpenInputValueBlock variable, [239](#)
- optimizing performance
 - by application design, [73](#)
 - of calculations on data forms, [109](#)
 - of currency conversion, [72](#)
 - over dial-up connections, [265](#)
 - when importing access permissions, [51](#)
- Oracle BI Publisher Desktop
 - customizing reports with, [242](#)
 - documentation for, [241](#)
 - extracting sample and template files, [242](#)
 - installing, [242](#)
 - making reports available, [243](#)

- report types, [241](#)
 - sample file names, [241](#)
 - template names, [243](#)
 - Oracle Data Integrator Adapter for Planning, [104](#)
 - orientation, in reports, [241](#)
 - Outline Load utility
 - Account dimensions, [91](#)
 - attribute dimensions, [101](#)
 - creating load files, [103](#)
 - Currency dimensions, [100](#)
 - dimension properties, [89](#)
 - Entity dimensions, [96](#)
 - exchange rates, [102](#)
 - load files, [103](#)
 - loading data, [80](#)
 - loading metadata, [80](#), [82](#)
 - member properties, [89](#)
 - overview, [78](#)
 - Period dimensions, [97](#)
 - Scenario dimensions, [99](#)
 - UDAs, [102](#)
 - user-defined dimensions, [98](#)
 - using, [83](#)
 - Version dimensions, [100](#)
 - Year dimensions, [98](#)
 - outline, considerations for updating, [28](#)
 - overview, of Planning, [15](#)
 - owner of the application, assigning, [167](#)
- P**
- P-T-D, renaming Period member when using, [226](#)
 - page axis
 - about, [110](#)
 - defining, [124](#)
 - enabling search facility, [125](#)
 - selecting members, [124](#)
 - page size, in reports, [241](#)
 - partitions, [230](#)
 - PasswordEncryption utility, [37](#)
 - passwords
 - entering to log on, [36](#)
 - suppressing prompts for utilities, [37](#)
 - PDF reports, customizing, [243](#)
 - percentage of page for headers, [241](#)
 - performance
 - optimizing currency conversion, [72](#)
 - optimizing in applications, [73](#)
 - optimizing over dial-up connections, [265](#)
 - when importing access permissions, [51](#)
 - when opening large data forms, [266](#)
 - Performance Management Architect
 - assigning dense and sparse dimensions, [73](#)
 - considerations for creating and refreshing, [28](#)
 - creating and managing applications and dimensions in, [16](#)
 - creating and refreshing application databases, [67](#)
 - described, [16](#)
 - documentation for, [16](#)
 - installing and configuring, [28](#)
 - managing currencies in, [69](#)
 - starting, [36](#)
 - Period(periodName) function, [240](#)
 - period, changing, [211](#)
 - Period-to-date, renaming Period when using, [227](#)
 - periods
 - alternate hierarchies, [228](#)
 - renaming when using P-T-D DTS members, [226](#)
 - plan building overview, [24](#)
 - plan types
 - Account members, [199](#)
 - data forms, [107](#)
 - effect of moving members, [190](#)
 - Entity members, [199](#)
 - specifying for Classic applications, [183](#)
 - planning cycle overview, [24](#)
 - planning units
 - annotations, customizing reports for, [243](#)
 - defined, [147](#)
 - enabling e-mail notification, [148](#)
 - printing annotations, [149](#)
 - starting the review process, [147](#)
 - Planning, tasks performed in Performance Management Architect, [18](#)
 - PlanningFormDefSample.xml, [241](#)
 - PlanningFormDefTemplate.rtf, [241](#)
 - PlanningFormDefTemplate_Custom.rtf, [243](#)
 - PlanningFormSample.xml, [241](#)
 - PlanningFormTemplate.rtf, [241](#)
 - PlanningFormTemplate_Income.rtf, [243](#)
 - PlanningPUAnnotationSample.xml, [242](#)
 - PlanningPUAnnotationTemplate.rtf, [242](#)
 - PlanningPUAnnotationTemplate_Custom.rtf, [243](#)
 - PlanningTaskListSample.xml, [241](#)
 - PlanningTaskListTemplate.rtf, [241](#)

PlanningTaskListTemplate_Custom.rtf, [243](#)

point of view

about, [110](#)

defining, [124](#)

hiding or showing members on, [125](#)

loading data for, [80](#)

selecting members, [124](#)

populating the application, overview, [22](#)

post-planning activities overview, [26](#)

precision, in reports, [241](#)

preferences, setting personal, [166](#)

