Oracle® Hyperion Profitability and Cost Management, Fusion Edition

Administrator's Guide

RELEASE 11.1.2.1

ORACLE®

ENTERPRISE PERFORMANCE
MANAGEMENT SYSTEM

Profitability and Cost Management Administrator's Guide, 11.1.2.1

Authors: EPM Information Development Team

Contents

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.

Access to Oracle Support for Hearing-Impaired Customers

Oracle customers have access to electronic support through My Oracle Support or by calling Oracle Support at 1.800.223.1711. Hearing-impaired customers in the U.S. who wish to speak to an Oracle Support representative may use a telecommunications relay service (TRS). Information about the TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html/, and a list of telephone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html. International hearing-impaired customers should use the TRS at +1.605.224.1837. An Oracle Support engineer will respond to technical issues according to the standard service request process.

1

Overview

In This Chapter

Oracle Hyperion Profitability and Cost Management, Fusion Edition is an analytical application that is accessed from Oracle Enterprise Performance Management Workspace, Fusion Edition, and used to accurately measure, allocate, and manage costs and revenues; compute profitability for business segments; and measure profitability by using cost decomposition, consumption-based costing, and scenario playing.

Profitability and Cost Management is an integral part of EPM Workspace, using various components to build and manage its applications:

- Use EPM Workspace to access Profitability and Cost Management, and manage other components to build the application, control security, and product reports.
- Use Oracle's Hyperion® Shared Services to create and manage user accounts, including the definition of security roles to determine which models users can access.
- Use Oracle Hyperion EPM Architect, Fusion Edition to build and maintain both Shared and Local dimensions and members for use in Profitability and Cost Management. Through the Dimension Library, you can build the application using common dimensions and members that already exist in other applications, such as Oracle Hyperion Planning, Fusion Edition.

  Using Performance Management Architect, dimensions can exist in an Application either as Shared or Local:

  - Shared Dimensions reside in the Shared Library in Performance Management Architect, and can be used by multiple applications.
  - Local Dimensions are detached, independent dimensions that only exist in one application, such as Profitability and Cost Management. These dimensions are used only for the application for which they have been created.
- Use Oracle Essbase to create the application outline and to store and execute calculation scripts.

- Use Oracle Hyperion Enterprise Performance Management System Lifecycle Management to migrate an application, multidimensional database, repository, or individual artifacts across product environments and operating systems.
- Create reports of the calculated results, using Oracle's Hyperion Reporting and Analysis, Oracle Hyperion Financial Reporting, Fusion Edition, Oracle's Hyperion® Web Analysis, or third party products, such as Microsoft Excel.

Accessibility in Profitability and Cost Management

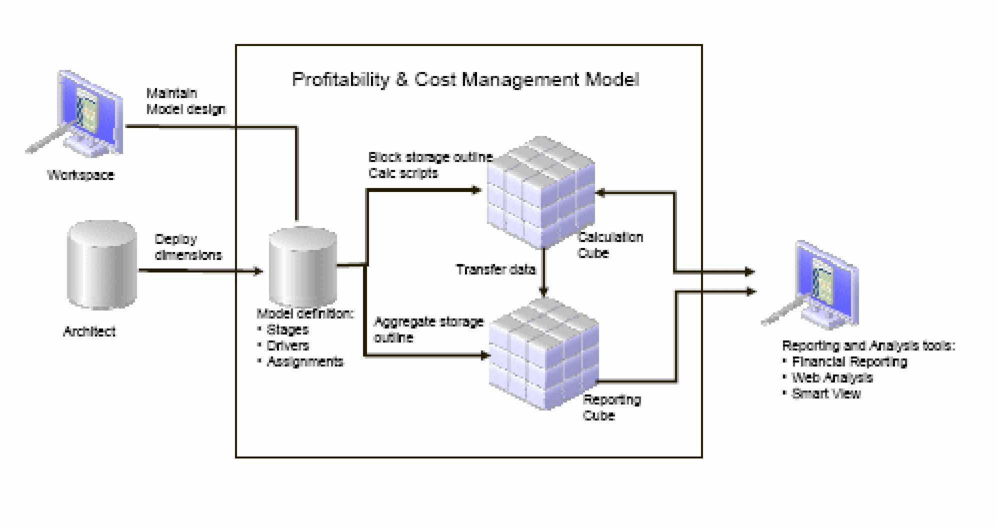
Profitability and Cost Management provides keyboard shortcuts for the main functions.

The Accessibility features are documented in the Accessibility Appendix in the *Oracle Hyperion Profitability and Cost Management User's Guide*. These features are unique to Profitability and Cost Management.

Understanding the Product Architecture

Accessed through EPM Workspace, Profitability and Cost Management is an analytical application that resides on top of Essbase. The application enables business users to model their business for profitability and cost management, and use that model information to create Essbase databases that can be fine-tuned for profitability and cost analysis without having to understand a scripting language.

Figure 1 Profitability and Cost Management Product Architecture



Profitability and Cost Management leverages Performance Management Architect and Shared Services for the centralized management of application metadata and security.

Application administrators create the Profitability and Cost Management dimensions using Performance Management Architect. User access is managed centrally with Shared Services. When the dimension metadata is ready, it is deployed to a Profitability and Cost Management application, or model. The dimensions in Performance Management Architect can be shared by multiple models.

The model design contains the information needed to generate the Essbase outline and calculation script required by the Essbase component of the model. Each model requires access to the following databases:

- Relational database that stores the model design, including the dimension metadata deployed from Performance Management Architect
- Essbase database that includes a Calculation database (BSO) and a Reporting database (ASO).

Note: Only one database is required to store multiple models.

Results from the Reporting database may be viewed in reporting and analysis tools, such as Financial Reporting, Web Analysis, and Oracle Hyperion Smart View for Office, Fusion Edition.

Working with Databases

For Profitability and Cost Management, you require both a relational database and an Essbase database. Details about these databases and connection information are specified on the System Information tab of the Model Summary in the application. For details, see the *Oracle Hyperion Profitability and Cost Management User's Guide*.

When you deploy an application, Performance Management Architect populates tables with appropriate values in your Profitability and Cost Management relational database. The Essbase application is created during model deployment in Profitability and Cost Management. Each database stores specific information, as shown in Table 1.

Table 1 Profitability and Cost Management Databases

Database Type	Information Type Stored
Essbase	● Dimension Hierarchies ● Cost Data ● Revenue Data ● Driver Data ● Currency Rates ● Calculation Scripts

Database Type	Information Type Stored
Product Relational Database	● Dimensions ● Stage Definitions ● POV Definitions ● Driver Definitions ● Driver Selections ● Assignments ● Assignment Rules ● Assignment Rules Selections ● Model Preferences

Profitability and Cost Management uses the information from the driver definitions, driver selections and assignments defined in the relational database to create calculation scripts in the Essbase application. The calculation scripts perform the required data allocations.

Additional Product Components

Profitability and Cost Management is used to create models and perform allocations. The power and flexibility of Profitability and Cost Management are extended through the use of the products shown in Table 2.

Table 2 Profitability and Cost Management Product Components

Product	Description
Essbase	Store and calculate Profitability and Cost Management application data.
Oracle Essbase Administration Services	Interface to Essbase server, using the Essbase Administration Services Console Used to design, develop, maintain, and manage multiple Essbase applications and databases.
Performance Management Architect	Manage dimensions and applications
EPM Workspace	Navigate to Profitability and Cost Management and other products, and manage applications
Shared Services	Provision users from external systems for Profitability and Cost Management Migrate Profitability and Cost Management applications
Oracle Hyperion Smart View for Office, Fusion Edition	Enter and report on Profitability and Cost Management data in Microsoft Excel spreadsheets
Financial Reporting	Create reports and charts for Web or print distribution
Web Analysis	Use a Web client to create ad hoc reports and charts for analysis

Administrative Tasks

The Profitability and Cost Management Administrator or *admin* role enables you to perform these tasks:

- Create and manage user accounts using Shared Services.
- Provision users for Shared Services authentication.
- Manage the Shared Library and Profitability and Cost Management dimensions and members in Performance Management Architect.
- Generate multidimensional Essbase databases.
- Create, update, and delete model stages, drivers and points of view (POVs).
- Create, update, and delete driver selections, assignments, assignment rules and assignment rules' selections.
- Create, update, and delete applications and application preferences.
- View and modify model data.
- View trace allocations.
- Back up and restore Profitability and Cost Management model components.
- Transfer applications from one environment to another using the Lifecycle Management Utility. Promote data from one environment, such as development or testing, to another environment, such as production.
- Monitor changes made to business objects.

The Profitability and Cost Management Administrator also manages the model metadata and data through the EPM Workspace. See Chapter 4, "Managing Applications and Dimensions Using Performance Management Architect".

Accessing Profitability and Cost Management After Installation

Profitability and Cost Management is an integral part of EPM Workspace, and uses common applications to manage the application and security. After installation, you must perform a number of steps to create the first Profitability and Cost Management application. After the application is created, you need to import data into Profitability and Cost Management.

➤ To create the first Profitability and Cost Management application after installation:

1 **Ensure the following products are installed, configured and running:**

    - EPM Workspace
    - Shared Services
    - Performance Management Architect
    - Essbase

- Profitability and Cost Management

Note: This list represents the minimum installation required to use Profitability and Cost Management; however, you may install additional products at any time.

For complete installation instructions, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide* and the *Oracle Hyperion Enterprise Performance Management System Installation Start Here*.

2. From the EPM Workspace main menu, select **Navigate**, then **Administer**, and then **Application Library.**

3. From the Application Library, select **File**, then **New**, and then **Application Wizard** to create a new Profitability and Cost Management application. For detailed instructions on creating a new application, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

4. On the EPM Workspacemain menu, select **Navigate**, then **Administer**, and then **Dimension Library** to access Performance Management Architect to create the dimensions required for the first application.

   For detailed instructions on creating dimensions, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*. For information on required dimensions, see Chapter 4, "Managing Applications and Dimensions Using Performance Management Architect".

5. From the Application Library, right-click the new application name and select **Validate.**

   The Validation task is submitted to verify the metadata. Click the link on the job status message to view the Job Console for the validation task and view warnings or errors in the log file under **Attachments**. You must correct errors before deploying the application.

6. From the Application Library, right-click the new application name and select **Deploy.**

   The application name and selected dimensions are forwarded to the Profitability and Cost Management server. See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

7. On the EPM Workspace main menu, select **Navigate**, then **Applications**, then **Profitability**, and then select the new application.

   Note: If the newly created application is not found, select Refresh, and then repeat step 7.

8. From **Task Areas**, select **Manage Models**, then **Import Staging Tables** to import the data you require to begin building a model. See Chapter 5, "Importing Data into Profitability and Cost Management".

Launching Profitability and Cost Management

Profitability and Cost Management can only be accessed through EPM Workspace.

➤ To access Profitability and Cost Management:

1. Ensure the following applications have been configured, and are running:

- EPM Workspace
- Shared Services
- Performance Management Architect
- Essbase
- Profitability and Cost Management

See the *Hyperion EPM System Installation and Configuration Guide* and the *Oracle Hyperion Enterprise Performance Management System Installation Start Here* for instructions.

2 **In your Web browser, access the EPM Workspace Web page.**

By default, the URL is `http://`*`server name`*`:19000/workspace/` where the server name is the Oracle HTTP Server (OHS) server name.

3 **Enter the EPM Workspace user name and password.**

Note: Both the user name and password are case-sensitive.

4 **Click Log On.**

The main EPM Workspace page is displayed.

5 **Select Navigate, then Applications, then Profitability and then select the application you want to view.**

2 Managing Security and Authorizing Users

In This Chapter

About User Setup and Provisioning

Before working with Profitability and Cost Management, the Administrator must set up users and groups, and assign the appropriate security role to each one. The authorization provided for each security role determines which functions and data a user or group may access. During configuration, select Shared Services as the authentication mode, as described in the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

Two guides are available for managing security and user provisioning:

- Use the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide* for technical information regarding security:
  - SSL (one-way, two-way, SSL OffLoading, SSL Termination)
  - Single Sign-On
  - Default EPM System SSO
  - Security Agents
  - Custom Login
  - Custom Authentication Modules
  - Guidelines for securing the EPM System
- Use the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide* for information on how to set up and manage user provisioning:
  - Oracle's Hyperion® Shared Services Console
  - User Directories
  - Applications and Application Groups
  - Delegated User Management

- ❍ Managing Native Directory
- ❍ Managing Provisioning
- ❍ Provisioning the EPM System

You create, maintain and provision users and groups for Profitability and Cost Management through EPM Workspace. The provisioning process requires you to have both Shared Services and Profitability and Cost Management configured and running. External authentication ensures that the applications communicate seamlessly to provision users easily and accurately.

The following steps provide an overview of the process to set up and provision users and groups:

1. From EPM Workspace, select **Navigate**, then **Administer**, and then **Shared Services Console** to access the Shared Services screens.

   Note: The first time that you log on, an administrator (*admin*) user is automatically created for your product.

2. Set the user assigned to the *admin* role to the Provisioning Manager role. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.
3. Create users. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.
4. Provision the users with the appropriate security role, and access to the required projects and applications.

   Caution! If the user needs to access EPM Workspace to perform tasks outside of Profitability and Cost Management, you must also provision the selected user with the EPM Workspace role.

5. Create groups, as required. You can provision groups within groups. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.
6. Log on to EPM Workspace, and open Profitability and Cost Management.
7. Verify that the user can log on to Profitability and Cost Management, and is able to see the provisioned projects and applications.

For detailed instructions on setting up and provisioning users and groups, see the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.

Assigning Security Roles

In Profitability and Cost Management, each user ID is assigned a security role:

- Administrator (*admin* is the default security role when you log on to Shared Services)
- Power User
- Interactive User
- View User

The assigned security role determines the level of access or privileges available for that user. A user can be granted multiple roles, and at the time an action is initiated, the permission is checked for that specific action.

Note: At least one user must be manually assigned the Provisioning Manager role in the Shared Services Console. This Provisioning Manager role enables that user to assign security roles to other users for the application. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*

The security roles in Table 3 are specific to Profitability and Cost Management. For a complete description of all security roles, see the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.

A user must exist and have an assigned security role before you can assign the user to a group. When an access level is assigned to a group of users, similar security access is granted to all members of that group. Depending on the access requirements for a particular user, the assigned security may be modified to attach a wider or narrower access. For example, a View User assigned to a group that has Power User security authorization assumes that higher level of security.

If a user must initiate and monitor taskflows, additional Shared Services roles are required, as shown on Table 3, "Profitability and Cost Management Security Roles," on page 20.

Caution! If the user requires access to other products, such as an Application Creator in EPM Workspace or Dimension Editor in Performance Management Architect, those additional security roles must be assigned separately. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.

See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide* for detailed instructions.

Table 3 Profitability and Cost Management Security Roles

Security Role	Type of Role	Description
Administrator (*admin*)	Power	● Create and maintain user accounts and security roles, and provision users, using Shared Services ● Generate Essbase databases ● Set up and maintain application preferences ● Build the model database using Performance Management Architect to select the common dimensions and members ● Create and maintain elements within the model, such as stages, drivers, POVs, driver selections, assignments, and application preferences ● Perform POV Copy, calculation, validation, data entry, and trace allocations ● Deploy to Essbase and generate calculation scripts ● Import and export data ● Use the Lifecycle Management Utility to promote data from one environment, such as development or testing, to another environment, such as production. ● Back up and restore Profitability and Cost Management model components. ● Monitor changes made to business objects. **Note:** The Power User does not necessarily require specific security roles to perform tasks. For example, if a Power User runs a calculation from the Calculate screen, this action creates and executes a taskflow behind the scenes. The Power User does not require the Manage Taskflow role to perform this task, unless the Power User wants to access this task directly from the Manage Taskflows task.
Power User	Power	● Create and maintain elements within the model, such as stages, drivers, POVs, driver selections, assignments, and application preferences. ● Perform POV Copy, calculation, validation, data entry and trace allocations. ● Deploy to Essbase and generate calculation scripts. ● Import and export data
Interactive User	Interactive	● View all modelling screens ● View and modify data in the Data Entry screen ● View Trace Allocations
View User	Interactive	View only access for these functions: ● Trace Allocations ● Application Preferences ● Model Stages, Drivers and POVs
Manage Taskflows	Shared Services Role	Required to create and edit taskflows. For more information, see the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.
Run Taskflows	Shared Services Role	Required to enable users to only run and view taskflows. Users with this role cannot create or edit taskflows. For more information, see the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.

Auditing Changes in Profitability and Cost Management

You can monitor activity and changes in your application using the Audit feature in the Shared Services Console, and then generate audit reports detailing the results.

There are three types of audit reports available:

- Security reports
- Artifact Reports
- Config Reports

The audit reports contain activity details for the selected audit area, including the following information:

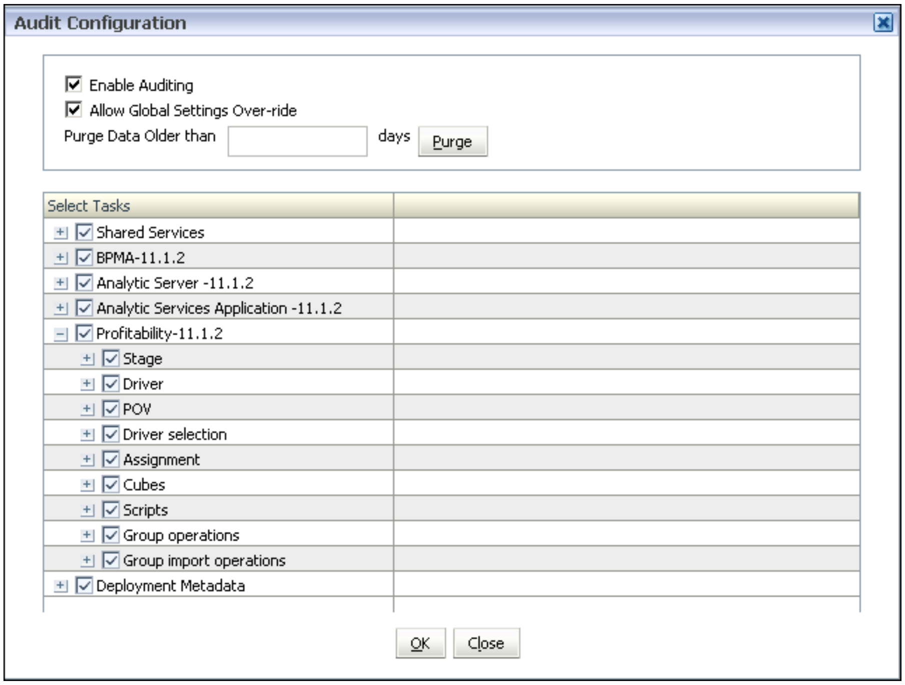
- Date
- Application
- User
- Artifact type and name
- Task that was performed

Auditing must be enabled before you can generate reports, as outlined in the following procedure. These reports can be exported as CSV files. See the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.

➤ To enable auditing:

1. From EPM Workspace, select **Navigate**, then **Administer**, and then **Shared Services Console.**
2. From the Shared Services Console, select **Administration**, then **Configure Auditing**.

   The Audit Configuration screen is displayed.



Audit Configuration

☑ Enable Auditing
☑ Allow Global Settings Over-ride
Purge Data Older than ______ days Purge

Select Tasks

- ☑ Shared Services
- ☑ BPMA-11.1.2
- ☑ Analytic Server -11.1.2
- ☑ Analytic Services Application -11.1.2
- ☑ Profitability-11.1.2
  - ☑ Stage
  - ☑ Driver
  - ☑ POV
  - ☑ Driver selection
  - ☑ Assignment
  - ☑ Cubes
  - ☑ Scripts
  - ☑ Group operations
  - ☑ Group import operations
- ☑ Deployment Metadata

OK Close

3 Select **Enable Auditing.**

The **Allow Global Settings Over-ride** and **Select Tasks** list are activated.

4 Under **Select Tasks**, select the areas of the application to be enabled for audit. You can select an entire area, or expand each option to choose separate steps to monitor.

Table 4 Profitability and Cost Management Audit Tasks

Monitored Area	**Available Monitored Tasks**
Stage	● Create stage ● Modify stage ● Delete stage ● Export stage ● Import stage
Driver	● Create driver ● Modify driver ● Delete driver ● Export driver ● Import driver
POV	● Create POV ● Modify POV ● Delete POV ● Export POV ● Import POV

Monitored Area	Available Monitored Tasks
Driver Selection	● Create driver selection ● Modify driver selection ● Delete driver selection
Assignment	● Create assignment ● Modify assignment ● Delete assignment
Assignment Rules	● Create assignment rule ● Modify assignment rule ● Delete assignment rule ● Create rule selection ● Remove rule selection
Cubes	● Deploy calculation cube ● Deploy reporting cube ● Transfer data
Genealogy	● Calculate genealogy
Scripts	● Deploy allocation script ● Deploy genealogy script ● Deploy POV copy script ● Execute allocation script ● Execute genealogy script ● Execute POV copy script
Group operations	● Copy assignments ● Delete assignments ● Delete assignment rules' selections ● Copy driver selections ● Delete driver selection rules ● Delete driver selection exceptions
Group import operations	● Import stages ● Import POVs ● Import drivers ● Import driver selections ● Import assignments
Group export operations	● Export stages ● Export POVs ● Export drivers ● Export driver selections ● Export assignments

Monitored Area	Available Monitored Tasks
Lifecycle Management (LCM)	● LCM Load from File ● LCM Save to File ● LCM Export ● LCM Import

5 Click **OK**.

A message is displayed to confirm the audit configuration has been saved.