Preview tab for Smart Lists, [235](#)

previewing data form design, [136](#)

printing

data form definitions, [137](#)

options for data forms, [127](#)

planning unit annotations, [149](#)

reports, [241](#)

process status states

customizing actions, [252](#)

customizing colors, [251](#)

customizing states, [252](#)

overview to customizing, [251](#)

properties

background processing, [34](#)

changing the JDBC driver, [31](#)

JDBC connection pools, [32](#)

MAX_CELL_NOTE_SIZE, [33](#)

offline compression, [33](#)

overview to configuring, [30](#)

Smart View messages, [34](#)

substitution variables retrieval interval, [122](#)

supporting detail cache, [74](#)

system database table, [30](#), [33](#)

ProvisionUsers utility, [60](#)

Q

Q-T-D, Quarter-to-date, [226](#)

R

range of time periods, setting, [210](#)

read-only cells, [114](#)

read-only data forms, [127](#)

Read-only option for data forms, [113](#)

Reference Application. *See* Sample Application

refreshing applications, considerations for, [21](#)

refreshing databases

considerations for, [21](#), [28](#), [68](#), [229](#)

considerations for Classic Planning and Performance Management Architect, [28](#)

problem-solving, [258](#)

registering

Classic applications, [184](#)

restored applications with Shared Services, [264](#)

relational database, optimizing, [30](#)

Remove Non-provisioned Users/Groups button, [63](#)

Repeat Every check box, [153](#), [154](#)

reporting currencies, [215](#)

reporting currency, setting, [69](#)

reports

access permissions, [54](#)

audit trails, [56](#)

customizing, [241](#), [242](#), [243](#)

making available, [243](#)

PDF, customizing, [241](#)

Required Dimension drop-down list, [175](#)

reserved characters, [275](#)

in customized text, [250](#)

in Essbase, [275](#)

naming restrictions, [275](#)

restoring dimension columns, [191](#)

revenue account type

example, [195](#)

summary, [196](#)

roll ups, defining in calendars, [209](#)

row definitions

about dynamically adding, [124](#)

adding, [116](#)

changing, [117](#)

dynamically adding, [114](#)

row properties, setting, [115](#)

Row/Column Layout tab, [112](#)

rows

deleting dynamically added rows, [117](#)

message when no data, [127](#)

separators for, [114](#)

suppress hierarchy, [114](#)

rtf files, for custom reports, [241](#)

Run in Background option, [69](#)

runtime prompts

and View users, [40](#)

capturing values, [130](#)

creating values file, [134](#)

effects of multiple settings, [131](#)
 overview, [130](#)
 selecting members for, [117](#)

S

- S-T-D, Season-to-date, [226](#)
- SampApp_data.zip file, [284](#)
- Sample Application
 - business rules, [281](#)
 - creating, [182](#)
 - creating and initializing, [283](#)
 - data forms, [280](#)
 - database, [283](#)
 - loading data, [284](#)
 - members, [280](#)
 - menus, [283](#)
 - overview, [279](#)
 - substitution variables, [282](#)
 - task lists, [283](#)
 - user variables, [281](#)
- sample files
 - adding fields to reports, [242](#)
 - extracting to customize reports, [242](#)
 - for all report types, [241](#)
 - using to load data, [242](#)
- SampleValidateData.js, [269](#)
- saved assumptions, [198](#)
- scaling, [215](#)
- scenarios
 - alternate hierarchies, [228](#)
 - changing the order, [226](#)
 - copying, [222](#)
 - creating, [221](#)
 - defining access rights, [221](#)
 - defining exchange rate tables, [220](#)
 - defining time periods, [220](#)
 - deleting, [222](#)
 - deleting supporting detail with SQL, [170](#)
 - editing, [221](#)
 - sorting, [225](#)
- scheduling messages, [66](#)
- sddelete.sql file, [170](#)
- searching
 - among pages, [125](#)
 - for members, [190](#)
- SecFile.txt file
 - for exporting access permissions, [51](#)
 - for importing access permissions, [49](#)
- security. *See* access permissions
- security filters
 - generating for all users, [69](#)
 - generating for selected users, [58](#)
 - validating, [67](#)
- segments
 - described, [112](#)
 - read-only, [113](#)
 - suppress hierarchy, for columns, [113](#)
 - suppress hierarchy, for rows, [114](#)
- Select Objects tab, [55](#)
- separators
 - described, [112](#)
 - setting for columns, [113](#)
 - setting for rows, [114](#)
- session timeout, [265](#)
- shading, in reports, [241](#)
- shared members, [204](#)
 - access permissions, [45](#)
 - applying access permissions to, [69](#)
 - creating, [205](#)
 - data forms, [109](#)
 - deleting, [231](#)
 - setting, [193](#)
- Shared option
 - overview, [194](#)
 - setting, [193](#)
- Shared Services Server URL option, [167](#)
- Show Account Annotations option, [127](#)
- Show Ancestors button, [192](#)
- Show Attribute Members option, [127](#)
- Show Cell Text option, [127](#)
- Show Currency Codes option, [127](#)
- Show Inherited From Group check box, [55](#)
- Show Matching Access of Relation check box, [55](#)
- Show Matching Access of Type option, [55](#)
- Show option, described, [165](#)
- Show Separator option, [113](#), [114](#)
- Show the Specified Members on Each Dimensions
 - Page option, [169](#), [190](#)
- Show the Specified Records on Each Assign Access
 - Page option, [169](#)
- Show Usage button, [192](#), [216](#)
- SIDs, updating, [59](#)
- Size-to-Fit column width, [113](#)
- Skip options, [197](#)