6 Click **Yes**.

7 **Optional:** Generate audit reports to review audit results, as outlined in the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.

Output Log Files

Administrators can generate log files throughout EPM Workspace to help technicians identify system or environmental problems, or to help developers debug reports or API programs.

The following log files are available for information concerning Profitability and Cost Management:

Table 5 Profitability and Cost Management Log Files

Log File	Description
`hpcm.log`	Profitability and Cost Management generates an application, server-side log file that collects application-specific messages that are sent from the application or server. By default, the log files are available at `C:\oracle\Middleware\user_projects\domains\EPMSystem\servers\Profitability0\logs`. Contact your system administrator for access to this log file.
`SharedServices_Security_Client.log`	A Shared Services Client-side log file provides details regarding the Profitability and Cost Management handshake with Common Security Services. By default, the log file is available at `C:\oracle\Middleware\user_projects\domains\EPMSystem\servers\Profitability0\logs`.

For additional log files for related products and applications, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

To modify the level of detail to be captured in the log files, see "Using System Logs" in the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.

Oracle Diagnostic Logging (ODL) Files

The Oracle Diagnostic Logging (ODL) location for Web applications for all the configuration logging files for each EPM Workspace product are centralized.

For Profitability and Cost Management, the configuration logging file is available at:

`C:\oracle\Middleware\user_projects\domains\EPMSystem\config\fmwconfig\servers\Profitability0`. The file name is `logging.xml`.

For additional information on the configuration logging file, see "Using EPM Logs" in the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*.

3 Working with Profitability and Cost Management Dimensions and Members

In This Chapter

About Dimensions and Metadata

Dimensions and metadata are created and maintained by the Profitability and Cost Management Administrator, using Performance Management Architect. By using common dimensions and members, Profitability and Cost Management can easily use and transfer common data with other products, such as Planning, saving time and effort, and improving accuracy. For detailed information on creating and maintaining dimensions and metadata, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Profitability and Cost Management uses the dimensions and members created in Performance Management Architect to represent many of the structural elements of the business model:

- Dimensions that provide the structure for the model and scenarios, such as the Measures dimensions.
- Business dimensions that reflect the business-specific elements of the model, such as departments, General Ledger accounts, activities, customers, or products
- Point Of View (POV) dimensions, such as time periods, scenarios and versions
- Alias dimensions, as required
- Attribute dimensions, as required

For each dimension, both a dimension type and dimension name must be specified:

- Dimension type is a dimension property that enables the use of predefined functionality for selected dimensions. See "Dimension Types" on page 29.

- Dimension name is assigned to identify the contents of the dimension, in relation to your organization or business. For example, a dimension of Account type may be given a dimension name, such as General Ledger or Chart of Accounts. The dimension name does not need to reflect the dimension type, although it may. For naming restrictions, see "Essbase Naming Restrictions for Dimensions, Members, and Aliases" on page 105.

Using Performance Management Architect, dimensions can exist in an Application either as Shared or Local:

- Shared Dimensions reside in the Shared Library in Performance Management Architect, and can be used by multiple products and applications.
- Local Dimensions are detached, independent dimensions that only exist in one application instance, such as an application of type Profitability and Cost Management. These dimensions are used only within that application for which they have been created, and are not visible or usable even for another application of the same type.

Through Performance Management Architect, you select dimensions and members that exist in other products, or create new dimensions and members specifically for the model. After the dimensions and members are selected for the Profitability and Cost Management model, they are automatically available in the Dimension Library. After deployment, the dimensions and members are available in the Profitability and Cost Management application.

Both the system-generated and user-defined dimensions and members must exist in Performance Management Architect. Dimensions may contain alphanumeric characters, or calculated values.

Caution! Although there is no physical limit to the number of dimensions and members that can be created, performance issues occur with large dimensional structures.

The sort order for dimensions in a Profitability and Cost Management model must be set in a certain sequence, to maximize processing and calculation. See "Dimension Sort Order and Density Settings" on page 39.

For detailed instructions on creating and maintaining the dimensions and members, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide* and the *Oracle Essbase Database Administrator's Guide*.

See these sections for information about the Profitability and Cost Management dimensions:

- "Dimension Types" on page 29
- "Measures Dimensions" on page 29
- "AllocationType Dimension" on page 35
- "Business Dimensions" on page 37
- "Attribute Dimensions" on page 38
- "Dimension Sort Order and Density Settings" on page 39

Dimension Types

For Profitability and Cost Management, these types of dimensions are available in the Essbase outline:

- System dimensions, such as the Measures and AllocationType dimensions, must be created based on the requirements listed in these sections:
  - "Measures Dimensions" on page 29
  - "AllocationType Dimension" on page 35
- Alias dimension is used to assign alternate names, descriptions, languages, or other items that help to define dimensions.
- Business dimensions are created to describe the business elements within the model, such as departments, general ledger accounts, activities, customers, or products. See "Business Dimensions" on page 37.
- Attribute dimensions enable analysis based on the attributes or qualities of dimension members. Attributes describe characteristics of data, such as the size or color of products.
- User-defined attributes (UDAs) enable analysis based on text attributes that can be used to classify members of another associated dimension. For example, you may add a UDA called "New Products."
- POV dimensions indicate a specific point of view or version of the model, such as year, scenario, or period. At least one POV dimension is required for each model.

  You can also create a Version dimension type to maintain separate versions of the same POV. The version is used to track different versions of the same model and evaluate the impact of changes or different strategies on the model.
- Some Performance Management Architect dimension types are available for use in Profitability and Cost Management models:
  - Account
  - Entity
  - Country
  - Currency

  See the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide* for information on using these dimension types.

Note: When defining dimensional outlines, there are restricted characters that may not be used for naming. Oracle strongly suggests that you review the Essbase naming conventions in the *Oracle Essbase Database Administrator's Guide*.

Measures Dimensions

The Measures dimension contains the members required to build, validate and calculate a model.

The Measures dimension is created in Performance Management Architect in one of two ways:

- The user creates the Profitability and Cost Management application using the application wizard, and selects the option to **Auto Create Local Dimensions**. The system automatically creates a Measures type dimension.
- The user creates the application manually, by selecting “Create Blank Application” from the application wizard. The user must create their own dimensions and select the Measures dimension type.

In addition to members required for the Measures dimension **which should not be modified**, users may add user-defined driver measures (or sub-hierarchies) to the hierarchy under the member 'UserDefinedDriverMeasures'. These dimension members can be modified by the user.

Note: You can give the Measures type dimension a different name, if required.

Within the Measures dimension, the types of measures can be grouped into these categories:

- Driver Measures are used in the creation of driver types and formulas. See “Driver Measures” on page 30
- Cost Layer Allocation Measures are used to control allocation of calculated and input costs. See “Cost Layer Allocation Measures” on page 31.
- Revenue Layer Allocation Measures are used to control the allocation of calculated and input revenue. See “Revenue Layer Allocation Measures” on page 33.
- Reporting Measures are used to generate reports, using the calculated and input values to generate total costs and revenue for the model. All non-level-0 Reporting Measures are calculated. See “Reporting Measures” on page 34.

Driver Measures

Table 6 Driver Measures

Member Name	Alias	Description	Calculated or Input
FixedDriverValue	FixedDV	Default measure for use with driver types that require a fixed driver value parameter	Input
Rate	Rate	Default measure for use with driver types that require a rate parameter	Input
Quantity	Qty	Default measure for use with driver types that require a quantity parameter	Input
Weight	Weight	Default measure for use with driver types that require a weight parameter	Input
Percentage	Percent	Default measure for use with a percentage driver type	Input
CalculatedDriverValue	CDV	Measure that is the result of the driver formula used in an assignment	Calculated

Member Name	Alias	Description	Calculated or Input
TotalDriverValue	TDV	Measure that is used as the denominator in the allocation formula DV/ TDV	Calculated
EffectiveTotalDriverValue	EffTDV	Measure that is used to store the Effective Driver Total for drivers that have the “Allow idle” box checked when the driver is defined.	Calculated
OverrideTotalDriverValue	OvrdTDV	User-entered value that overrides the TotalDriverValue measure as the denominator in allocations.	Input
TotalDriverValueAfterReciprocals	TDVAftRcp	Measure that is used as the denominator in allocation formulas for non-reciprocal intrastage and post-stage allocations, when a source has been involved in a reciprocal assignment	Calculated
IdleDriverValue	IdleDV	Measure that is used as the driver value (DV) for calculating IdleCost	Calculated
UserDefinedDriverMeasures	N/A	The member UserDefinedDriverMeasures is where the application-specific, user-defined driver measures are stored. Set the ASOMember DataStorage and BSOMember DataStorage properties as follows: – If this member does not have children, set to **StoreData**. – If members are added as children to this member, and all these children have the consolidation symbols of IGNORE, then set to **LabelOnly**. **Note:** All driver measures must be unique in the outline. Do not use the name of an existing driver measure in a dimension in the outline as the name of another member (including system, POV and business dimensions) ; otherwise, the Data Entry screen will not properly display the values.	N/A

Cost Layer Allocation Measures

Table 7 Cost Layer Allocation Measures

Member Name	Alias	Description	Calculated or Input
UnassignedCost	UnAsgCost	Remainder of cost at a source intersection after all assignment and idle calculations have been completed.	Calculated
CostAssigned	CostAsg	Total cost assigned from a source to post-stage destinations and non-reciprocal intrastage destinations	Calculated
CostAssignedIntraStage	CostAsgInt	Sum of costs assigned to intrastage destinations, excluding reciprocal destinations	Calculated
CostAssignedPostStage	CostAsgPost	Sum of costs assigned to post-stage destinations	Calculated
OverDrivenCost	OverDrivenCost	For a standard basis driver, if the total cost assigned is greater than the `NetCostForAssignment`, then the overage amount is posted to `OverDrivenCost`.	Calculated

Member Name	Alias	Description	Calculated or Input
IdleCost	IdleCost	Depending on the driver type, idle cost is generated differently: ● For actual basis drivers, idle cost is generated using the allocation formula: `IdleDriverValue/ OverrideTotalDriverValue` ● For standard basis drivers, idle cost is generated if the total cost assigned is less than NetCostForAssignment.	Calculated
NetCostForAssignment	NetCostAsg	Total cost available for assignment after accounting for all prior stage, intrastage, and reciprocal assignments Set the DataStorage (BSO) property to **StoreData**.	Calculated
GrossReceivedCost	GrRecCost	Sum of all costs assigned from prior stages and intrastage assignments, excluding reciprocals and user input costs Set the DataStorage (BSO) property to **StoreData**.	Calculated
StandardCostRate	StandardCostRate	For a Standard Basis driver, the user assigns a standard cost rate, and enters that value for use in calculations for the standard basis cost driver, as follows: `CostReceivedPriorStage =StandardCostRate * TotalDriverValue`	Input
CostInput	CostInput	User-entered cost value for the intersection	Input
CostReceived	CostRec	Sum of all costs assigned to an intersection by prior stage and intrastage assignments, excluding costs resulting from reciprocal assignments Set the DataStorage (BSO) property to **StoreData**.	Calculated
CostReceivedPriorStage	CostRecPri	Sum of costs received on assignments from a prior stage	Calculated
CostReceivedIntraStage	CostRecInt	Sum of all costs received on intrastage assignments, excluding reciprocal assignments	Calculated
NetReciprocalCost	NetRcpCost	Net effect of a reciprocal assignment on the amount available for assignment to poststage and nonreciprocal intrastage destinations Set the DataStorage (BSO) property to **StoreData**.	Calculated
ReciprocalCostAssigned	RcpCostAsg	Total cost assigned to reciprocal destination, excluding costs received from the reciprocal	Calculated
ReciprocalCostReceived	RcpCostRec	Total cost received from a reciprocal destination	Calculated
ReciprocalIntermediate Cost	RcpIntCost	Intermediate value calculated for an intersection after simultaneous equations have been applied, but before reciprocal adjustments have been made	Calculated
CostPerDrvUnit	Cost Per Driver Unit	This measure is a child of AllocationMeasures. The formula uses the assigned cost (CostAssigned) divided by the sum of all driver values (TotalDriverValue) to calculate the cost for each unit of driver value.	Calculated

Member Name	Alias	Description	Calculated or Input
UnitCost	Unit Cost.	This measure is a child of AllocationMeasures. The formula uses the amount of cost at a source intersection (NetCostForAssignment) divided by a user-input quantity to calculate the cost per unit.	Calculated

Revenue Layer Allocation Measures

Table 8 Revenue Layer Allocation Measures

Member Name	Alias	Description	Calculated/ Input
UnassignedRevenue	UnAsgRev	Remainder of revenue at a source intersection after all assignment and idle calculations have been completed	Calculated
RevenueAssigned	RevAsg	Total Revenue assigned from a source to post-stage and non-reciprocal intrastage destinations	Calculated
OverDrivenRevenue	OverDrivenRevenue	For a standard basis driver, if the total revenue is greater than the `NetRevenueForAssignment`, then the overage amount is posted to `OverDrivenRevenue`.	Calculated
RevenueAssignedIntraStage	RevAsgInt	Sum of revenue assigned to intrastage destinations, excluding reciprocal destinations	Calculated
RevenueAssignedPostStage	RevAsgPos	Sum of revenue assigned to post-stage destinations	Calculated
IdleRevenue	IdleRev	Depending on the driver type, idle revenue is generated differently: ● For actual basis drivers, idle revenue is generated using the allocation formula: `IdleDriverValue/ OverrideTotalDriverValue`. ● For standard basis drivers, revenue cost is generated if the total cost assigned is less than `NetRevenueForAssignment`.	Calculated
NetRevenueForAssignment	NetRevAsg	Total revenue available for assignment after accounting for all prior stage, intrastage, and reciprocal assignments Set the DataStorage (BSO) property to to **StoreData**.	Calculated
GrossReceivedRevenue	GrRecRev	Sum of all revenue assigned from prior stages and intrastage assignments, excluding reciprocal assignments and user input revenue Set the DataStorage (BSO) property to **StoreData**.	Calculated
StandardRevenueRate	StandardRevenueRate	For a Standard Basis driver, the user assigns a standard revenue rate, and enters that value for use in calculations for the standard basis revenue driver, as follows: `RevenueReceivedPriorStage =StandardRevenueRate * TotalDriverValue`	Input

Member Name	Alias	Description	Calculated/ Input
RevenueInput	RevInput	User-entered revenue values for the intersection Define and store Revenue categories as a hierarchy under `RevenueInput`. Set the DataStorage (BSO) property to **StoreData**.	Input
RevenueReceived	RevRec	Sum of all revenue assigned to an intersection by prior stage and intrastage assignments, excluding revenue results from reciprocal assignments Set the DataStorage (BSO) property to **StoreData**.	Calculated
RevenueReceivedPriorStage	RevRecPri	Sum of revenue received on assignments from a prior stage	Calculated
RevenueReceivedIntraStage	RecRecInt	Sum of all revenue received on intrastage assignments, excluding reciprocal assignments	Calculated
NetReciprocalRevenue	NetRcpRev	Net effect of a reciprocal assignment on the amount of revenue available for assignment to poststage destinations and nonreciprocal intrastage destinations Set the DataStorage (BSO) property to **StoreData**.	Calculated
ReciprocalRevenueAssigned	RcpRevAsg	Total revenue assigned to reciprocal destinations, but excluding revenue received from the reciprocal assignment	Calculated
ReciprocalRevenueReceived	RcpRevRec	Total revenue received from a reciprocal destination	Calculated
ReciprocalIntermediateRevenue	RcpIntRev	Intermediate value calculated for an intersection after simultaneous equations have been applied, but before reciprocal adjustments have been made	Calculated

Reporting Measures

Table 9 Reporting Measures

Member Name	Alias	Description	Calculated or Input
Profit	Profit	Value of the calculated profit for the selected intersection. This value is the result of the calculation: `NetRevenueForAssignment – NetCostForAssignment`	Calculated
GrossCost	GrossCost	Total cost for an intersection, including all possible inputs: ● Input values ● Prior stage assignments ● Intrastage assignments including reciprocals. This calculation represents the true total cost of the intersection.	Calculated
StandardCost		For Standard Basis drivers, the calculated cost of `StandardCostRate * TotalDriverValue`	Calculated

Member Name	Alias	Description	Calculated or Input
StandardRevenue		For Standard Basis drivers, the calculated revenue of `StandardRevenueRate * TotalDriverValue`	Calculated
InitialCost	InitialCost	Cost of an intersection before intrastage or reciprocal costs are calculated, including input costs and cost received on assignments from prior stages.	Calculated
NetCostAfterIntraStage	NetCostAftInt	Cost of an intersection, including all intrastage cost assignments	Calculated
GrossRevenue	GrossRev	Total revenue for an intersection, including all possible inputs: ● Input values ● Prior stage assignments ● Intrastage assignments including reciprocals. This calculation represents the true total revenue for the intersection.	Calculated
InitialRevenue	InitialRev	Revenue for an intersection before intrastage or reciprocal revenue is calculated, including input revenue, and revenue received on assignments from prior stages.	Calculated
NetRevenueAfterIntra Stage	NetRevAftInt	Revenue for an intersection after accounting for all types of intrastage revenue assignments	Calculated

AllocationType Dimension

The AllocationType dimension is used to store both direct allocations and allocation genealogy data. When you create reports, the AllocationType dimension enables you to specify which type of allocation data to retrieve.

Note: You can give the AllocationType type dimension a different name, if required.

The system automatically generates the AllocationType dimension if the user creates an application using the wizard, and checks “Auto Create Local Dimensions.” If the user selects “Create Blank Application”, the user must create their own dimensions and select the AllocationType dimension type.

This dimension cannot be modified, and it is not visible in Performance Management Architect or Profitability and Cost Management.

In the Essbase outline created by Profitability and Cost Management, the AllocationType dimension contains the following members:

- **AllAllocations** contains the following child members:
  - **DirectAllocation** stores calculated data that has been directly allocated within the model, between a specified source intersection and destination intersection. Direct allocation must be defined as an assignment by the user.

- ❍ **GenealogyAllocation** stores the allocation genealogy that is calculated on indirect links between various concerned intersections in the model.

  Genealogy allocation is not directly defined by the user, but it exists because of two or more direct allocations. For example, A-B-C allocation genealogy data exists because there is a direct allocation from A to B (A-B) and B to C (B-C).

- **SysAllocVar1** stores the value for intrastage assignments on the virtual link, which is part of the value allocated to another node on the same stage, and is available in the `DirectAllocation` member.
- **SysAllocVar2** is used to obtain a sum of the source links of `DirectAllocation`, `Genealogy Allocation` and `SysAllocVar1`.
- **SysAllocVar3** stores calculated genealogy data that is used within the system. Do not use this member in reports.
- **TotalAllocation** dynamically calculates the sum of the source links of `DirectAllocation`, `GenealogyAllocation,` and `SysAllocVar3`.
- **IndirectAllocation** dynamically calculates the sum of the source links of `GenealogyAllocation` and `SysAllocVar3`.

Caution! Do not edit the system members in this dimension, as modifications may result in the loss of data or the corruption of your model.

Alias Dimensions

Aliases are alternate names, descriptions, languages, or other items that help to define dimensions. For example, you may refer to a customer number in the system, but you can assign an alias that displays the company name on the screen, to make it easier to identify that client. You can assign one or more aliases to accounts, currencies, entities, scenarios, periods, versions, years, and user-defined dimension members.

Note: Duplicate member names or aliases are not allowed within the same dimension.

For Profitability and Cost Management, the alias must be set in Performance Management Architect. For detailed instructions on creating Alias dimensions, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Caution! If an Alias association is deleted in Performance Management Architect, it is not deleted from the model.

When installation is complete, a “Default” alias table is available. After redeployment, you can view the alias on all screens that use the Common Member Selector, including Driver Selections, Assignments, Data Entry, and Traceability. To search and filter aliases, you must first select **Show Alias** from the Context Menu.

Note: The Alias View is not available on the Select Driver selector, which is accessed when adding or modifying a driver.

Aliases may be cloned if a particular dimension is cloned in Essbase.

➤ To view Aliases:

1 **From the application, select any screen that uses the Common Member Selector, for example, Driver Selections, Assignments, Data Entry, or Trace Allocations. The appropriate screen with Common Member Selector is displayed, showing all available members.**

2 **Optional: Click the selector button ... or the Add button 🗋, for example, on the Driver Selections or Trace Allocations screen.**

The Select Member dialog box opens, showing all available members.

3 **Optional: In the Common Member Selector, click the Context Menu button , and select Show Alias.**

Aliases are displayed in the list of members.

Note: If you select Show Alias from the Context Menu, and no alias is assigned, the member name is displayed within square brackets. For example, the member name Product is displayed as `[Product]`

Business Dimensions

Business dimensions describe the business-specific objects within each stage in the model, such as products, customers, regions, employees, and so on. These dimensions and members are created in Performance Management Architect.

Business dimensions may use some or all of the following dimension types, and may apply to one or more stages or models:

- Generic
- Account
- Entity
- Country

When the Essbase outlines are deployed, the business dimensions are created in the Profitability and Cost Management application as basic or generic dimensions, with no type. This feature enables Profitability and Cost Management to reuse the dimension member and hierarchies that were defined for other applications, such as Planning.

Note: This dimension type does not apply to aggregate storage outlines.

When creating a business dimension, the following requirements apply:

- The following properties for the Gen1 member of the dimension must be set to `LABEL_ONLY`:
  - DataStorage(BSO)
  - DataStorage(ASO)
- The first Gen2 child under the Gen1 dimension name is usually set to an ALL member. For example, `AllDepartments` for the Departments dimension.

  The primary hierarchy is hosted under the first Gen2 child. Only the first Gen2 hierarchy is used in allocation modeling, and this hierarchy cannot contain shared members.
- Additional Gen2 members can host alternate hierarchies, but these hierarchies are not used in allocation modeling. If the dimension is going to host alternate hierarchies, set the DimensionHierarchyType to 'Enabled.'

  These alternate hierarchies are not visible in Profitability and Cost Management modeling screens, and can only be viewed in Essbase.
- A `NoMember` member is required. The last Gen2 child in the hierarchy must always be `NoMember` with consolidation set to `IGNORE (~)`.

Note: A `NoMember` member need not be created for ADS or flat files, because it is added automatically by Performance Management Architect. Do not delete this member.

POV Dimensions

POV dimensions indicate a specific point of view or version of the model, such as year, scenario, or period. The dimension can be customized to reflect the requirements of your organization. For example, the POV may consist of quarters, months, seasonal groupings, and so on.

At least one POV dimension is required for each model, but you can create up to four POV dimensions. The POV dimensions are set in the Performance Management Architect Property grid for the model.

A Version dimension is also available, and is used to create another instance of your model. This version can be modified to enable you to experiment with strategies or business options to play "what-if" scenarios. By modifying the version, you can implement features, and compare results to determine the best course of action.

Attribute Dimensions

Attribute dimensions are a special type of dimension that are associated with a business dimension, and contain members that can be used to classify members of another, associated dimension. Attribute dimensions describe characteristics of data, such as the size and color of products.

You can use these attributes to analyze data, based on the attributes or qualities of dimension members. The attribute dimensions are also used for filtering destination intersections when assignment rules are created.

Note: For naming restrictions, see "Essbase Naming Conventions for Attribute Calculations Dimension" on page 108.

Two Attribute dimension types exist in Performance Management Architect:

- Attribute Dimensions:
  - The attribute can be created using different structures, such as Boolean, Date, Numeric, and Text.
  - An attribute has a hierarchy, and the hierarchies can be rolled up for aggregate values.
  - Only one attribute from a given attribute dimension can be associated with one member.
- User-Defined Attribute Dimensions (UDAs):
  - The attribute can only be created using Text.
  - A UDA does not have a hierarchy and cannot be easily used in reports to generate sums.
  - Multiple UDAs can be associated with one member.

Each type of attribute dimension offers different advantages, depending on your model and reporting needs. For detailed information about working with attribute dimensions, see the *Oracle Essbase Database Administrator's Guide*.

You can use these attributes to analyze data, based on the attributes or qualities of dimension members. The attribute dimensions are also used for filtering destination intersections when creating assignment rules.

For naming restrictions, see "Essbase Naming Conventions for Attribute Calculations Dimension" on page 108.

Dimension Sort Order and Density Settings

The Dimension Sort Order property controls the order of dimensions in the Essbase outline that is generated by Profitability and Cost Management. The dimension sort order must be set on all dimensions within a model, except Alias and UDA.

Caution! If the sort order for a dimension is left blank, the validation will fail.

The Dimension Sort Order property is set in Performance Management Architect, and passed on to Profitability and Cost Management during deployment for use in generating the Essbase outline. For instructions, see "Setting the Dimension Sort Order" on page 40.

The Dimension Sort Order settings for the model are validated in Performance Management Architect. See "Validating and Deploying the Application in Performance Management Architect" on page 61.

You can also set the density for dimensions from the Dimension Sort Order Settings dialog box.

Dimension Sort Order Recommendations

Oracle recommends that you set the dimension sort using the following recommendations:

- A dimension sort order must be set for every dimension in the model, except Alias and UDA.

  Note: The Alias and UDA dimensions are ignored for Dimension Sort Order, as they do not exist as dimensions in Profitability and Cost Management and Essbase.

- The dimension sort order must be sequential, unique, and greater than or equal to 1.
- Measures dimension is set to 1, by default.
- AllocationType dimension is set to 2, by default.
- Business and POV dimensions must be set to 3 or higher.
- Attribute dimensions must always be sorted as the last dimensions. For example, if you have four attribute dimensions in a sequence of 12 dimensions, the attribute dimensions must be set as 9, 10, 11, and 12.

Setting the Dimension Sort Order

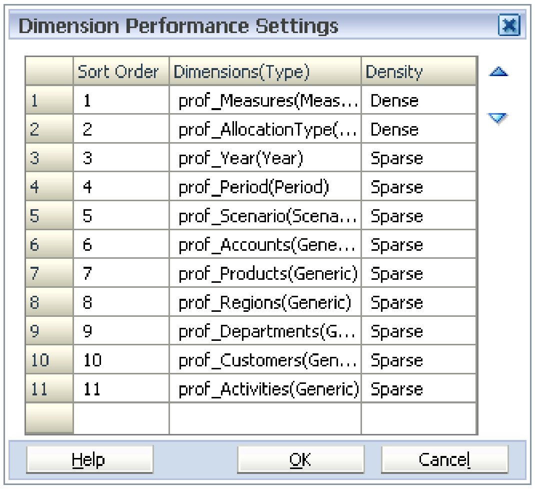
The processing order for every dimension in the model must be set at the dimension-level using the Dimension Sort Order property. The Dimension Sort Order restrictions must be met; otherwise, validation of the model will fail. For a complete list of restrictions, see "Dimension Sort Order Recommendations" on page 40.

There are two ways to set the dimension sort order in Performance Management Architect:

- Using the Dimension Performance Settings dialog box to set the sort order for all dimensions at one time
- Using the Property Grid in the Dimension Library to set individual dimension sort order for one dimension at a time

➤ To set the Dimension Sort Order using the Dimension Performance Settings dialog box:

1. From EPM Workspace, select **Navigate**, then **Administer**, and then **Dimension Library** to display the Shared Library.
2. Right-click the name of the application, and select **Dimension Performance Settings.**



	Sort Order	Dimensions(Type)	Density
1	1	prof_Measures(Meas...	Dense
2	2	prof_AllocationType(...	Dense
3	3	prof_Year(Year)	Sparse
4	4	prof_Period(Period)	Sparse
5	5	prof_Scenario(Scena...	Sparse
6	6	prof_Accounts(Gene...	Sparse
7	7	prof_Products(Generic)	Sparse
8	8	prof_Regions(Generic)	Sparse
9	9	prof_Departments(G...	Sparse
10	10	prof_Customers(Gen...	Sparse
11	11	prof_Activities(Generic)	Sparse

3 Select a dimension, and use the Up ⌃ and Down ⌄ arrows to move each dimension into the correct sort order, as described in "Dimension Sort Order Recommendations" on page 40. The number of the original position for the dimension is displayed under Sort Order.

4 **Optional:** Under **Density** for the selected dimension, double-click the cell to display the options, and then select the appropriate density for the dimension. See "Optimizing Dimension Settings for Essbase" on page 44.

5 Click **OK**.

➤ To set Dimension Sort Order for individual dimensions:

1 From EPM Workspace, select **Navigate**, then **Administer**, and then **Dimension Library**.

2 In the Profitability and Cost Management application, select the dimension for which you want to set the Dimension Sort Order.

3 In the Property Grid, select the Dimension Sort Order property, and under **Value**, enter the required number for the sort order. See "Dimension Sort Order Recommendations" on page 40.

4 Click the Save icon 💾.

4 Managing Applications and Dimensions Using Performance Management Architect

In This Chapter

Working with Applications and Dimensions

Profitability and Cost Management uses Performance Management Architect to select dimensions to build the Essbase outline that is used for the profitability model. All dimensions and members are created in Performance Management Architect, and imported into the Profitability and Cost Management application to build the model.

Note: Lifecycle Management can be used not only to import or export model data, but also to import or export applications in Performance Management Architect.

From Performance Management Architect, you can perform the following tasks:

- Create, edit and copy dimensions
- Set up aliases
- Create, view, and delete dimension associations
- Create, view, rename, and delete members
- Edit property values
- Deploy applications to Profitability and Cost Management
- View transaction logs
- Synchronize data between Profitability and Cost Management and other applications, Essbase databases (ASO and BSO), external source (flat files), and interface tables

Refer to the following sections for more information on working with metadata and dimensions using Performance Management Architect:

- "Optimizing Dimension Settings for Essbase" on page 44
- "Importing Metadata" on page 45
- "Profitability and Cost Management Dimension and Member Properties" on page 46
- "Validating and Deploying Applications Using the Wizard" on page 62
- "Synchronizing Data" on page 65

Optimizing Dimension Settings for Essbase

A typical Profitability and Cost Management application contains one Measures dimension, one AllocationType dimension, several POV dimensions and a number of business dimensions. Profitability and Cost Management duplicates business dimensions if they are used in more than one stage. This process increases the Sparsity of the Essbase Calculation Cube outline generated by the application, and may have a performance impact when the calculation scripts are run.

Some optimization can be achieved simply by changing the Dimension Storage Type directly for the generated Calculation Cube outline, using Oracle Essbase Administration Services (EAS) console.

Caution! Changes to the Dimension Storage Type property should only be performed by a Database Administrator (DBA). For detailed instructions, refer to the *Oracle Essbase Administration Services Developer's Guide*.

The default recommendation to set the Dimension Storage Type property for these dimensions follows:

- Set the Measures and the AllocationType dimensions to Dense
- Set all business and POV dimensions to Sparse

This default setting results in an Essbase calculation cube with the block size of about 3K, and a large number of potential blocks based on the dimensionality of the Sparse dimensions. The most dense dimension (based on the existence of data) in the largest stage should be set to Dense. In that case, the AllocationType and Measures dimension should be set to Sparse.

Note: The block size should still remain within the limits of the Essbase best practices recommendations. See the *Oracle Essbase Administration Services Developer's Guide*.

Example

The largest stage is defined by the largest number of potential nodes in the stage.

If a typical Profitability and Cost Management model has its last stage defined as Customers X Products (to calculate Customer x Product profitability), and it is the largest stage, then either the Customer or the Product dimension in this stage can be set to Dense. Set the associated Measures and AllocationType dimensions to Sparse. The decision to set Customers to Dense or Products to Dense is based on the data density of the governing drivers.

For example, if Sales Volume is the driver that is used to allocate to this stage from various sources in previous stages, and the Customers dimension has more dense data than the Products dimension (few Products sold to more Customers), the Customers dimension can be set to Dense. If more Products are sold to fewer Customers, then Products can be set to Dense. In either case, the Essbase block size should remain within the limits of the Essbase best practices recommendations.

Note: Dimensions with attribute associations cannot be set to Dense. Essbase allows attribute associations to be set as Sparse dimensions only.

Importing Metadata

The Dimension Library in Performance Management Architect provides a central location for administrators to view, create, and manage dimensions and hierarchies. You can use existing dimensions that are shared with other applications, or create local dimensions that are only for your model.

To use the Dimension Library, you must have the Dimension Editor security role. This role permits access to all Dimension Library functionality, such as creating, deleting, and changing dimensions and members, creating import profiles, and running transaction logs. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.

Performance Management Architect has two types of dimensions:

- Shared dimensions are linked to the Shared Library and inherit all changes that are made to the dimension.
- Local dimensions are copied from the Shared Library to the application. Local dimensions do not inherit future changes made to the dimension in the Shared Library.

Caution! All Profitability and Cost Management properties are local values. If you make changes to this value, it is not automatically inherited if the property exists in other hierarchies. If you modify a property in one hierarchy, you cannot assume that the values will be inherited to other hierarchies.

From the Dimension Library, you can edit these properties:

- Applications
- Dimensions
- Members

Note: Standard dimensions for Time, Currency and Country are available in Performance Management Architect for all products.

➤ To access the Dimension Library, select **Navigate**, then **Administer**, then **Dimension Library**.

See *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide* for detailed instructions on working with dimensions and members.

Profitability and Cost Management Dimension and Member Properties

The properties for Profitability and Cost Management dimensions and members are displayed in alphabetical order in Table 10, "Profitability and Cost Management Dimension and Member Properties," on page 46, which displays the following information:

- The Property Label, which provides a more readable display name for the property. If applicable, the associated database type is appended to the name (ASO or BSO). If no database type is specified, the property applies to both types.
- A Description of each property
- The Property Name, which provides a unique identifier for the property that is used when updating data in the import and batch client.

You can modify any property that presents a drop-down list or data entry text box when you select the property in the Dimension Library.

See *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide* for detailed instructions on working with dimensions and members.

Caution! All Profitability and Cost Management properties are local values. If you modify a property in one hierarchy, you cannot assume that the values will be inherited by other hierarchies.

Table 10 Profitability and Cost Management Dimension and Member Properties

Property Label	Description	Property Name
Alias	Enter the alias for the selected dimension member. The alias is the dimension member name that is displayed in a deployed application. ● To assign or change the default alias name for the current dimension member, click the existing alias name and enter the new one. Follow the proper naming rules. All databases have an alias table named Default. ● To assign or change an alias name for the current dimension member, click the existing alias name next to the alias table and enter the new name. Follow the proper naming rules.	
Attributes	Enter the characteristics of a dimension member. For example, Employee dimension members may have attributes of Name, Age, or Address. Product dimension members may have several attributes, such as a size and flavor.	Attributes

Property Label	Description	Property Name
Attribute Type	Associate an attribute type with the selected member. Create attribute values for attributes assigned to dimension members that you can use to query on and to filter members. Attribute types: ● Text ● Boolean ● Date ● Numeric For example, the attribute type for “Caffeinated” would be Boolean; data values would be True or False.	AttributeDataType
Comment	Enter a comment for the dimension or member. Comments can contain up to 255 characters. By default, this text box displays the current comment, if one exists. You cannot assign a comment to an attribute dimension or member. **Note:** For Unicode-enabled databases, a maximum of 80 characters is allowed.	Comment
Consolidation	Member consolidation properties determine how children roll up into their parents. If the current member is not a dimension or an attribute, select the consolidation operator to assign to the member: ● + (addition) - Default ● - (subtraction) ● * (multiplication) ● / (division) ● % (percent) ● ~ (ignore during consolidation) ● ^ (never consolidate) ● NotUsed **Note:** Some restrictions exist regarding the use of consolidation operators in aggregate storage outlines. See the *Oracle Essbase Database Administrator's Guide* .	Consolidation
Data Storage(ASO)	Select the ASO storage options for dimensions and members: ● StoreData—Data is stored with the dimension. ● ShareData—Data associated with this member can be shared. The ShareData property applies to the member only. The Dimension Root Member cannot be shared. ● NeverShare—Data associated with this dimension cannot be shared, even if there is an implied share relationship, such as with a parent with one child. In this case, the data is duplicated in the parent and child. **Note:** This option does not apply to stored hierarchies in aggregate storage outlines. ● LabelOnly—No data is associated with this dimension.	● ASODimensionDataStorage (for dimension root member) ● ASOMemberDataStorage (for dimension members)

Property Label	Description	Property Name
Data Storage(BSO)	Select the BSO storage options for dimensions and members: ● StoreData—Data is stored with the dimension. ● ShareData—Data associated with this member can be shared. The ShareData property applies to the member only. The Dimension Root Member cannot be shared. ● NeverShare—Data associated with this dimension cannot be shared, even if there is an implied share relationship, such as with a parent with one child. In this case, the data is duplicated in the parent and child. ● LabelOnly—No data is associated with this dimension. ● DynamicCalcAndStore—Data associated with this dimension is not calculated until requested by a user. After calculation, the data is stored. ● DynamicCalc—Data associated with this dimension is not calculated until requested by a user. The data is not stored but is discarded after the request is completed.	● BSODimensionDataStorage (for dimension root member) ● BSOMemberDataStorage (for dimension members)
Dimension Formula(ASO)	Double-click the cell, and then click the Selector button to open the Memo Editor. Enter the formula, and then click OK. Use the appropriate ASO MDX syntax.	ASODimensionFormula (for dimension root member)
Dimension Formula(BSO)	Double-click the cell, and then click the Selector button to open the Memo Editor. Enter the formula, and then click OK. Use the appropriate BSO CALCULATOR syntax.	BSODimensionFormula (for dimension root member)
Dimension Solve Order	Enter the numeric value in the solution sequence for selected dimensions. For example, if this dimension is to be solved second, enter “2.”	DimensionSolveOrder
Dimension Sort Order	Enter the numeric value in the sequence to set the order of dimensions in the Essbase outline that is generated by Profitability and Cost Management. For example, if this dimension is to be the second in the Essbase outline, enter “2.” Dimension Sort Order must be set for every dimension in the model, except Alias and UDA dimensions. The dimension sort order must be sequential, unique, and greater than or equal to 1. Set the sort order as outlined in “Dimension Sort Order and Density Settings” on page 39.	DimensionSortOrder
Dimension Storage Type	The **Dimension Storage Type** property applies only to Block Storage (BSO) databases in Essbase. Select the type of storage required for the dimension: ● Dense ● Sparse	DimensionStorageType

Property Label	Description	Property Name
Hierarchy Type (Dimensions Only)	The **Hierarchy Type** property applies only to Aggregate Storage (ASO) databases in Essbase. The Reporting database is generated by Profitability and Cost Management. Set the type of hierarchy for the dimension: ● Select **STORED** for any dimension members that use the following consolidation symbols: ❍ +ADDITION ❍ ~ IGNORE **Note:** The no-consolidation or IGNORE (~) operator can only be used in a STORED hierarchy if the member's parent is set to LABEL_ONLY ● Select **DYNAMIC** for dimension members that use any consolidation symbol, including ADDITION and IGNORE, or if the dimension member contains a formula. ● Select **ENABLED** to support alternate hierarchies	DimensionHierarchyType
Hierarchy Type (Members Only)	Set the type of hierarchy for the member: ● Stored (For first Gen2 child) ● Dynamic (For Second Gen2 child onwards. The Second Gen2 member can host the alternate hierarchies) ● None (Do Not Use)	HierarchyType
Level Usage for Aggregation	This property applies to the Gen1 or Gen2 hierarchy members who host a stored hierarchy. in an ASO outline. Set to **DEFAULT** For additional information on this property, see the *Oracle Essbase Database Administrator's Guide*.	
Member Formula(ASO)	Double-click the cell and then click the Selector button to open the Memo Editor. Enter the formula, and then click OK. Use the appropriate ASO MDX syntax.	ASOMemberFormula
Member Formula(BSO)	Double-click the cell and then click the Selector button to open the Memo Editor. Enter the formula, and then click OK. Use the appropriate BSO (BSO CALCULATOR) syntax.	BSOMemberFormula
Member Solve Order (Members Only)	Enter the numeric value in the solution sequence for selected member. For example, if this member is to be solved second, enter “2.” This property applies to ASO databases only. Members that have a solve order of 0 inherit the solve order of their dimension. Members with the same solve order are evaluated in the order in which their dimensions appear in the database outline, unless otherwise specified by the dimension sort order property. Members with no solve order are evaluated after members with a solve order.	MemberSolveOrder
POV Dimension	Click the checkbox to set this dimension as a POV dimension for the selected model.	IsPOVDimension

Property Label	Description	Property Name
POV Display Order	If there are multiple POV dimensions, enter the numeric value (such as 1, 2, 3, and so on) to set the display order for each POV dimension.	POVDisplayOrder
Primary Level Weighting	Primary level weighting restricts the levels which can be selected by the view selection engine during aggregation operations. These settings only apply to Essbase (ASO) applications. Select one of the following options: ● **Default** – The view selection engine is free to decide which levels to pick. This is the initial value for new hierarchies. ● **AllLevels** – The view selection engine considers all levels of the hierarchy for aggregation. ● **NoAggregation** – The view selection engine cannot select any levels for aggregation. ● **TopLevelOnly** – The view selection engine considers only the highest level of the dimension for aggregation. ● **BottomaTop** – The view selection engine considers only the highest level and the lowest level of the dimension for aggregation. ● **BottomLevelOnly** – The view selection engine considers only the bottom level of the dimension for aggregation.	PrimaryLevelWeighting
Two Pass Calculation (Dimensions Only)	For BSO databases only, select the check box to calculate a member on the second pass through the outline.	TwoPassCalc (for dimension root member)
Two Pass Calculation (Members Only)	For BSO databases only, select the check box to calculate a member on the second pass through the outline.	TwoPassCalc
UDA	Enter the UDA name for the selected member. A UDA dimension must exist in Performance Management Architect, and be associated to the base to which the UDA is being added. See “Viewing Dimension Associations”. You can enter up to 256 alphanumeric characters. . **Note:** For Unicode-enabled databases, a maximum of 80 characters is allowed.	UDA
Unicode	Select **Unicode** to set the application to Unicode mode. Unicode-mode applications support multiple character sets.	Unicode

Creating Profitability Applications

You can create Profitability applications in Performance Management Architect using one of two methods:

- Create the Profitability and Cost Management application using the application wizard to automatically create the application and dimensions.
- Create a blank application, and manually select and name the dimensions.