- Smart Lists
 - adding or changing entries, [234](#)
 - and data entry, [232](#)
 - previewing, [235](#)
 - representing #MISSING, [235](#)
 - setting properties, [233](#)
 - using in formula expressions, [239](#)
 - working with, [232](#)
 - Smart View
 - ad hoc analysis, [127](#)
 - considerations for using, [144](#)
 - currency conversion calc scripts, [214](#)
 - Enable for Smart Slice option, [127](#)
 - inputting multiple currencies, [214](#)
 - overview, [16](#)
 - terminating for application refresh, [229](#)
 - SMART_VIEW_DISPLAY_WARNING setting, [34](#)
 - sorting members
 - in the Dimension page, [190](#)
 - with a utility, [230](#)
 - SortMember utility, [230](#)
 - source plan type
 - and Accounts, [199](#)
 - effect of moving members, [190](#)
 - sparse dimensions
 - and performance, [73](#)
 - effect of changing to dense, [205](#)
 - overview, [188](#)
 - spreading data
 - enabling for data forms, [126](#)
 - examples, [254](#)
 - in exchange rate tables, [71](#)
 - spread patterns, [253](#)
 - Spreadsheet Add-in. *See* Smart View
 - spreadsheets. *See* Smart View
 - SQL database, optimizing, [30](#)
 - Start Expanded option, [115](#)
 - starting Planning, [36](#)
 - statistics on usage, [67](#)
 - status checks, setting for jobs, [34](#)
 - storage options, [193](#)
 - Store option
 - about, [194](#)
 - setting, [193](#)
 - Streamline View option, [175](#)
 - strings, customizing, [248](#)
 - style sheets, customizing, [243](#)
 - substitution variables
 - enabling display in runtime prompts, [167](#)
 - selecting as members, [122](#)
 - summary time periods
 - changing the range, [210](#)
 - creating, [210](#)
 - deleting, [210](#)
 - supporting detail
 - associated with a scenario, deleting with SQL, [170](#)
 - clearing, [161](#)
 - copying, [160](#)
 - detection cache
 - allocating memory for, [74](#)
 - percentage used, [67](#)
 - SUPPORTING_DETAIL_CACHE_SIZE setting, [74](#)
 - Suppress Hierarchy option
 - for columns, [113](#)
 - for rows, [114](#)
 - Suppress Missing Data option
 - for columns, [113](#)
 - for rows, [114](#)
 - suppressed hierarchy, described, [112](#)
 - suppressing password prompts, [37](#)
 - system settings, [33](#), [34](#), [166](#)
- ## T
- target versions
 - description, [223](#)
 - setting up, [23](#)
 - task list folders
 - creating, [150](#)
 - deleting, [151](#)
 - moving, [150](#)
 - renaming, [150](#)
 - task lists
 - assigning access, [157](#)
 - changing access, [158](#)
 - copy document link, [156](#)
 - creating, [149](#)
 - customizing reports for, [243](#)
 - deleting, [156](#)
 - exporting definitions, [159](#)
 - moving, [156](#)
 - overview, [149](#)
 - removing access, [158](#)
 - renaming, [150](#)
 - reordering, [155](#), [156](#)