The following sections provide step-by-step procedures for creating Profitability applications:

- "Selecting the Application Type" on page 51
- "Selecting Dimensions" on page 52
- "Modifying the Application Settings" on page 57
- "Validating and Deploying the Application in Performance Management Architect" on page 61

Selecting the Application Type

➤ To create Profitability and Cost Management applications:

1 **Populate the new shared library in Performance Management Architect using a flat file import or a Performance Management Architect interface table import.**

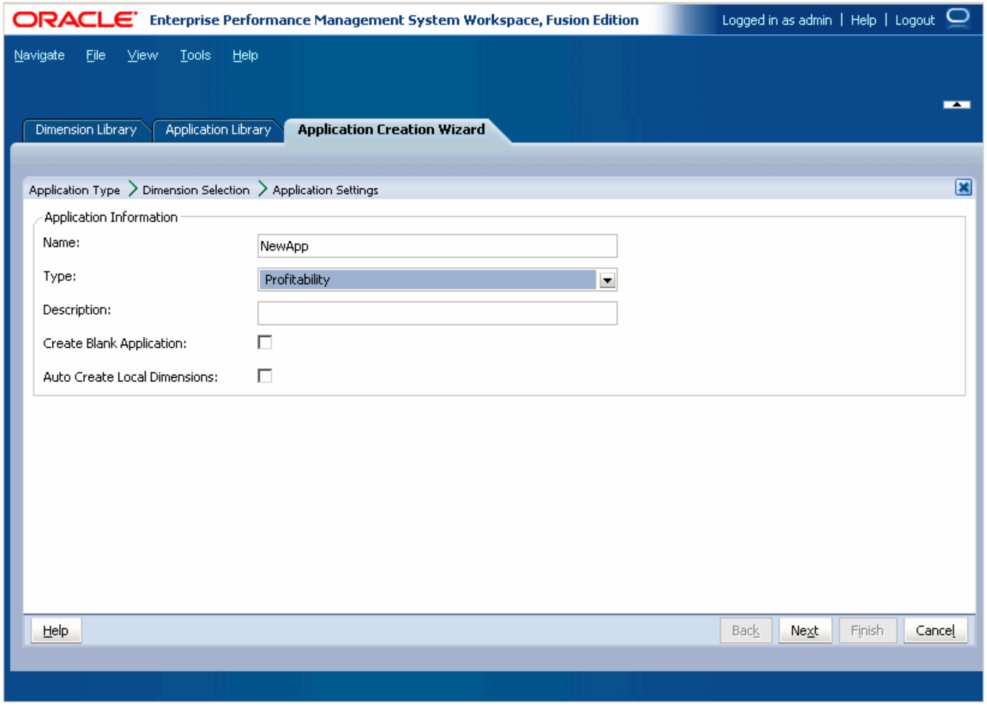
Caution! Add business dimensions to be included in the application, for example, Generic, Account, Entity, Time, or Country, to the Dimension Library before creating the application; otherwise, the dimensions will not be available for the Application Wizard to select.

2 **From EPM Workspace, select Navigate, then Administer, and then Application Library.**

The Application Library is displayed.

3 **Select File, then New, and then Application**

The first screen of the wizard is displayed - Application Type.


ORACLE
Enterprise Performance Management System Workspace, Fusion Edition
Logged in as admin | Help | Logout
Navigate File View Tools Help
Dimension Library
Application Library
Application Creation Wizard
Application Type > Dimension Selection > Application Settings
Application Information
Help
Back
Next
Finish
Cancel

Name: NewApp

Type: Profitability

Description:

Create Blank Application: ☐

Auto Create Local Dimensions: ☐

4 **Under Name, enter the application name.**

Names must be 7 characters or less, and must not contain special characters, including “&” (ampersands). See “Essbase Naming Restrictions for Applications and Databases” on page 104

5 Under **Type**, select **Profitability.**

6 **Optional:** Under **Description**, enter a description.

7 **Optional:** To manually add dimensions to the blank application, select **Create Blank Application**, and then click **Finish.** See “Adding Dimensions Manually ” on page 56.

8 **Optional:** To automatically create all required dimensions, select **Auto Create Local Dimensions**, and then click **Next.** See “Selecting Dimensions” on page 52.

Selecting Auto Create Local Dimensions automatically creates new dimensions for all dimensions that are required in the application. The dimension name for each new dimension is identical to the dimension type with (New) in parentheses.

The second screen of the wizard is displayed - Dimension Selection.

9 Click **Next.** See “Selecting Dimensions” on page 52.

Selecting Dimensions

After creating the application, you must select the dimensions that are to be included in the application.

The following dimensions are required for Profitability and Cost Management applications:

- Measures dimension contains the members required to build, validate and calculate a model, including driver measures, reporting measures, and allocation measures. Users may add user-defined driver measures (or sub-hierarchies) to the hierarchy under the member 'UserDefinedDriverMeasures'.
- AllocationType dimension is used to store direct allocations and genealogy allocations.
- POV dimensions provide a specific view of your model information for a selected time period, such as a year, status and scenario. At least one Point of View (POV) dimension must be defined by the user. You can also create a POV Version.
- At least one Business Dimension must be defined by the user. Business, or user-defined, dimensions contain members that store information that is specifically related to the requirements of your business or organization, such as product types, sales regions, manufacturing processes, general ledger, payroll, departments, and so on. This dimension type does not apply to aggregate storage outlines.

  Business dimensions may include some or all of the following dimension types, and may apply to one or more stages or models:

  - Generic
  - Account
  - Entity
  - Time

    - Country

    Note: Although these business dimensions can be included as part of a Profitability and Cost Management application, when the Essbase outlines are deployed, they are created as basic or generic dimensions, with no type.

- Attribute dimensions are a special type of dimension that are associated with a business dimension. Attributes describe characteristics of data, such as the size or color of products.

  There are two Attribute dimension types in Performance Management Architect:

    - Attribute dimensions:
        - The attribute can be created using different structures, such as Boolean, Date, Numeric and Text.
        - An attribute has a hierarchy, and the hierarchies can be rolled up for aggregate values.
        - Only one attribute can be associated with one member.
    - User-defined attribute dimensions (UDAs):
        - The attribute can only be created using Text.
        - A UDA does not have a hierarchy, and cannot be easily used in reports to generate sums.
        - Multiple UDAs can be associated with a member.

  Each type of attribute dimension offers different advantages, depending on the requirements for your model and reporting needs. For detailed information about working with attribute dimensions, see *Oracle Essbase Database Administrator's Guide*.

- Alias is optional, and only required if you want to use aliases in your model. For naming conventions, see "Essbase Naming Restrictions for Dimensions, Members, and Aliases" on page 105.

  Caution! Duplicate member names or aliases are not allowed within the same dimension.

Note: Model stages are not dimensions, and are therefore not available in Performance Management Architect. Stages are added to a model in Profitability and Cost Management, and are used to organize dimensions into logical steps or stages.

After creating the application, select the dimensions to be added to the application using the appropriate procedure:

- If you are using the wizard, see "Adding Dimensions Using the Wizard" on page 54.
- If you created a blank application, see "Adding Dimensions Manually " on page 56.

Adding Dimensions Using the Wizard

When you use the application wizard, all required dimensions are automatically displayed. If there is an exact match, it automatically populates the dimension column for the dimension type. The required dimension types for Profitability and Cost Management are automatically categorized and displayed with a shaded heading:

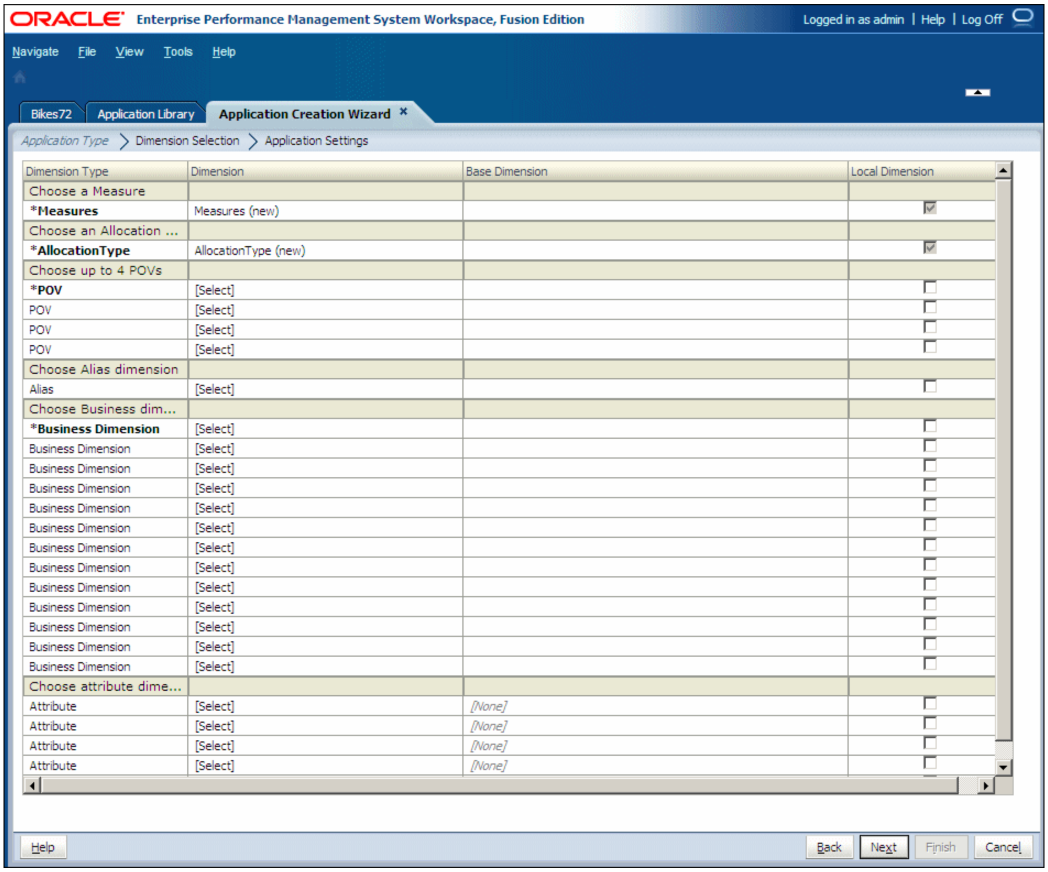
- Measures Dimension
- AllocationType Dimension
- POV Dimension
- Alias Dimension (Optional)
- Business Dimensions
- Attribute Dimensions (Optional)

If you selected Auto Create Local Dimensions when selecting the application type, new local dimensions are created for each required dimension. The name of each new dimension is the same as the dimension type, with (New) in parentheses. For example, Account (New).

Business dimensions you want to include in the application, for example, Account, Entity, Time, or Country, must be added to the Dimension Library before creating the application; otherwise, the dimensions will not be available for the Application Wizard to select. To view restricted words and characters for names, see Appendix C, “Essbase Naming Conventions”.

➤ To define dimensions:

1 On the Dimension Selection tab (Step 2 of the Application Wizard), under **Dimension Type**, review the required dimension types.


ORACLE Enterprise Performance Management System Workspace, Fusion Edition
Logged in as admin | Help | Log Off
Navigate File View Tools Help
Bikes72
Application Library
Application Creation Wizard ×
Application Type > Dimension Selection > Application Settings
Help
Back
Next
Finish
Cancel

Dimension Type	Dimension	Base Dimension	Local Dimension
Choose a Measure			
***Measures**	Measures (new)		☑
Choose an Allocation ...			
***AllocationType**	AllocationType (new)		☑
Choose up to 4 POVs			
***POV**	[Select]		□
POV	[Select]		□
POV	[Select]		□
POV	[Select]		□
Choose Alias dimension			
Alias	[Select]		□
Choose Business dim...			
***Business Dimension**	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Business Dimension	[Select]		□
Choose attribute dime...			
Attribute	[Select]	*[None]*	□
Attribute	[Select]	*[None]*	□
Attribute	[Select]	*[None]*	□
Attribute	[Select]	*[None]*	□

2 **For each existing dimension to be added for the application, under the Dimension column, click [Select] to display the drop-down list of available dimensions for that dimension type.**

Only the dimensions for the associated dimension type are displayed. For example, if you are mapping a dimension to the Measures dimension type, only measure dimensions display in the list.

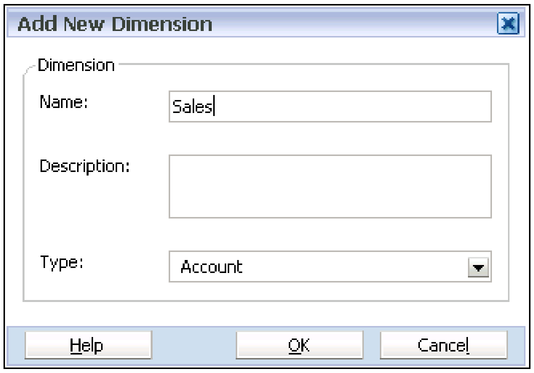
3 **Select a dimension from the drop-down list.**

4 **Optional: Create a new dimension. To create a new dimension:**

a. In the **Dimension** column, click **[Select]** to display the drop-down list of available dimensions for that dimension type.

b. From the drop-down list, select **[Create New Dimension]**.

The Add New Dimension dialog box is displayed.



Name: Sales

Description:

Type: Account

c. Enter the **Name** and **Description** of the new dimension. The **Type** is automatically selected, based on the dimension type.

d. Click **OK**.

5 **Repeat step 3 and step 4 to select or create new dimensions.**

Note: All required dimensions must be mapped to a dimension or [None] before you can move to the next screen.

6 **Optional: To designate the dimension as a local dimension, select the Local Dimension check box at the end of each row.**

For information on shared and local dimensions, see "Selecting Dimensions" on page 52.

7 **Click Next.**

The Application Settings screen is displayed.

8 **Modify the application settings, as required. See "Modifying the Application Settings" on page 57.**

Adding Dimensions Manually

When you create blank applications, you use the Dimension Library to select dimensions.

The Dimension Library has three panes:

- Shared Library—Displays on the left and displays all dimensions currently available in the Shared Library.
- Application—Displays in the middle and, when expanded, lists the names of all dimensions in the current application.
- Property Grid—Displays on the right and displays the properties of the selected dimension or member.

For detailed information on using the Dimension Library, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*

➤ To add dimensions to blank applications:

1 **In the Dimension Library, drag the dimensions from the Shared Library to the application.**

The Copy Dimension dialog box is displayed.

Tip: To create new local dimensions, right-click the application name in the Application View, and select Create Dimension. The Add New Dimension dialog box is displayed.

2 **Select Local Dimension or Shared Dimension.**

Tip: To add only members to an application, in the Shared Library, right-click the dimension and select **Add to App View**.

3 **Add all required dimensions to the application. (Measures, AllocationType, POV, and so on.)**

The expanded application lists all dimensions.

Tip: If you add a member to the Shared Library and include the dimension in the application, click the Refresh button 🔄, and the new member will display in the application.

4 **Set the Consolidation Property Value for NoMember to ~ (Ignore).**

5 **Reorder the dimensions so that `NoMember` is moved to the last Gen2 position. See "Reordering Children Members" on page 59.**

6 **Modify the application settings, as required. See "Modifying the Application Settings" on page 57.**

Modifying the Application Settings

The next step in creating an application is to modify application properties, associations, and other application settings. If you are creating a blank application and using the Dimension Library to modify the application settings, you have similar options, except that you are using the Dimension Library instead of the wizard to make the modifications.

Note: Dimension associations are required for shared hierarchies, and may not be required in all models.

See the following procedures to modify application settings:

- Use the Dimension Library for blank applications. See "Modifying Application Settings Using the Dimension Library" on page 57.
- Use the Application Wizard for auto-generated applications. See "Modifying Applications Settings Using the Application Wizard" on page 58.

Modifying Application Settings Using the Dimension Library

In the Dimension Library, the Shared Library, Application and Property Grid are displayed. Use the right-click menu to modify the application dimensions and perform these actions:

- For a Profitability and Cost Management model, perform the following tasks:
    - Set the Properties for POV Dimensions, and the POV Display Order for multiple POV dimensions, if required. See "Setting the Properties for POV Dimensions" on page 59.

    - Set the Dimension Sort Order for all model dimensions. See "Setting the Dimension Sort Order" on page 40.
    - Reorder the `NoMember` to display this member as the last generation 2 member on the list. See "Reordering Children Members" on page 59
- Modify dimensions, as required, using the procedures provided in the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*:
    - Set the dimension alias.
    - View application membership.
    - View dimension associations.
    - Synchronize dimensions.
    - Create members.
    - Insert shared members.
    - Manage orphan members.

Modifying Applications Settings Using the Application Wizard

If you are creating the application using the wizard, the application dimensions and members display on the left, the Property Grid displays on the right, and the validation and deployment options display at the bottom.

Use the right-click menu to modify the application dimensions, and perform these tasks: :

- For a Profitability and Cost Management model, perform the following tasks:
    - Set the Properties for POV Dimensions, and the POV Display Order for multiple POV dimensions, if required. See "Setting the Properties for POV Dimensions" on page 59.
    - Set the Dimension Sort Order for all model dimensions. See "Setting the Dimension Sort Order" on page 40.
    - Reorder the `NoMember` to display this member as the last generation 2 member on the list. See "Reordering Children Members" on page 59
- Modify dimensions, as required, using the procedures provided in the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*:
    - Set the dimension alias.
    - View application membership.
    - View dimension associations.
    - Synchronize dimensions.
    - Create members.
    - Insert shared members.
    - Manage orphan members.

Setting the Properties for POV Dimensions

At least one POV, or Point of View, dimension must be set for every Profitability and Cost Management model. The POV dimension may be set to any custom value, but usually denotes time periods, such as Year, Period, or Scenario. You can also create a POV Version.

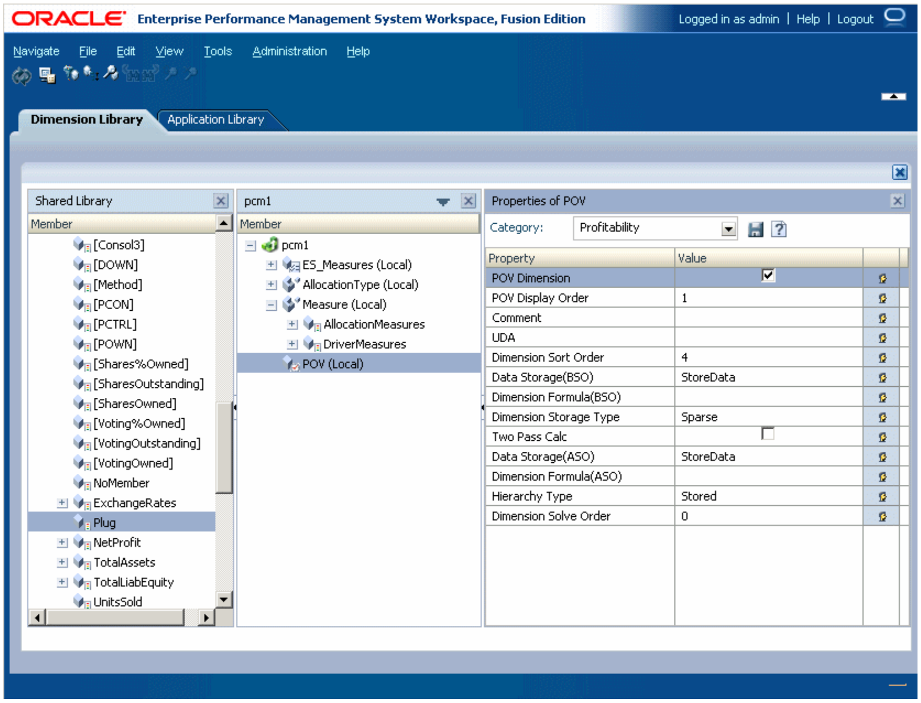
If more than one POV dimension is designated, a POV Display Order must also be set to sequence these dimensions for calculation. The POV Display Order that is set in Performance Management Architect is automatically picked up in Profitability and Cost Management.

➤ To set POV dimension properties:

1 **In the Dimension Library, select the first POV dimension, such as Year.**

2 **In the Property Grid under Category, select Profitability.**

3 **Select POV Dimension.**



Category: Profitability

Property	Value
POV Dimension	☑
POV Display Order	1
Comment	
UDA	
Dimension Sort Order	4
Data Storage(BSO)	StoreData
Dimension Formula(BSO)	
Dimension Storage Type	Sparse
Two Pass Calc	□
Data Storage(ASO)	StoreData
Dimension Formula(ASO)	
Hierarchy Type	Stored
Dimension Solve Order	0

4 **Optional: Repeat step 1 to step 3 to select other POV dimensions, such as Period or Scenario.**

5 **Optional: If more than one POV Dimension was selected, set the POV Display Order for each POV Dimension.**

Double-click the POV Display Order cell and type the sequence number for the selected POV dimension. For example, if Year, Period and Scenario are set as POV Dimensions, set the POV Display Order for Year to 1, Period to 2, and Scenario as 3 as the display order.

Reordering Children Members

For all business dimensions in the model, you must reorder the list of members to display `NoMember` as the last Gen2 member; otherwise, validation of the model will fail.

You can also reorder members to suit your particular requirements as long as `NoMember` remains in the last position.

Note: This step is not required for system dimensions, such as Alias, AllocationType, Measures, Periods, Scenarios or Years.

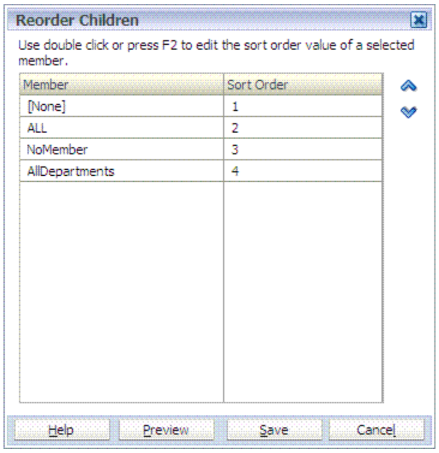
➤ To reorder children members:

1 **From EPM Workspace, select Navigate , then Administer, and then Dimension Library to display the Shared Library.**

2 **Select the dimension to be reordered, as described below:**

- For Shared dimensions, under **Shared Library**, right-click the dimension that you want to reorder and then select **Reorder Children**.
- For Local dimensions, under the Application column, right-click the dimension that you want to reorder and then select **Reorder Children**.

The Reorder Children dialog box is displayed.

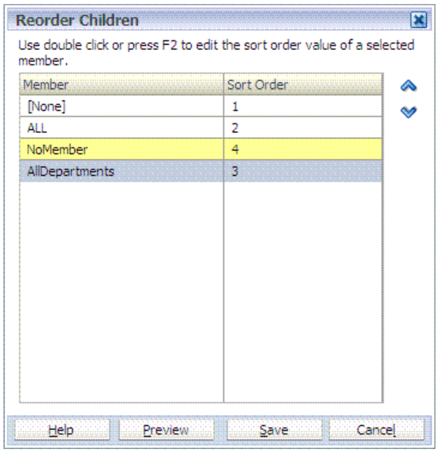


Reorder Children

Use double click or press F2 to edit the sort order value of a selected member.

Member	Sort Order
[None]	1
ALL	2
NoMember	3
AllDepartments	4

3 **Under Sort Order, double-click the cell for which the member is to be reordered, and enter the number of the new sort order location.**



Reorder Children

Use double click or press F2 to edit the sort order value of a selected member.

Member	Sort Order
[None]	1
ALL	2
NoMember	4
AllDepartments	3

Note: Ensure the `NoMember` dimension is moved to the end of the list as the last Gen2 member; otherwise, the deployment will fail.

4 Repeat step 3 to order all members.

5 To preview the changes, click **Preview**.

6 Click **Save**.

Validating and Deploying the Application in Performance Management Architect

After creating a Profitability and Cost Management application in Performance Management Architect, the application must be validated and deployed to Profitability and Cost Management.

See these sections:

- "Profitability and Cost Management Validations" on page 61
- "Validating and Deploying Applications Using the Wizard" on page 62
- "Validating and Deploying Applications through Performance Management Architect" on page 63

Note: You can add, rename, reparent, or delete individual members, multiple members, or combinations of members during redeployment of an application.

For detailed instructions on working with Performance Management Architect to create and deploy Profitability and Cost Management applications, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.

Profitability and Cost Management Validations

For Profitability and Cost Management, the following conditions are validated:

Table 11 Profitability Validations

Validation Level	Validations
Application	● The name of the application must be seven characters or fewer, and cannot contain special characters. ● At least one dimension must be set to POV type. Up to four dimensions may be marked as POV dimensions. Only one occurrence of each POV dimension class is allowed. ● At least one business dimension must be defined. ● At least one Measures dimension must be defined. ● At least one AllocationType dimension must be defined. ● There is only one dimension of type "Account." ● There is only one dimension of type "Entity." ● Application names do not include Essbase special characters and reserved words.

Validation Level	Validations
Dimension	● Root members of business dimensions must have the ASO and BSO data storage set to **LabelOnly**. ● Dimension Sort Order is set for all dimensions in the model, excluding Alias and UDA dimensions, and satisfies the following conditions: ❍ A dimension sort order must be set for every dimension in the model, except Alias and UDA dimensions. **Note:** The Alias and UDA dimensions are ignored for Dimension Sort Order. ❍ The dimension sort order must be sequential. ❍ Measures dimension is set to 1 by default. ❍ AllocationType dimension is set to 2 by default. ❍ POV and business dimensions are set to 3 or higher. ❍ Attribute dimensions are sorted as the last dimensions. For example, if you have four attribute dimensions in a sequence of 12 dimensions, the attribute dimensions must be set as 9, 10, 11, and 12. ● Duplicate members do not exist in the same dimension. ● POV dimensions must have at least one member. ● Attribute dimensions must satisfy the following conditions: ❍ Only a Level-0 member from the attribute dimension may be assigned as an attribute. ❍ Attributes may be assigned only to members at the same level in the base dimension. ❍ Attribute dimensions can only be associated with sparse dimensions.
Member	● Allow only ASO and BSO data storage to be defined. ● No shared members are allowed in the first Gen 2 member. ● A Shared Member must always appear after its corresponding Base Member in the outline order. ● `NoMember` must be set as the last generation 2 member for all business dimensions and must be set to Ignore (~) in the Property Grid. **Note:** This requirement does not apply to POVs, Measures, AllocationType, Alias, UDA, or Attribute dimensions. ● Shared members must reside in a Dynamic hierarchy, using one of the following methods: ❍ The Gen 1 member is `HierarchyType=dynamic` ❍ The Gen 1 member is `HierarchyType=HierarchiesEnabled`, and the Gen 2 ancestor of the shared member is `HierarchyType=dynamic` ● No duplicate member names or aliases exist for any members within the dimension. ● Member names do not include Essbase special characters and reserved words.

Validating and Deploying Applications Using the Wizard

If you are using the application wizard, validate the application and correct any errors prior to finalizing the creation of the application. If required, you can bypass the validation and deployment options in the wizard and later use the deployment option in the Application Library.

➤ To validate and deploy an application using the wizard:

1 On the **Application Settings** screen of the Application wizard, click **Validate**.

All errors display in the grid, showing the error types and error message.

2 **Optional:** To deploy the application after validation, select **Deploy when finished.**

3 If validation errors occur, correct any problems before deploying the application.

4 Click **Finish.**

If you selected "Deploy when finished," the Deploy dialog box is displayed.

Note: If you change a shared dimension, all applications that use that shared dimension are affected. For changes to take effect, the application must be redeployed.

5 Deploy the application. See "Validating and Deploying Applications Using the Wizard" on page 62.

Validating and Deploying Applications through Performance Management Architect

➤ To validate and deploy a Profitability and Cost Management application in Performance Management Architect:

1 From EPM Workspace, select **Navigate**, then **Administer**, and then **Application Library** to display the Application Library.

2 Right-click the application name, and select **Validate** to validate the newly created Profitability and Cost Management application in Performance Management Architect. For Profitability and Cost Management validation conditions, see "Profitability and Cost Management Validations" on page 61.

When the validation is complete, the Job Task dialog box is displayed, showing the associated job number.

3 Click the link for the associated job to view the Job Console.

4 Review the Job Console Summary, and if there are errors, click the validation log link under **Attachments** to see a complete listing of all validation errors or messages.

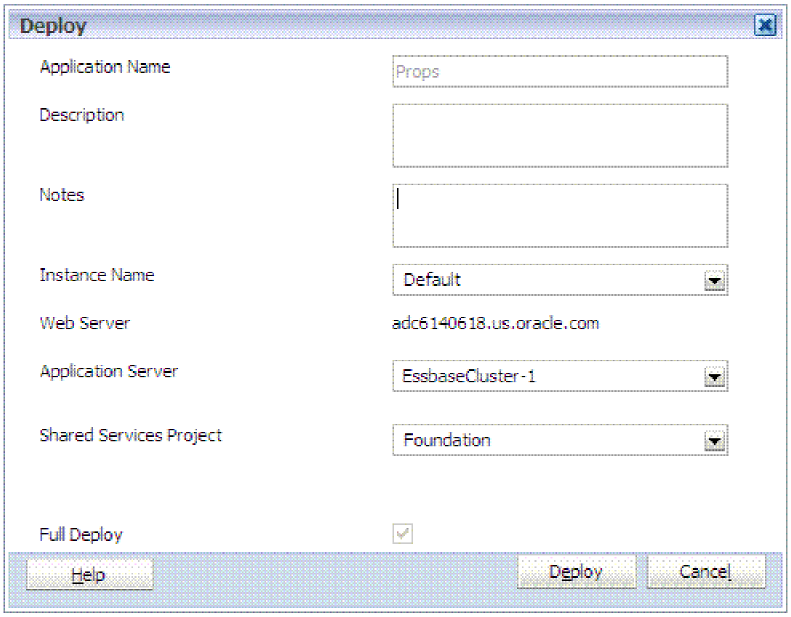
5 Fix any errors listed in the Validation Log.

6 Repeat step 2 to step 5 until the validation succeeds with no errors.

The application is ready to be deployed.

7 In the Application Library, right-click the application name, and select **Deploy** from the drop-down list.

Applications are validated before they are deployed. Any validation errors display in the Job Console. You must correct all validation errors before deploying. When the validation is complete, the Deploy dialog box is displayed.



Application Name: Props

Description:

Notes:

Instance Name: Default

Web Server: adc6140618.us.oracle.com

Application Server: EssbaseCluster-1

Shared Services Project: Foundation

Full Deploy: ☑

8 **Provide the required information for the deployment:**

- **Optional:** Under **Notes**, add notes pertaining to the deployment of this application. These notes are appended to the job description in the Job Console. This text field allows up to 200 characters and may contain any characters.
- Under **Instance Name**, select the name of the Profitability and Cost Management installation to which you want to deploy the application.

  A Profitability and Cost Management application may be installed on a single machine, or on more than one machine as part of a cluster. Each installation, or instance, is displayed on the Instance Name drop-down list. Default is the name of the instance used by the Configurator Tool for the first installation.
- Select the **Application Server.**
- Under **Shared Services Project**, select the Profitability and Cost Management application group to which this application is to be assigned.
- **Optional:** Decide whether **Full Deploy** should be selected, as follows:
  - If this is the first deployment of the application, the **Full Deploy** checkbox is disabled because Performance Management Architect automatically sends all dimensions to Profitability and Cost Management.
  - If **Full Deploy** is checked for any subsequent deployment, then Performance Management Architect sends all dimensions to Profitability and Cost Management.
  - If **Full Deploy** is not checked, then Performance Management Architect sends only modified dimensions to Profitability and Cost Management. If no changes were made to a dimension, then that dimension is not included in the deployment.

9 **Click Deploy.**

The Job Task window is displayed, to confirm the Validation Job has been submitted, and display the Job ID.

10 **On the Job Task window, click the link to display the job status.**

When the job is complete, a message is displayed under **Detail** to indicate the new application has been created and deployed to Profitability and Cost Management. The new application is available for selection.

Note: If the deployment of the application is not successfully completed, from the Application Library, right-click the application name, and select **Application Cleanup**. This option forces a delete call to both Shared Services and Profitability and Cost Management to remove any traces of the application, and resets the application in Performance Management Architect to the undeployed state.

Synchronizing Data

Data synchronization enables you to synchronize and map data between Oracle applications, interface tables, and external files.

Using the Data Synchronization module in Performance Management Architect, you can synchronize data between Oracle Hyperion Financial Management, Fusion Edition, Planning, Profitability and Cost Management, Essbase (ASO and BSO) as destinations, and the following sources:

- Oracle Hyperion Financial Management, Fusion Edition
- Planning
- Profitability and Cost Management
- Essbase (ASO and BSO)
- External source (flat file)
- Interface table

Data synchronization currently supports data transfer from BSO databases only.

Note: The Essbase application and database names should be the initial names given after the first Profitability and Cost Management application deployment. The database name should consist of the *application name* + “C” for the BSO database. If the names are different, data synchronization will fail.

See *Oracle Hyperion Enterprise Performance Management Architect Batch Client User's Guide* for detailed instructions on synchronizing data.

5 Importing Data into Profitability and Cost Management

In This Chapter

You can enter data directly into Profitability and Cost Management; however, the data entry may be very time-consuming. To facilitate the population of the application you can import data directly into Profitability and Cost Management using a set of import staging tables and import configurations.

Model data is imported from several sources:

- Model metadata and dimensions are imported from the Performance Management Architect Dimension Library through EPM Workspace. See the chapter on importing metadata in the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.
- Model definition data is imported into Profitability and Cost Management. The data may be created in another application, such as Excel, and imported using the staging tables. See Appendix A, “Import Staging Tables”.
- Model and end-user data can be imported and exported from Essbase. See the *Oracle Essbase Database Administrator's Guide*.
- Model data and applications can be imported using Lifecycle Management. See the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*.

Caution! Oracle recommends that, before importing data, you create a backup directory of your databases in EPM Workspace and Essbase. See Chapter 7, “Backing Up Profitability and Cost Management Components”.

You need to create an import configuration to specify which tables and data are to be imported. The configuration may be saved, and used multiple times to import the same set of data.

For detailed instructions, refer to these sections:

About Staging Tables

When data is imported into Profitability and Cost Management, users create the staging tables that provide the predefined structure to manage the import.

To import model data from relational databases into Profitability and Cost Management, you must create a set of staging tables in a separate database schema from the location in which the Profitability and Cost Management database tables were created to format the information for use in the application.

Caution! Do not create import staging tables inside the product schema. Modifications of the product schema are not only unsupported, but can produce unpredictable results.

The tables are created using a relational database, such as Oracle or SQL Server, to organize the data into a format that can be easily matched to the application. Staging database scripts are available for MS SQL Server and Oracle Database. Use the `create_staging.sql` script to create the import tables in the new database.

You can create all tables simultaneously, or create only the tables that you want to import. You must populate at least one of the following data groups:

- Stages
- POV
- Driver
- Driver Selection
- Assignment
- Assignment Rule Selection

The staging tables are created by the Profitability and Cost Management administrator (*admin*), using the format specified in Appendix A, “Import Staging Tables”.

Tables can be created manually or by running the SQL script `create_staging.sql`. Separate versions of this script are available for SQL and Oracle. After installation, the SQL scripts are included in the application folder. By default, the location is `%`*`EPM_ORACLE_HOME`*`%\products\Profitability\database\Common\Oracle`.

To import data, you must have the appropriate user role and security authorization. See the *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*.

Creating Import Configurations

Data can be imported only if all of the following elements are available:

- A source database and its associated staging tables and data
- A target application to receive the data
- An import configuration to define which tables and data are to be imported. The import configuration can be used multiple times.

To streamline importing, consider creating separate configurations for different sets of information. By using smaller import groups, you can reduce import times, and avoid repetitive updates of static information. For example, you might group model elements for an import configuration, as follows:

- Infrequently changed: POV, and stages
- Frequently changed: Drivers, driver selections, and assignments.

After a configuration is created, run the configuration to import data into the application.

➤ To create import configurations:

1 **Select a database to be used as the source for the import.**

   You can use an existing database that has been formatted for the import, or create a new, blank database.

   There are no naming restrictions for the database.

   Caution! Oracle recommends that, before importing data or artifacts, you create a backup directory of your databases in EPM Workspace and Essbase. Contact your administrator for assistance.

2 **Run the `create_staging.sql` script against the source database to automatically create the staging tables. The tables are generated using the schema provided in Appendix A, "Import Staging Tables".**

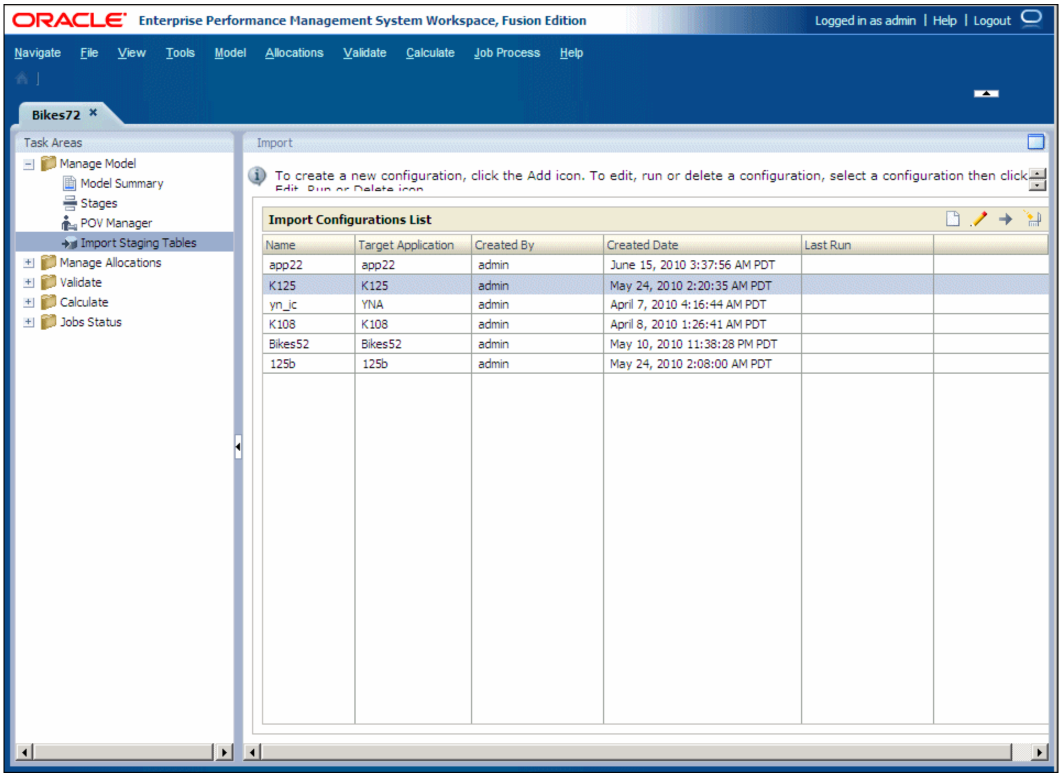
3 **Load the model data into the source staging tables.**

4 **Review the entries in the staging tables for obvious issues, and remove null rows from the source database.**

5 **Verify that the source database is accessible.**

6 In EPM Workspace, select **Navigate**, then **Applications**, then **Profitability**, and then the model name to access the Profitability and Cost Management model.

7 From **Task Areas**, select **Manage Model**, then **Import Staging Tables**.


ORACLE Enterprise Performance Management System Workspace, Fusion Edition
Logged in as admin | Help | Logout
Navigate File View Tools Model Allocations Validate Calculate Job Process Help
Bikes72
Task Areas
Manage Model
Model Summary
Stages
POV Manager
Import Staging Tables
Manage Allocations
Validate
Calculate
Jobs Status
Import

To create a new configuration, click the Add icon. To edit, run or delete a configuration, select a configuration then click Edit, Run or Delete icon.