- TaskListDefUtil utility, 159
- tasks
 - alert messages, 152
 - copying, 155
 - defining, 152
 - deleting, 156
 - dependencies, 153, 155
 - due date messages, 152
 - moving, 155
 - overview, 149
 - performed in Planning and Performance Management Architect, 18
- template files
 - customizing in Word, 242
 - extracting to customize reports, 242
 - for all report types, 241
 - naming, 243
 - opening, 242
 - previewing, 243
 - saving, 243
- terminating applications for refresh, 229
- text
 - customizing in the UI, 248
 - overview to customizing, 248
- Text data type, 205, 206
- threshold for offline compression, setting, 33
- time balance calculations, 197
- Time Balance property, 196
- time periods
 - changing, 211
 - defining for scenarios, 220
 - practical limit, 209
- Time Periods dimension, about, 209
- timeout for sessions, 265
- Tools menu, documentation for, 16
- Tools page, customizing, 169, 243
- tools, setting custom, 169
- tracking changes, 56
- triangulation currency, setting, 69
- triangulation currency, setting in Performance Management Architect, 70
- triangulation overview, 70, 219
- troubleshooting, 257
- two pass calculation, considerations for, 228
- changing, 236
- creating, 236
- deleting, 236
- HSP_NOLINK, 29, 235
- HSP_UDF, 29, 235
- selecting, 236
- with @XREF, 235
- working with, 235
- Unicode mode, 179
- Unicode Mode option, 179, 180
- unlocking applications, 65, 260
- UpdateNativeDir utility, 59
- UpdateUsers utility, 59
- URL menu item
 - properties, 175
 - setting, 175
- URL tasks, single sign-on, 153
- URL, for Workspace, 36
- URLs
 - in data form cells, 127
 - setting up, 169
- usage scenario, 18
- usage statistics, 67
- Use Last Value property, 131
- Use Members on Data Form option
 - interaction with other settings, 131
 - setting, 130
- user name, 36
- user names, displaying full names, 167
- user variables
 - associating with a form, 123
 - creating, 137
 - deleting, 137
 - enabling dynamic variables, 125
 - overview, 136
- users
 - controlling access permissions, 39
 - deleting or deprovisioning, 63
 - overview to setting up, 20
 - synchronizing with ProvisionUsers, 60
 - synchronizing with UpdateUsers.cmd, 59
- users and groups on Assign Access page, setting the number of, 169
- utilities
 - BroadcastMessage, 66
 - CalcMgrCmdLineLauncher, 134
 - CubeRefresh, 228

U
UDAs

DeleteSharedDescendant, [231](#)
 deleting shared descendants, [231](#)
 ExportSecurity, [51](#)
 for application database refresh, [228](#)
 FormDefUtil, [143](#)
 HBRMigrateSecurity, [62](#)
 HspUnlockApp, [65](#)
 ImportFormDefinition, [139](#)
 importing and exporting data form definitions, [143](#)
 importing and exporting task list definitions, [159](#)
 importing data form definitions, [139](#)
 importing metadata and data, [78](#)
 ImportSecurity, [48](#)
 Outline Load, [78](#)
 PasswordEncryption, [37](#)
 ProvisionUsers, [60](#)
 sorting members, [230](#)
 SortMember, [230](#)
 TaskListDefUtil, [159](#)
 text file format, [37](#)
 unlocking applications, [66](#)
 UpdateNativeDir, [59](#)
 UpdateUsers, [59](#)

V

Valid For plan type setting
 and Account members, [199](#)
 and custom dimensions, [200](#)
 and Entity members, [199](#)
 effect of moving members, [190](#)
 specifying, [200](#)
 Validate Member Formula option, [237](#)
 ValidateData.js, [269](#)
 validateForm, [273](#)
 variance reporting and account type, [196](#)
 versions
 alternate hierarchies, [228](#)
 and data forms, [109](#)
 bottom-up, [109](#)
 changing the order, [226](#)
 copying, [259](#)
 creating, [224](#)
 deleting, [225](#)
 displaying, [225](#)
 editing, [224](#)
 error when copying, [259](#)

setting up, [225](#)
 setting up bottom-up versions, [223](#)
 setting up target versions, [223](#)
 sorting, [225](#)
 Visible option, for data forms, [125](#)

W

W-T-D, Week-to-date, [226](#)
 Warn If Data Form Larger Than Cells Specified option, [267](#)
 Web Analysis, overview, [18](#)
 Web client
 about, [16](#)
 starting, [36](#)
 web.xml file, [265](#)
 Weighted Average time balance properties, [196](#)
 Workflow menu item, [175](#)
 Workspace
 documentation for, [16](#)
 logging on, [36](#)
 tasks performed in, [16](#)
 URL for, [36](#)

X

XML files, for custom reports, [241](#)

Y

Y-T-D, renaming Year member when using, [226](#)
 years
 adding to calendar, [211](#)
 changing current, [211](#)
 maximum, [209](#)
 practical limit, [209](#)
 renaming when using Y-T-D DTS members, [226](#)

Z

zeros in accounts calculations, [197](#)

A B C D E F G H I J L M N O P Q R S T U V W X Y Z