Import Configurations List

Name	Target Application	Created By	Created Date	Last Run
app22	app22	admin	June 15, 2010 3:37:56 AM PDT	
K125	K125	admin	May 24, 2010 2:20:35 AM PDT	
yn_jc	YNA	admin	April 7, 2010 4:16:44 AM PDT	
K108	K108	admin	April 8, 2010 1:26:41 AM PDT	
Bikes52	Bikes52	admin	May 10, 2010 11:38:28 PM PDT	
125b	125b	admin	May 24, 2010 2:08:00 AM PDT	

8 From **Import**, click the **Add new import configuration** button 🗋 to create a new import configuration.

Step 1 of the Import Data dialog box is displayed.

9 Under **Data Source Details**, enter the access details for the server on which the source staging tables reside:

- From **Database Type**, select the type of database being used for the source database.
- Under **Host**, enter the path to the location of the source database that contains the source staging tables.
- Under **Port**, enter the name of the port associated with the source database.
- Under **Database Name**, enter the name of the source database.

10 Under **Credentials**, enter the **User Name** and **Password** for the authorized user of the source database, if required.

11 Click **Next.**

Step 2 of the Import Data dialog box is displayed.

12 Under **Configuration**, select the **Target Application.**

The target application is the Profitability and Cost Management application into which the data is being imported.

13 Under **Staging Tables Details**, select one or more staging tables to be imported.

The name of existing staging table is listed under Table Name, and the type of data in each staging table is shown under Table Type.

14 Click **Next.**

Step 3 of the Import Data dialog box is displayed.

15. Under **Configuration Summary**, enter a **Configuration Name** to store this import configuration.

    The database name associated with this configuration is displayed below the configuration name. The configuration name cannot contain more than 80 characters.

16. Click **Finish**.

    The import configuration is added to the Import Configurations List, and can be reused to import the same set of data many times.

17. **Optional**: From the Import Configurations List, select the import configuration, and click the **Run import configuration** button → to run the import configuration. See "Running Import Configurations " on page 72.

Modifying Import Configurations

When updating an import configuration, the target application, set of staging tables and import configuration name can be changed.

➤ To modify import configurations:

1. From **Task Areas**, select **Manage Model**, then **Import Staging Tables.**

   The Import page is displayed.

2. From the **Import Configurations List**, select the import configuration to be modified.

3. Click the **Edit import configuration** button ✏.

   The Import Data dialog box is displayed.

4. Review the connection information and credentials for the selected import configuration, and then click **Next.**

5. **Optional:** Under **Target Application**, select a target application for this import configuration.

6. **Optional:** Under **Staging Tables Details**, change the set of staging tables to be imported.

7. Click **Next.**

8. **Optional:** Under **Configuration Summary**, enter a different **Configuration Name** to store the modified configuration, and review the connection URL.

9. Click **Finish.**

   The modified configuration is stored, and can be run at any time. See "Running Import Configurations " on page 72.

Deleting Import Configurations

➤ To delete import configurations:

1 From **Task Areas**, select **Manage Model**, then **Import Staging Tables.**

The Import page is displayed.

2 Select the configuration to be deleted, and click the **Delete import configuration** button .

A message asks you to confirm the deletion.

3 Click **Yes.**

The configuration is deleted.

Running Import Configurations

After an import configuration has been created, you can run the configuration to import data into the application.

➤ To run an import configuration:

1 Ensure that you have the connection information for the relational database on which the staging tables reside.

2 From **Task Areas**, select **Manage Model**, and then **Import Staging Tables**

The Import page is displayed.

3 From the **Import Configurations List**, select the import configuration that you want to run.

4 Click the **Run import configuration** button .

The Import Data dialog box is displayed. All data connection details are listed.

5 Click **Next.**

6 Under **Configuration**, verify that the target database and selected staging tables are correct.

7 Click **Next.**

The Configuration Summary is displayed. The Configuration Name and connection URI are identified.

8 Select one of the following operations:

- Click **Run Now** to run the import immediately.
- Click **Run Later** to schedule a more convenient date and time to run the import.

9 Click **Finish.**

- If you selected **Run Now**, the import runs and populates the target application with the selected data.

- If you selected **Run Later**, the job is saved. To run the saved import job, select Job Process, then Manage Taskflow. From this screen, you can run the job, or schedule it to run at a more convenient date and time. See the *Oracle Hyperion Profitability and Cost Management User's Guide* for detailed instructions.

10 When the import is complete, verify the imported data. See "Verifying Imported Data" on page 73.

Verifying Imported Data

After importing the data into the target database, you must verify that the data has been imported correctly and completely.

➤ To verify imported data:

1 In EPM Workspace, select **Navigate**, then **Applications**, then **Profitability**, and then the application name to access the Profitability and Cost Management model.

2 In the target application, open the model, and review the data that was expected in the import.

   For example, if you imported Stages, select **Manage Model**, then **Stages** and ensure the information for all stages is present and correct.

3 Review the Exceptions column in each imported staging table to determine whether there are errors or warnings. See Appendix A, "Import Staging Tables" for a list of error messages for each table.

4 Correct errors in the source staging tables, and then rerun the import.

   Repeat step 1 through step 4 as required until no errors are generated during the import. For a listing of the error messages for each staging table, see the Appendix A, "Import Staging Tables".

6

Migrating Data Using EPM System Lifecycle Management

In This Chapter

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 Shared Services Console is consistent for all EPM System products that support Lifecycle Management. However, EPM System products display different artifact listings and export and import options in the Lifecycle Management interface.

You can select one or more of the following Profitability and Cost Management artifacts for migration through Lifecycle Management:

- Application preferences
- Driver definitions
- Stage definitions
- POV definitions
- Driver selections (Driver Rules and Driver Exceptions)
- Regular Assignments
- Assignment Rules
- Assignment Rule Selections

Using Lifecycle Management features, you can perform the following tasks:

- View applications and folders
- Search for artifacts
- Migrate directly from one application to another
- Migrate to and from the file system
- Save and load migration definition files

- View selected artifacts
- Audit migrations
- View 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 Oracle's Hyperion® 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*.

Using Lifecycle Management to Migrate Profitability and Cost Management Files to Another Environment

To migrate a complete Profitability and Cost Management application from one environment to another, you must perform three separate migrations to reconstruct the entire application:

- Performance Management Architect application, which contains the collection of dimensions in the EPM Architect dimension library
- Profitability and Cost Management Model (or Application)
- Essbase databases that contain the data and perform the calculations.

Caution! Managing Profitability and Cost Management application artifacts through Oracle Hyperion Enterprise Performance Management System Lifecycle Management is only supported within the same version of the product.

The following procedure provides the general steps required to migrate from one environment to another. For detailed instructions on performing the migration, see the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*.

➤ To migrate Profitability and Cost Management applications to another environment:

1 **Back up the destination environment before beginning the migration. See the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.**

2 **From EPM Workspace, select Navigation, then Administer, and then Shared Services Console.**

3 **Create the export migration definition for the Performance Management Architect metadata as follows:**

   a. From Hyperion Shared Services Console, select **Application Groups**, then **Foundation**, and then **EPM Architect**.

b. From the **Artifact list**, expand **Application Metadata**, then expand **Profitability Applications**, and then select the specific application to be exported.

   If required, expand the application and select one or more of the following elements for export:

   - Import Profile
   - Local Dimensions
   - Profitability Settings

c. Click **Define Migration**, and then complete the Migration wizard steps, as outlined in the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*, and using the following information:

   - Source is Foundation/EPM Architect
   - Destination is the File System

   Tip: Enter a folder name that identifies that the folder contains the Performance Management Architect metadata, for example: Bike_epma.

d. **Optional:** When the migration is complete, on the Shared Services Console, under **File System** check for the newly created Performance Management Architect migration folder.

4 **Create the export migration definition for the Profitability and Cost Management metadata as follows:**

a. From Hyperion Shared Services Console, select **Application Groups**, then the Default Application Group (or other Shared Services project) that contains the Profitability projects, and then the Profitability and Cost Management model. For example, Bike.

b. From the **Artifact List**, select the artifacts that you want to migrate, and then click **Define Migration**. For example, you may select individual artifacts, such as Assignment Rules or Drivers, or you can select all artifacts.

c. Complete the Migration wizard steps, as outlined in *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*, using the following information:

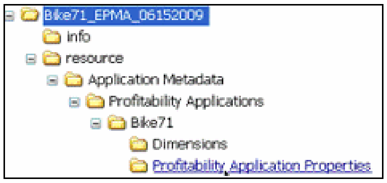
   - Source is HPM/*Model Name*
   - Destination is the File System

   Tip: Enter a folder name that identifies that the folder contains the Profitability and Cost Management metadata, for example: Bike_hpm.

5 **Navigate to the Windows File System to zip the migration files:**

a. From the Windows File System, navigate to the Performance Management Architect migration folder. By default, the migration folder is in `EPM_ORACLE_INSTANCE/import_export/user@Native Directory`.

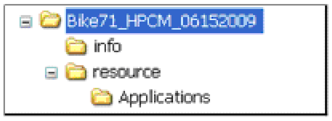
b. Select the Performance Management Architect migration folder (for example, Bike_epma), and zip the complete folder structure.



```
⊟ Bike71_EPMA_06152009
    info
  ⊟ resource
    ⊟ Application Metadata
      ⊟ Profitability Applications
        ⊟ Bike71
            Dimensions
            Profitability Application Properties
```

c. From the Windows File System, navigate to the Performance Management Architect migration folder. By default, the migration folder is located at `EPM_ORACLE_INSTANCE/import_export/user@Native Directory`.

d. Select the Profitability and Cost Management migration folder (for example, Bike_hpm), and ensure that all files within the migration folder are included in the ZIP file.



```
⊟ Bike71_HPCM_06152009
    info
  ⊟ resource
      Applications
```

6 **E-mail the Performance Management Architect and Profitability and Cost Management ZIP files to the appropriate party.**

7 **On the destination server, unzip the Performance Management Architect ZIP file to the same file structure (for example, unzip the file to** `EPM_ORACLE_INSTANCE/import_export/user@Native Directory`**) as follows:.**

a. From Hyperion Shared Services Console, select **File System**, and then the EPMA Migration file. For example, Bike_epma.

b. From the **Artifact List**, expand **Application Metadata**, then check **Profitability Applications**, and then click **Define Migration**.

c. Complete the Migration wizard steps, as outlined in the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*, using the following information:

- Source is File System
- Destination is the Foundation/EPM Architect

8 **From EPM Workspace, select Navigate, then Administer, then Application Library, and then select Deploy to deploy the Profitability and Cost Management application. For complete instructions, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator's Guide*.**

At this point, this is just a shell application. It does not contain any model structure or data.

9 **On the destination server, unzip the Profitability and Cost Management ZIP file to the same file structure (for example, unzip the file to** `EPM_ORACLE_INSTANCE/import_export/user@Native Directory`**) as follows:**

a. From Hyperion Shared Services Console, select **File System**, and then the Profitability and Cost Management Migration file. For example, Bike_hpm.

b. From the **Artifact List**, select the artifacts to be imported, and then click **Define Migration**.

c. Complete the Migration wizard steps, as outlined in the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*, using the following information:

   - Source is File System
   - Destination is the HPM/*Model Name*

10 Within Profitability and Cost Management, under Task Areas, select **Manage Model**, then **Model Summary**, and the **Model Level Preferences** tab and then select the Essbase host in the imported environment.

11 Within Profitability and Cost Management, under Task Areas, select **Calculate**, and then **Manage Database** to deploy the model to add the model data. For complete instructions, see the *Oracle Hyperion Profitability and Cost Management Administrator's Guide*.

12 Import model data from Essbase into the newly deployed model. See the *Oracle Essbase Database Administrator's Guide*.

The model is now complete and available for calculation.

Modifying Default Timeout Settings

If you are using Lifecycle Management to import large models, the import may take longer to process than the time specified in the default timeout settings on the Oracle WebLogic Server. To work around this issue, you must reset the default timeout settings.

➤ To modify the default timeout for WebLogic Server:

1 Navigate to `EPM_ORACLE_INSTANCE\httpConfig\OHS\ohs_component\mod_wl_ohs.conf`.

2 In the configuration file, locate the section **LocationMatch/profitability**, and add the line **`WLIOTimeoutSecs 3000`**, as shown below:

```
SetHandler weblogic-handler

PathTrim

KeepAliveEnabled ON

KeepAliveSecs 20

WLIOTimeoutSecs 3000

WeblogicCluster servername:6756
```

3 Navigate to `EPM_ORACLE_INSTANCE\httpConfig\OHS\ophs_component\htppd.conf`

4 Set **Timeout** to **3000**, as shown in the following text:

```
# Timeout: The number of seconds before receives and sends time out.
Timeout 3000
```

Note: The server timeout shown above is a suggested limit. Because timeout settings vary from application server to application server, refer to the timeout settings specific to your application server.

7 Backing Up Profitability and Cost Management Components

There are several Profitability and Cost Management databases that need to be backed up on a regular basis to ensure the safety and integrity of your data:

- Operational Data Store
- Import Staging Tables
- Block Storage Option (BSO) Database

The frequency of backup is determined by the amount of change in the databases, and the general standards of your organization.

Table 12, "Profitability and Cost Management Database Backup Requirements," on page 81, lists the databases that must be backed up, their type, and suggested backup options.

Caution! Lifecycle Management (LCM) cannot currently be used to back up the import staging area or the operational data store.

Table 12 Profitability and Cost Management Database Backup Requirements

Database	Database Type	Backup Options
Import Staging Area	Relational Database (RDB)	Standard backup techniques for any RDB can be used, including the following options: ● Database scripting ● Scheduler scripts ● Backup tools, such as TOAD ● Backup procedure from SQL Server or Oracle.
Operational Data Store	Relational Database (RDB)	Standard backup techniques for any RDB can be used, including the following options: ● Database scripting ● Scheduler scripts ● Backup tools, such as TOAD ● Backup procedure from SQL Server or Oracle.

Database	Database Type	Backup Options
Block Storage Option (BSO) and Aggregated Storage Option (ASO) Database and Assignment Rules	Essbase	Standard Essbase backup techniques should be used. For detailed backup and recovery procedures, see the *Oracle Essbase Database Administrator's Guide*.

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

A

Import Staging Tables

In This Appendix

Using Import Staging Tables

In order to import model data from relational databases into Profitability and Cost Management, you must create a set of staging tables in the source database to format the information for use in the application. Model data and metadata for the import are loaded into the staging tables in an Oracle or MS SQL Server database. To import data and metadata using these staging tables, see the *Oracle Hyperion Profitability and Cost Management User's Guide*.

The Profitability and Cost Management staging table schema has defined table names and contents.

Caution! Do not modify the staging table names, columns or overall table structure, as this may cause the import to fail.

A unique record ID is required for every staging table. This field is required, and is entered by the user, whether through an automatic program, script or manual entry.

Error messages for each table are self-explanatory, and are presented in the format `ERROR_<Error Condition>` For example, `ERROR_NAME_NOT_FOUND`, or `ERROR_DUPLICATED_DIMENSIONS`.

Use the schema tables in this appendix to create your staging tables:

- "Creating Import Database Tables" on page 84
- "HPM_STG_STAGE" on page 84
- "HPM_STG_POV" on page 85

- "HPM_STG_DRIVER" on page 87
- "HPM_STG_DRIVER_SELECTION " on page 89
- "HPM_STG_ASSIGNMENT" on page 90
- "HPM_STG_ASGN_RULE_SELECTION" on page 92

Creating Import Database Tables

The following tables are available in the Staging Database:

- Stages (HPM_STG_STAGE)
- POVs (HPM_STG_POV)
- Drivers (HPM_STG_DRIVER)
- Driver selections (HPM_STG_DRIVER_SELECTION)
- Assignments (HPM_STG_ASSIGNMENT)
- Assignment Rules (HPM_STG_ASGN_RULE_SELECTION)

Staging database scripts are available for MS SQL Server and Oracle Database. You can create a database and use the `create_staging.sql` script to create tables.

➤ To create staging tables:

1 **Create a database.**

2 **Locate** `create_staging.sql` **in the default location for your database type:**

   - `EPM_ORACLE_HOME/products/Profitability/database/Common/MSSQLServer`
   - `EPM_ORACLE_HOME/products/Profitability/database/Common/Oracle 2`

3 **Run** `create_staging.sql`**.**

HPM_STG_STAGE

The HPM_STG_STAGE table provides a list of the stages within the model, and identifies the display order, prefix, associated dimensions and driver dimension for each stage.

Dependencies: None

Table 13 HPM_STG_STAGE

Field	SQL Server Data Type	Oracle Data Type	Required	Description
id	identity	integer	Yes	Unique record ID
name	nvarchar (80)	varchar2 (80)	Yes	Name of the model stage.

Field	SQL Server Data Type	Oracle Data Type	Required	Description
description	nvarchar (255)	varchar2 (255)		Brief explanation of the purpose of the stage.
stage_order	integer	integer	Yes	The sequence position in which the selected stage is to be used within the model while modeling (assignments and rules creation), in calc script generation and calculation.
stage_prefix	nvarchar (80)	varchar2 (80)	Yes	Differentiating prefix for the stage name.
dim1_name	nvarchar (80)	varchar2 (80)	Yes	Name of the first dimension included in the stage.
dim2_name	nvarchar (80)	varchar2 (80)		Name of the second dimension included in the stage, if available.
dim3_name	nvarchar (80)	varchar2 (80)		Name of the third dimension included in the stage, if available.
driver_dim_name	nvarchar (80)	varchar2 (80)	Yes	Name of the dimension selected as the driver for the stage.
Is_intra	nvarchar (1)	varchar2 (255)	Yes	Enter “Y” (Yes) to enable intrastage assignments within the model stage. Default is “N” (No). This field is populated from the application.
last_upload_date	datetime	date		The date and time of the last import.
exception	nvarchar (255)	varchar2 (255)		Message detailing errors that occurred during the import of this table.
created_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who initiated the last import.
created_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the last import. This value is populated by the import program upon update.
modified_userid	nvarchar(32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who modified the last import.
modified_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the modified import.

HPM_STG_POV

The HPM_STG_POV table stores the states of each combination of dimension members included in a point of view (POV).

Dependencies: None

Table 14 HPM_STG_POV

Field	SQL Server Data Type	Oracle Data Type	Required	Description
id	identity	integer	Yes	Unique record ID
pov_dim1_ member_name	nvarchar (80)	varchar2 (80)	Yes	Name of the first POV dimension member.
pov_dim2_ member_name	nvarchar (80)	varchar2 (80)		Name of the second POV dimension member, if available.
pov_dim3_ member_name	nvarchar (80)	varchar2 (80)		Name of the third POV dimension member, if available. Additional dimensions and member names may be added, as required.
pov_dim4_ member_name	nvarchar (80)	varchar2 (80)		Name of the fourth POV dimension member, if available. Additional dimensions and member names may be added, as required.
pov_state	nvarchar (80)	varchar2 (80)	Yes	The current status of the POV: ● Draft ● Published ● Archived
last_upload_date	datetime	date		The date and time of the last import.
exception	nvarchar (255)	varchar2 (255)		Message detailing errors that occurred during the import of this table.
created_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who initiated the last import.
created_ timestamp	datetime	date	Populated by the import program upon update.	The date and time of the last import. This value is populated by the import program upon update.
modified_userid	nvarchar(32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who modified the last import.
modified_ timestamp	datetime	date	Populated by the import program upon update.	The date and time of the modified import.

HPM_STG_DRIVER

The HPM_STG_DRIVER table provides details about the driver for the selected stage, including driver type, formula and layer.

Dependencies: None

Table 15 HPM_STG_DRIVER

Field	SQL Server Data Type	Oracle Data Type	Required	Description
id	identity	integer	Yes	Unique record ID
name	nvarchar (80)	varchar2 (80)	Yes	Name of the driver.
description	nvarchar (255)	varchar2 (255)		Description of the purpose for the selected driver.
display_order	integer	integer		Display position of the driver within the list of all drivers in the model.
driver_type	nvarchar (80)	varchar2 (80)	Yes	Type of driver: ● Even ● Simple ● Simple Weighted ● Simple Variable ● Weighted Variable ● Fixed and Variable ● Percentage ● Custom
fixed_member	nvarchar (80)	varchar2 (80)		Member in the Measures dimension that corresponds to the standard driver measure “FixedDriverValue.”
fixed_location	nvarchar (30)	varchar2 (30)		Location of the standard driver measure “FixedDriverValue.”
rate_member	nvarchar (80)	varchar2 (80)		Member in the Measures dimension that corresponds to the standard driver measure “Rate.”
rate_location	nvarchar (30)	varchar2 (30)		Location of the standard driver measure “Rate.”
weight_ member	nvarchar (80)	varchar2 (80)		Member in the Measures dimension that corresponds to the standard driver measure “Weight.”
weight_ location	nvarchar (30)	varchar2 (30)		Location of the standard driver measure “Weight.”
volume_ member	nvarchar (80)	varchar2 (80)		Member in the Measures dimension that corresponds to the standard driver measure “Volume.”
volume_ location	nvarchar (30)	varchar2 (30)		Location of the standard driver measure “Volume.”

Field	SQL Server Data Type	Oracle Data Type	Required	Description
custom_ formula	nvarchar (500)	varchar2 (500)		Formula created for the driver by a user. This formula must be created using Essbase calculation script syntax. See the *Oracle Hyperion Profitability and Cost Management User's Guide*.
cost_layer	nvarchar (30)	varchar2 (30)		The driver layer is set to Cost layer, which contains the cost values for the model. **Note:** Either the cost_layer, revenue_layer, or both must be selected.
revenue_layer	nvarchar (30)	varchar2 (30)		The driver layer is set to Revenue layer, which contains the revenue values for the model. **Note:** Either the cost_layer, revenue_layer, or both must be selected.
allow_idle_ cost	nvarchar (1)	varchar2 (1)		Enter the appropriate value to determine whether idle costs are allowed for this driver: ● Enter “Y” (Yes) to allow idle cost for a driver. ● Enter “N” (No) to disable idle cost for a driver. This is the default.
driver_basis_ type	nvarchar (80)	VARCHAR2 (80)	Yes	Select the required driver basis: ● Actual Basis ● Standard Basis **Note:** The Standard Basis driver cannot be used with Even or Percentage driver types.
priority	integer	integer	No	Enter the calculation priority of a driver so that allocations within a stage can be run in the specified order. The driver with the lowest priority is processed first. By default, the priority is set to 100, and the highest priority is 1. Numbers need not be sequential. Drivers with the same priority are executed in no particular order. Only whole, positive numbers are valid.
last_upload_ date	datetime	date		The date and time of the last import.
exception	nvarchar (255)	varchar2 (255)		Message detailing errors that occurred during the import of this table.
created_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who initiated the last import.
created_ timestamp	datetime	date	Populated by the import program upon update.	The date and time of the last import. This value is populated by the import program upon update.

Field	SQL Server Data Type	Oracle Data Type	Required	Description
modified_ userid	nvarchar(32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who modified the last import.
modified_ timestamp	datetime	date	Populated by the import program upon update.	The date and time of the modified import.

HPM_STG_DRIVER_SELECTION

The HPM_STG_DRIVER_SELECTION table stores the drivers associated with the intersections of the exact stage.

Dependencies:

- HPM_STG_DRIVER
- HPM_STG_STAGE
- HPM_STG_POV

Table 16 HPM_STG_DRIVER_SELECTION

Field	SQL Server Data Type	Oracle Data Type	Required	Description
id	identity	integer	Yes	Unique record ID
pov_dim1_member_ name	nvarchar (80)	varchar2 (80)	Yes	Name of the first dimension member in the selected point of view.
pov_dim2_member_ name	nvarchar (80)	varchar2 (80)		Name of the second dimension member in the point of view, if applicable.
pov_dim3_member_ name	nvarchar (80)	varchar2 (80)		Name of the third dimension member in the point of view, if applicable.
pov_dim4_member_ name	nvarchar (80)	varchar2 (80)		Name of the fourth dimension member in the point of view, if applicable.
layer_name	nvarchar (80)	varchar2 (80)	Yes	Name of the selected layer for the point of view: ● Cost (Default) ● Revenue
stage_name	nvarchar (80)	varchar2 (80)	Yes	Name of the model stage for the selected point of view.

Field	SQL Server Data Type	Oracle Data Type	Required	Description
dim1_member_name	nvarchar (80)	varchar2 (80)	Yes, if driver dimension for the selected stage	Member name of the first dimension of the selected stage.
dim2_member_name	nvarchar (80)	varchar2 (80)	Yes, if driver dimension for the selected stage	Member name of the second dimension of the selected stage, if applicable.
dim3_member_name	nvarchar (80)	varchar2 (80)	Yes, if driver dimension for the selected stage	Member name of the third dimension of the selected stage, if applicable.
driver_name	nvarchar (80)	varchar2 (80)	Yes	Name of the driver for the selected intersection.
last_upload_date	datetime	date		The date and time of the last import.
exception	nvarchar (255)	varchar2 (255)		Message detailing errors that occurred during the import of this table.
created_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who initiated the last import.
created_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the initiated the last import. This value is populated by the import program upon update.
modified_userid	nvarchar(32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who modified the last import.
modified_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the modified import.

HPM_STG_ASSIGNMENT

The HPM_STG_ASSIGNMENT table provides details about each assignment, including the source stage, POV, driver layer, source dimension members, destination stage, and destination dimension members.

Dependencies:

- HPM_STG_POV
- HPM_STG_STAGE
- HPM_STG_DRIVER

Table 17 HPM_STG_ASSIGNMENT

Field	SQL Server Data Type	Oracle Data Type	Required	Description
id	identity	integer	Yes	Unique record ID

Field	SQL Server Data Type	Oracle Data Type	Required	Description
pov_dim1_member_name	nvarchar (80)	varchar2 (80)	Yes	Name of the first dimension member in the selected point of view.
pov_dim2_member_name	nvarchar (80)	varchar2 (80)		Name of the second dimension member in the point of view, if applicable.
pov_dim3_member_name	nvarchar (80)	varchar2 (80)		Name of the third dimension member in the point of view, if applicable.
pov_dim4_member_name	nvarchar (80)	varchar2 (80)		Name of the fourth dimension member in the point of view, if applicable.
layer_name	nvarchar (80)	varchar2 (80)	Yes	Name of the selected layer for the point of view: ● Cost (Default) ● Revenue **Note:** Either the cost_layer or the revenue_layer must be selected.
src_stage_name	nvarchar (80)	varchar2 (80)	Yes	Name of the source stage for the assignment.
src_dim1_member_name	nvarchar (80)	varchar2 (80)	Yes	Name of the first dimension member in the source stage.
src_dim2_member_name	nvarchar (80)	varchar2 (80)	Yes, if stage source/ destination stage have this dimension	Name of the second dimension member in the source stage, if available.
src_dim3_member_name	nvarchar (80)	varchar2 (80)	Yes, if stage source/ destination stage have this dimension	Name of the third dimension member in the source stage, if available.
dst_stage_name	nvarchar (80)	varchar2 (80)	Yes	Name of the destination stage for the assignment.
dst_dim1_member_name	nvarchar (80)	varchar2 (80)	Yes	Name of the first dimension member in the destination stage.
dst_dim2_member_name	nvarchar (80)	varchar2 (80)		Name of the second dimension member in the destination stage, if available.
dst_dim3_member_name	nvarchar (80)	varchar2 (80)		Name of the third dimension member in the destination stage, if available.
last_upload_date	datetime	date		The date and time of the last import.
exception	nvarchar (255)	varchar2 (255)		Message detailing errors that occurred during the import of this table.
created_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who initiated the last import.

Field	SQL Server Data Type	Oracle Data Type	Required	Description
created_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the initiated the last import. This value is populated by the import program upon update.
modified_userid	nvarchar(32)	varchar2 (32)	Populated by the import program upon update.	The ID of the user who modified the last import.
modified_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the modified import.

HPM_STG_ASGN_RULE_SELECTION

The HPM_STG_ASGN_RULE_SELECTION table stores details about the source stage and dimension members for a rule controlling the assignments for the selected stage.

Dependencies:

- HPM_STG_POV
- HPM_STG_STAGE
- HPM_STG_DRIVER

Table 18 HPM_STG_ASSGN_RULE_SELECTION

Field	SQL Server Data Type	Oracle Data Type	Required	Description
id	identity	integer	Yes	Unique record ID
pov_dim1_member_ name	nvarchar (80)	varchar2 (80)	Yes	Name of the first dimension member in the selected point of view.
pov_dim2_member_ name	nvarchar (80)	varchar2 (80)		Name of the second dimension member in the point of view, if applicable.
pov_dim3_member_ name	nvarchar (80)	varchar2 (80)		Name of the third dimension member in the point of view, if applicable.
pov_dim4_member_ name	nvarchar (80)	varchar2 (80)		Name of the fourth dimension member in the point of view, if applicable.
layer_name	nvarchar (80)	varchar2 (80)	Yes	Name of the selected layer for the point of view: ● Cost (Default) ● Revenue **Note:** Either the cost_layer, revenue_layer, or both is required.
src_stage_name	nvarchar (80)	varchar2 (80)	Yes	Name of the source stage for the assignment rule.

Field	SQL Server Data Type	Oracle Data Type	Required	Description
src_dim1_member_ name	nvarchar (80)	varchar2 (80)	Yes, if the source stage has the appropriate dimension.	Name of the first dimension member in the source stage.
src_dim2_member_ name	nvarchar (80)	varchar2 (80)	Yes, if the source stage has the appropriate dimension.	Name of the second dimension member in the source stage, if available.
src_dim3_member_ name	nvarchar (80)	varchar2 (80)	Yes, if the source stage has the appropriate dimension.	Name of the third dimension member in the source stage, if available.
rule_name	nvarchar (80)	varchar2 (80)	Yes	Name of the rule controlling the selected assignment. The rule must be present in the target database.
last_upload_date	date	date		Date of the last import.
exception	nvarchar (255)	varchar2 (255)		Message detailing errors that occurred during the import of this table.
created_timestamp	datetime	date	Populated by the import program upon update.	The date and time of the last import.
created_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	ID of the user who initiated the last import.
modified_userid	nvarchar (32)	varchar2 (32)	Populated by the import program upon update.	ID of the user who initiated the reimport.
modified_timestamp	date, time	date	Populated by the import program upon update.	The date and time of the reimport.

B Exporting Model Definition Data

In This Appendix

After the model has been created, you can query the database to display the model artifacts as output in a database view.

An Administrator can create database views in the system database that mirror the columns used in the staging tables, showing the model data that is stored in the system:

- Stages (HPM_EXP_STAGE)
- POVs (HPM_EXP_POV)
- Drivers (HPM_EXP_DRIVER)
- Driver Selections (HPM_EXP_DRIVER_SELECTION)
- Assignments (HPM_EXP_ASSIGNMENT)
- Assignment Rule Selections (HPM_EXP_ASGN_RULE_SELECTION)

After installation, a SQL script, `create_sql`, is included in the application folder. The default location is *`EPM_ORACLE_HOME`*`\products\Profitability\database\Common\Oracle2`. The script `create_sql`, which contains the SQL queries for artifacts within the model, must not be modified; however, after the database specific views are created, you can create your own views by copying existing views and then modifying the copied views.

Error messages for each table are self-explanatory, and are presented in the format `ERROR_<Error Condition>` For example, `ERROR_NAME_NOT_FOUND`, or `ERROR_DUPLICATED_DIMENSIONS`.

➤ To create model definition data database views:

1 **Connect to the database server that contains the Profitability and Cost Management application and the schema, and locate the `create_sql` script.**

2 Locate the `create_sql` script to create Database Views in the default location for your database type:

- `EPM_ORACLE_HOME/products/Profitability/database/Common/MSSQLServer/view`
- `EPM_ORACLE_HOME/products/Profitability/database/Common/Oracle/view`

3 Run a query to create a database view against your Profitability and Cost Management database.

4 Open the view to display a list of the requested model definition data.

5 Repeat step 3 and step 4 for each view that you want to create.

HPM_EXP_STAGE

The HPM_EXP_STAGE table provides a list of the stages within the model, and identifies the display order, prefix, associated dimensions and driver dimension for each stage.

Table 19 HPM_EXP_STAGE

Field	SQL Server Data Type	Oracle Data Type	Description
application_name	nvarchar (80)	varchar2 (80)	Name of the application
id	identity	integer	Unique record ID
name	nvarchar (80)	varchar2 (80)	Name of the model stage.
description	nvarchar (255)	varchar2 (255)	Brief explanation of the purpose of the stage.
stage_order	integer	integer	The sequence position in which the selected stage is to be used within the model while modeling (assignments and rules creation), in calc script generation and calculation.
stage_prefix	nvarchar (80)	varchar2 (80)	Differentiating prefix for the stage name.
dim1_name	nvarchar (80)	varchar2 (80)	Name of the first dimension included in the stage.
dim2_name	nvarchar (80)	varchar2 (80)	Name of the second dimension included in the stage, if available.
dim3_name	nvarchar (80)	varchar2 (80)	Name of the third dimension included in the stage, if available.
driver_dim_name	nvarchar (80)	varchar2 (80)	Name of the dimension selected as the driver for the stage.
Is_intra	nvarchar (1)	varchar2 (255)	Enter “Y” (Yes) to enable intrastage assignments within the model stage. Default is “N” (No). This field is populated from the application.

HPM_EXP_POV

The HPM_EXP_POV table stores the states of each combination of dimension members included in a point of view (POV).

Table 20 HPM_EXP_POV

Field	SQL Server Data Type	Oracle Data Type	Description
application_name	nvarchar (80)	varchar2 (80)	Name of the application
id	identity	integer	Unique record ID
pov_dim1_ member_name	nvarchar (80)	varchar2 (80)	Name of the first POV dimension member.
pov_dim2_ member_name	nvarchar (80)	varchar2 (80)	Name of the second POV dimension member, if available.
pov_dim3_ member_name	nvarchar (80)	varchar2 (80)	Name of the third POV dimension member, if available. Additional dimensions and member names may be added, as required.
pov_dim4_ member_name	nvarchar (80)	varchar2 (80)	Name of the fourth POV dimension member, if available. Additional dimensions and member names may be added, as required.
pov_state	nvarchar (80)	varchar2 (80)	The current status of the POV: ● Draft ● Published ● Archived

HPM_EXP_DRIVER

The HPM_EXP_DRIVER table provides details about the driver for the selected stage, including driver type, formula and layer.

Table 21 HPM_EXP_DRIVER

Field	SQL Server Data Type	Oracle Data Type	Description
application_ name	nvarchar (80)	varchar2 (80)	Name of the application
id	identity	integer	Unique record ID
driver_name	nvarchar (80)	varchar2 (80)	Name of the driver.
description	nvarchar (255)	varchar2 (255)	Description of the purpose for the selected driver.
display_order	integer	integer	Display position of the driver within the list of all drivers in the model.

Field	SQL Server Data Type	Oracle Data Type	Description
driver_type	nvarchar (80)	varchar2 (80)	Type of driver: ● Even ● Simple ● Simple Weighted ● Simple Variable ● Weighted Variable ● Fixed and Variable ● Percentage ● Custom
complete_ formula	nvarchar (500)	varchar2 (500)	Formula created for the driver by a user. This formula must be created using Essbase calculation script syntax. See the *Oracle Hyperion Profitability and Cost Management User's Guide*.
fixed_location	nvarchar (30)	varchar2 (30)	Location of the standard driver measure “FixedDriverValue.”
rate_member	nvarchar (80)	varchar2 (80)	Member in the Measures dimension that corresponds to the standard driver measure “Rate.”
rate_location	nvarchar (30)	varchar2 (30)	Location of the standard driver measure “Rate.”
weight_member	nvarchar (80)	varchar2 (80)	Member in the Measures dimension that corresponds to the standard driver measure “Weight.”
weight_location	nvarchar (30)	varchar2 (30)	Location of the standard driver measure “Weight.”
volume_member	nvarchar (80)	varchar2 (80)	Member in the Measures dimension that corresponds to the standard driver measure “Volume.”
volume_location	nvarchar (30)	varchar2 (30)	Location of the standard driver measure “Volume.”
custom_formula	nvarchar (500)	varchar2 (500)	Formula created for the driver by a user. This formula must be created using Essbase calculation script syntax. See the *Oracle Hyperion Profitability and Cost Management User's Guide*.
cost_layer	nvarchar (30)	varchar2 (30)	The driver layer is set to Cost layer, which contains the cost values for the model. **Note:** Either the cost_layer, revenue_layer, or both must be selected.
revenue_layer	nvarchar (30)	varchar2 (30)	The driver layer is set to Revenue layer, which contains the revenue values for the model. **Note:** Either the cost_layer, revenue_layer, or both must be selected.
allow_idle_cost	nvarchar (1)	varchar2 (1)	Enter appropriate value to determine whether idle costs are allowed for this driver: ● Enter “Y” (Yes) to allow idle cost for a driver. ● Enter “N” (No) to disable idle cost for a driver. This is the default.

Field	SQL Server Data Type	Oracle Data Type	Description
driver_basis_ type	nvarchar (80)	VARCHAR2 (80)	Select the required driver basis: ● Actual Basis ● Standard Basis **Note:** The Standard Basis driver cannot be used with Even or Percentage driver types.
priority	integer	integer	Enter the calculation priority of a driver so that allocations within a stage can be run in the specified order. The driver with the lowest priority is processed first. By default, the priority is set to 100, and the lowest (or first) priority is 0. Numbers do not have to be sequential. Drivers with the same priority are executed in no particular order. Only whole, positive numbers are valid.

HPM_EXP_DRIVER_SELECTION

The HPM_EXP_DRIVER_SELECTION table stores drivers selected for the intersections of the exact stage.

Table 22 HPM_EXP_DRIVER_SELECTION

Field	SQL Server Data Type	Oracle Data Type	Description
application_name	nvarchar (80)	varchar2 (80)	Name of the application
id	identity	integer	Unique record ID
pov_dim1_member_name	nvarchar (80)	varchar2 (80)	Name of the first dimension member in the selected point of view.
pov_dim2_member_name	nvarchar (80)	varchar2 (80)	Name of the second dimension member in the point of view, if applicable.
pov_dim3_member_name	nvarchar (80)	varchar2 (80)	Name of the third dimension member in the point of view, if applicable.
pov_dim4_member_name	nvarchar (80)	varchar2 (80)	Name of the fourth dimension member in the point of view, if applicable.
layer_name	nvarchar (80)	varchar2 (80)	Name of the selected layer for the point of view: ● Cost (Default) ● Revenue
stage_name	nvarchar (80)	varchar2 (80)	Name of the model stage for the selected point of view.
dim1_member_name	nvarchar (80)	varchar2 (80)	Member name of the first dimension of the selected stage.
dim2_member_name	nvarchar (80)	varchar2 (80)	Member name of the second dimension of the selected stage, if applicable.

Field	SQL Server Data Type	Oracle Data Type	Description
dim3_member_name	nvarchar (80)	varchar2 (80)	Member name of the third dimension of the selected stage, if applicable.
driver_name	nvarchar (80)	varchar2 (80)	Name of the driver for the selected intersection.

HPM_EXP_ASSIGNMENT

The HPM_EXP_ASSIGNMENT table provides details about each assignment, including the source stage, POV, driver layer, source dimension members, destination stage, and destination dimension members.

Table 23 HPM_EXP_ASSIGNMENT

Field	SQL Server Data Type	Oracle Data Type	Description
application_name	nvarchar (80)	varchar2 (80)	Name of the application
id	identity	Identity	Unique record ID
pov_dim1_member_name	nvarchar (80)	varchar2 (80)	Name of the first dimension member in the selected point of view.
pov_dim2_member_name	nvarchar (80)	varchar2 (80)	Name of the second dimension member in the point of view, if applicable.
pov_dim3_member_name	nvarchar (80)	varchar2 (80)	Name of the third dimension member in the point of view, if applicable.
pov_dim4_member_name	nvarchar (80)	varchar2 (80)	Name of the fourth dimension member in the point of view, if applicable.
layer_name	nvarchar (80)	varchar2 (80)	Name of the selected layer for the point of view: ● Cost (Default) ● Revenue **Note:** Either the cost_layer or the revenue_layer must be selected.
src_stage_name	nvarchar (80)	varchar2 (80)	Name of the source stage for the assignment.
src_dim1_member_name	nvarchar (80)	varchar2 (80)	Name of the first dimension member in the source stage.
src_dim2_member_name	nvarchar (80)	varchar2 (80)	Name of the second dimension member in the source stage, if available.
src_dim3_member_name	nvarchar (80)	varchar2 (80)	Name of the third dimension member in the source stage, if available.
dst_stage_name	nvarchar (80)	varchar2 (80)	Name of the destination stage for the assignment.
dst_dim1_member_name	nvarchar (80)	varchar2 (80)	Name of the first dimension member in the destination stage.

Field	SQL Server Data Type	Oracle Data Type	Description
dst_dim2_member_name	nvarchar (80)	varchar2 (80)	Name of the second dimension member in the destination stage, if available.
dst_dim3_member_name	nvarchar (80)	varchar2 (80)	Name of the third dimension member in the destination stage, if available.

HPM_EXP_ASGN_RULE_SELECTION

The HPM_EXP_ASGN_RULE_SELECTION table stores details about the source stage and dimension members for a rule controlling the assignments for the selected stage.

Table 24 HPM_EXP_ASSGN_RULE_SELECTION

Field	SQL Server Data Type	Oracle Data Type	Description
application_name	nvarchar (80)	varchar2 (80)	Name of the application
id	identity	integer	Unique record ID
pov_dim1_member_name	nvarchar (80)	varchar2 (80)	Name of the first dimension member in the selected point of view.
pov_dim2_member_name	nvarchar (80)	varchar2 (80)	Name of the second dimension member in the point of view, if applicable.
pov_dim3_member_name	nvarchar (80)	varchar2 (80)	Name of the third dimension member in the point of view, if applicable.
pov_dim4_member_name	nvarchar (80)	varchar2 (80)	Name of the fourth dimension member in the point of view, if applicable.
layer_name	nvarchar (80)	varchar2 (80)	Name of the selected layer for the point of view: ● Cost (Default) ● Revenue **Note:** Either the cost_layer, revenue_layer, or both is required.
src_stage_name	nvarchar (80)	varchar2 (80)	Name of the source stage for the assignment rule.
src_dim1_member_name	nvarchar (80)	varchar2 (80)	Name of the first dimension member in the source stage.
src_dim2_member_name	nvarchar (80)	varchar2 (80)	Name of the second dimension member in the source stage, if available.
src_dim3_member_name	nvarchar (80)	varchar2 (80)	Name of the third dimension member in the source stage, if available.
rule_name	nvarchar (80)	varchar2 (80)	Name of the rule controlling the selected assignment. The rule must be present in the target database.

C

Essbase Naming Conventions

In This Appendix

Generated Calculation Script Naming Conventions

When Profitability and Cost Management generates Essbase scripts, the scripts are automatically named using specific conventions. The driver calculation scripts and allocation calculation scripts are combined into one script per stage. The script name is created in the following format:

```
String scriptName = scriptSuffix + POV-identifier + Stage Order Number
+ “_” + index;
```

- The script Suffix is based on the type of script. The following table displays the list of suffixes.

Table 25 Calculation Script Suffixes

Script Type	Cost Layer	Revenue Layer
Interstage Allocation	“a”	“r”
Intrastage Allocation	“i”	“t”

- The POV-identifier is based on the POV ID and may include up to three digits. A script is generated and identified for every POV.
- The Stage Order Number is the order number for the Source stage (for example, 1, 2, 3, and so on).
- If multiple scripts are generated because of script splitting, the _index displays the numerical sequence of the scripts for the same type, POV, Stage, and Layer, starting with _01, _02, and so on.

Sample Script Names

- `a3682_01.csc` represents a calculation script for POV identifier 368, Source Stage 2, and the cost layer.

- t4533_02.csc represents an intrastage calculation script for POV identifier 453, Source Stage 3, and the revenue layer.

Essbase Naming Restrictions for Applications and Databases

When creating names for applications and databases, enter the name in the case in which you want the word displayed. The application or database name is created exactly as it is entered. If you enter the name as all capital letters (for instance, NEWAPP), Essbase does not automatically convert it to upper- and lowercase (for instance, Newapp).

Note: This list provides a partial set of the restrictions. For the complete list of all restrictions, Oracle recommends that you refer to the *Oracle Essbase Database Administrator's Guide:*

The following naming restrictions apply when you are naming applications and databases:

- For non-Unicode-mode application and database names, use no more than 8 bytes.
- For Unicode-mode application and database names, use no more than 30 characters.
- Do not use spaces in the name.
- Do not use the following special characters in the name:
  - * asterisks
  - + plus signs
  - \ backslashes
  - ? question marks
  - [] brackets
  - " double quotation marks
  - : colons
  - ; semicolons
  - , commas
  - ‘ single quotation marks
  - = equal signs
  - / slashes
  - > greater than signs
  - tabs
  - < less than signs
  - | vertical bars
  - . periods

- For aggregate storage databases only, do not use the following words as application or database names:
  - DEFAULT
  - LOG
  - METADATA
  - TEMP

Essbase Naming Restrictions for Dimensions, Members, and Aliases

When defining dimensional outlines, there are restricted characters that may not be used for naming dimensions, members and aliases. A list of the most common restricted characters is provided in this section; however, Oracle strongly suggests that you review the Essbase naming conventions described in the *Oracle Essbase Database Administrator's Guide* for a complete list.

Note: This list provides a partial set of the restrictions. For the complete list of all restrictions, Oracle recommends that you refer to the *Oracle Essbase Database Administrator's Guide:*

When naming dimensions, members and aliases, follow these naming restrictions:

- 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. To enable case-sensitivity, see "Setting Outline Properties" in the *Oracle Essbase Database Administrator's Guide*.
- Do not use HTML tags in dimension or member names, aliases, and descriptions.
- Do not use quotation marks, periods, brackets, backslashes, or tabs within a name.

  Caution! Brackets are permitted but not recommended in block storage outlines because they cause errors when converting to aggregate storage outlines.

- Duplicate member names or aliases are not allowed within the same dimension.
- Do not use the following characters to begin dimension or member names:
  - at signs (@)
  - backslashes (\)
  - brackets ([])
  - commas (,)
  - dashes
  - hyphens
  - minus signs (-)

- equal signs (=)
  - less than signs (<)
  - parentheses ( )
  - periods (.)
  - plus signs (+)
  - single quotation marks (')
  - quotation marks (")
  - underscores (_)
  - vertical bars(|)
- Do not place spaces at the beginning or end of names, as they are ignored by Essbase.
- Do not use forward slashes in member names.
- For time periods in custom calendars, do not use spaces in prefixes.
- Do not use the following items as dimension or member names.
  - Calculation script commands, operators or keywords. For a complete list of commands, see the *Oracle Essbase Database Administrator's Guide*.
  - Report writer commands
  - Function names and function arguments
  - If Dynamic Time Series is enabled, do not use History, Year, Season, Period, Quarter, Month, Week, or Day.
  - Names of other dimensions and members (unless the member is shared), generation names, level names, and aliases in the database.
- Do not use the following words:
  - ALL
  - AND
  - ASSIGN
  - AVERAGE
  - CALC
  - CALCMBR
  - COPYFORWARD
  - CROSSDIM
  - CURMBRNAME
  - DIM
  - DIMNAME
  - DIV
  - DYNAMIC
  - EMPTYPARM

- EQ
- EQOP
- EXCEPT
- EXP
- EXPERROR
- FLOAT
- FUNCTION
- GE
- GEN
- GENRANGE
- GROUP
- GT
- ID
- IDERROR
- INTEGER
- LE
- LEVELRANGE
- LOOPBLOCK
- LOOPPARMS
- LT
- MBR
- MBRNAME
- MBRONLY
- MINUS
- MISSING
- MUL
- MULOP
- NE
- NON
- NONINPUT
- NOT
- OR
- PAREN
- PARENPARM
- PERCENT

- PLUS
- RELOP
- SET
- SKIPBOTH
- SKIPMISSING
- SKIPNONE
- SKIPZERO
- TO
- TOLOCALRATE
- TRAILMISSING
- TRAILSUM
- UMINUS
- UPPER
- VARORXMBR
- XMBRONLY
- $$UNIVERSE$$
- #MISSING
- #MI

Essbase Naming Conventions for Attribute Calculations Dimension

The members that Essbase creates in the attribute calculations dimension (Sum, Count, Min, Max, and Avg) are not considered reserved words because you can change these names in the attribute calculations dimension and then use the standard name in an attribute or standard dimension.

If the outline is tagged as a unique member outline, avoid using Sum, Count, Min, Max, and Avg as member names. For example, if you use Max in a standard dimension and then create an attribute dimension, in which Essbase creates the Max member in the attribute calculations dimension, Essbase detects a duplicate name and returns the following error message:

```
"Analytic Server Error(1060115): Attribute Calculations dimension/
member name already used."
```

If the outline is tagged as a duplicate member outline, and an attribute dimension (and, therefore, the attribute calculations dimension) exists before you use Sum, Count, Min, Max, and Avg as a base member, Essbase allows the duplicate name. However, if, you use Sum, Count, Min, Max, and Avg as a base member before creating an attribute dimension, the duplicate name is not allowed.

D

Performance Tuning

In This Appendix

About Performance Tuning in Profitability and Cost Management

Every instance of Profitability and Cost Management is composed of a unique combination of applications, hardware, software, databases, customizations, and so on. With such diversity in installations, any changes to the current configuration, such as new hardware or software, may result in changes in performance.

After installing a new release or patch, or after making substantial changes within your environment, including increases in data and users, or modeling style, some tuning of these components is probably required.

Performance tuning is an iterative process. To maximize performance, all components in your installation should be maintained, tuned and monitored on an ongoing basis. In addition to any modifications to your model, you may also need to tune the components listed in "Profitability and Cost Management and Related Components" on page 110.

It is difficult to supply definitive tuning solutions that will work in every situation. For example, different versions or patches may exhibit slightly different behavior that needs to be managed. Depending on your environment, the interplay between components may yield different results. Customization of this product or others that share the same environment may impact results.

This appendix is designed for information purposes only, to suggest some areas for examination, and to direct you to information sources that may help you to fine-tune your Profitability and Cost Management installation.

Caution! Before experimenting with tuning, settings, and so on, make a backup of your databases and models. See *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.

Profitability and Cost Management and Related Components

Profitability and Cost Management works through EPM Workspace, and uses a number of other products or components. Some or all of these components may require fine-tuning when any changes are implemented through new releases or patches.

The following components are always associated with Profitability and Cost Management, and should be evaluated for performance:

- EPM Workspace
- Performance Management Architect
- Shared Services
- Essbase

In addition, you should review the performance of any other products that are included in your installation, such as Financial Reporting, Oracle Hyperion Planning, Fusion Edition, and so on:

Within Profitability and Cost Management itself, performance should also be evaluated for the following components:

- Hardware
- Operating system
- Relational and Essbase databases
- Application servers
- Web servers
- Network settings, and clusters, if applicable
- Third-party applications, such as Microsoft Word and Microsoft Excel.

Profitability and Cost Management Major Operations and Performance Considerations

The following table highlights the major processes involved in the creation and calculation of a Profitability and Cost Management model. The table also shows the areas for each main model task that may require some tuning over time, such as the number of attributes or assignments.

Table 26 Major Processes in Developing the Profitability and Cost Management Model

Operation	Components Involved	Actions	User Design Considerations	Technical Considerations
Metadata Management	● Performance Management Architect	● Manage dimensions ● Synchronize metadata and data movement between Oracle applications	● Number of dimensions and members	
Performance Management Architect Deploy	● Performance Management Architect ● Profitability and Cost Management	● Performance Management Architect writes new metadata out to XML ● Profitability and Cost Management reads XML, and updates its databases: ❍ New deployment sends all members to Profitability and Cost Management ❍ Redeployment sends full member sets of dimensions that have been edited	● Number of dimensions and members ● Use of Attributes	● JVM memory settings ● Database tuning
Profitability and Cost Management Staging Import	● Profitability and Cost Management	Profitability and Cost Management reads data from staging table database and updates its database	● Staging table size - especially explicit assignments ● View use may have an impact on performance	Database tuning
Essbase Database Deploy	● Profitability and Cost Management ● Essbase	● User creates Essbase database, with metadata stored in Profitability and Cost Management ● Profitability and Cost Management calls Essbase to deploy metadata and constructs and Essbase Database, Application and Outline ● Profitability and Cost Management updates Performance Management Architect applications with cloned dimensions	● Member counts ● Dimension count. Stage settings can multiply dimensions several times	Tune Essbase, considering the following: ● Dense and Sparse settings ● Outline ordering ● Caches
Allocation Model Definition	● RDBMS ● Oracle	● Create and manage business model ● Generate automatic calc scripts	Manage the number of model elements, such as: ● Stages ● Number of Assignments ● Drivers	● Database tuning

Operation	Components Involved	Actions	User Design Considerations	Technical Considerations
Calc Script Generation	● Profitability and Cost Management ● Essbase	● Store Cost and Revenue and Driver data. Profitability and Cost Management generates large text-based file and writes it to the Essbase Database	● Model size, including Stages and Dimension/members ● Model complexity, or the density of assignment paths ● Use of intrastage assignments ● Heavy use of explicit assignments. This option causes scripts to quickly increase in size. ● Remove "0" data values from the allocation, if they are not necessary. Meaningless data (such as zeroes or very small values) may take up space that can either be removed or aggregated.	● JVM memory settings
Calc Script Execution	● Profitability and Cost Management ● Essbase Calculation Cube	● Calculate Model and store results. Profitability and Cost Management launches a job (calc script execution) in Essbase.	● Use of explicit assignments ● Use of rules without well-defined boundaries. For example, creating Assignment "All x" cross "All y" where only a few combinations have actual driver data present	● JVM memory settings ● Tune Essbase settings, such as: ❍ Dense or Sparse settings ❍ Outline ordering ❍ Caches
Genealogy Calculation	● Profitability and Cost Management ● Essbase	● Profitability and Cost Management queries Essbase Reporting cube, calculates results and writes back	Model complexity: ● Number of stages ● Number of stages included in the calculation ● Density of assignment paths	JVM memory settings
Profitability and Cost Analysis	● Essbase Reporting Database	● S9 BI+ ● Oracle Hyperion Financial Reporting, Fusion Edition ● Oracle's Hyperion® Web Analysis ● Profitability Analysis ● Cost Management Analysis - Genealogy Data		

Output Log Files

Administrators can generate output log files throughout EPM Workspace to help technicians identify system or environmental problems, or to help developers debug reports or API programs.

The level of detail in the output logs may impact performance. To modify the log level settings, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*. Information regarding logging is in the “Using EPM System Logs” chapter.

The following table shows the output log files that are available for Profitability and Cost Management. To view information about log files for related products and applications, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

Table 27 Profitability and Cost Management Output Log Files

Log File	Description
`hpcm.log`	Profitability and Cost Management generates an application, server-side log file that collects application-specific messages that are sent from the application or server. By default, the log files are available at `C:\oracle\Middleware\user_projects\domains\EPMSystem\servers\Profitability0\logs`. Contact your system administrator for access to this log file.
`SharedServices_Security_Client.log`	A Shared Services Client-side log file provides details regarding the Profitability and Cost Management handshake with Common Security Services. By default, the log file is available at `C:\oracle\Middleware\user_projects\domains\EPMSystem\servers\Profitability0\logs.`

Hardware Considerations

For information about the software and hardware requirements for release compatibility with earlier releases, and database and Java application server requirements, review the documents available at `http://www.oracle.com/technology/products/bi/hyperion-supportedplatforms.html`.

Hardware Requirements

After installing a new release or patch, you may find that existing hardware is insufficient for new requirements and may need to be tuned or upgraded. For additional information, see the following documentation:

- *Oracle Hyperion Enterprise Performance Management System Certification Matrix*
- *Oracle Hyperion Enterprise Performance Management System Installation Start Here*

64-bit Versus 32-bit

Although the 32-bit edition may be sufficient for your installation, the 64-bit edition of Essbase can handle larger outlines and cache sizes than 32-bit Essbase. In computing environments that support it, implementing 64-bit Essbase can improve the performance of existing applications and can sustain much larger applications.

For additional information, see the following documentation:

- *Oracle Hyperion Enterprise Performance Management System Installation Start Here*
- *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide.*

Disk Space and RAM Requirements

Disk space and RAM requirements for client and server software may need to be reviewed after installing a new release or patch.

For additional information, see the *Oracle Hyperion Enterprise Performance Management System Installation Start Here.*

Sizing

Sizing of your installation is an important factor in performance.

In order to get the most out of your configuration, or to identify bottlenecks that may be limiting your performance, contact your consultant or Oracle representative for a review of models, allocations, intersections, and so on. Based on this review, your consultant or Oracle representative will be able to size your particular configuration to maximize performance.

Database Tuning

Perform database tuning on a regular basis to optimize the performance of the database and system resources.

You may perform tuning on a variety of components, such as the following examples:

- Database files design
- Hardware and software, including the operating system and CPU
- Customization of settings and the configuration
- Query tuning to improve performance by tuning various elements, such as memory size, type of file system, number and type of processors, and so on
- Database indexing to improve the speed of data retrieval operations and potentially save disk space

For information on tuning databases, refer to the documentation listed in Table 28, Database Tuning Documentation.

Table 28 Database Tuning Documentation

Component	Documentation Library Tab	Documentation
Essbase	Essbase	*Oracle Essbase Database Administrator's Guide*
Oracle Database	Related	*Oracle Database documentation (11g and 10g)*
Application Server	Related	● *Oracle Application Server (10g)* ● WebLogic Tuning Guide: JRockIT Production Deployments JVM GC, available from http://download.oracle.com/docs ● See also the Sun JVM GC Tuning Guide at http://java.sun.com/docs/

Essbase Settings

When creating Essbase database structures, you need to consider many factors, such as the order of the Essbase outline, whether the dimensions should be set as dense or sparse, and how to optimize the Essbase caches.

See the following sections for a brief description of some Essbase settings:

- "Dense and Sparse Settings" on page 115
- "Essbase Caches" on page 117
- "Ordering Essbase Outlines" on page 117
- "Transmission Control Protocol/Internet Protocol (TCP/IP)" on page 117

The best resource for information about optimizing Essbase is the *Oracle Essbase Database Administrator's Guide* which is available from the Oracle Technology Network (http://www.oracle.com/technology/documentation/epm.html). In particular, review the suggestions in "Improving Essbase Performance."

Dense and Sparse Settings

A typical Profitability and Cost Management application contains one Measures dimension, one AllocationType dimension, several POV dimensions, and a number of business dimensions. The Dense and Sparse settings must be managed to reduce the size of Essbase databases and to avoid writing the same block multiple times in Essbase, which may cause fragmentation.

Profitability and Cost Management duplicates business dimensions if they are used in more than one stage. This process increases the Sparsity of the Essbase Calculation Cube outline generated by the application, and may have a performance impact when the calculation scripts are run.

Some optimization can be achieved simply by changing the Dimension Storage Type directly for the generated Calculation Cube outline, using Oracle Essbase Administration Services (EAS) console.

Caution! Changes to the Dimension Storage Type property should only be performed by a Database Administrator (DBA). For detailed instructions, refer to the *Oracle Essbase Administration Services Developer's Guide*.

For Profitability and Cost Management, the default recommendation to set the Dimension Storage Type property for these dimensions follows:

- Set the Measures and the AllocationType dimensions to Dense as the default.

  Note: The exception to this rule would be if the largest dimension in the largest stage is set to Dense, then the Measures and AllocationType dimensions should then be set to Sparse.

- Set all business and POV dimensions to Sparse

Note: Dimensions with attribute associations must be set as Sparse dimensions only.

This default setting results in an Essbase calculation cube with the block size of about 3K, and a large number of potential blocks based on the dimensionality of the Sparse dimensions. The most dense dimension (based on the existence of data) in the largest stage should be set to Dense. In this instance, the Measures and AllocationType dimensions should be set to Sparse.

Example of Dense/Sparse Settings

The following quick example may help make this decision.

The largest stage is defined by the largest number of potential nodes in the stage. If a typical Profitability and Cost Management model has its last stage defined as Customers X Products (to be able to calculate Customer x Product Profitability), and it is typically the largest stage, then either the Customer or the Product dimension in this stage can be set to Dense. If the largest dimension in the largest stage is set to Dense, then the Measures and AllocationType dimensions should be set to Sparse.

The decision on whether to set Customers or Products to Dense is based on the data density of the governing drivers. For example:

- If Sales Volume is the driver that is used to allocate to this stage (from various sources in previous stages), if the Customers dimension has more dense data than the Products dimension (few Products sold to more Customers), the Customers dimension can be set to Dense.
- If more Products are sold to fewer Customers), then Products can be set to Dense.

In either case, the Essbase block size should be within the limits of what Essbase best practices recommend. See the *Oracle Essbase Administration Services Developer's Guide*.

Essbase Caches

Essbase uses different types of memory caches to coordinate memory usage.

Essbase provides default size settings for each cache. Adjust Essbase cache settings according to the model requirements. Appropriate cache size is affected by many factors, including database size, block size, index size, and available server memory. Cache size settings can significantly affect database and general server performance.

The Essbase caches can be tuned to optimize performance. Refer to the *Oracle Essbase Database Administrator's Guide* for detailed instructions on working with caches.

Ordering Essbase Outlines

When creating the Essbase outline, the position of dimensions in the outline and the storage properties of dimensions can affect two areas of performance—how quickly calculations are run and how long it takes users to retrieve information.

Refer to the *Oracle Essbase Database Administrator's Guide* for detailed instructions on designing an outline to optimize performance.

Transmission Control Protocol/Internet Protocol (TCP/IP)

Communication between the client and database tiers in Essbase is through TCP/IP or HTTP.

Refer to the *Oracle Essbase Database Administrator's Guide* for instructions on working with TCP/IP.

Increasing JVM Memory Settings

Depending on the size of your environment, you might need to increase the JVM memory settings (heap size) for your Web application server to optimize performance. While the default can be used in a test environment (single-server-type configuration), the heap size must be increased for use in a production environment (multiple servers).

You can modify the JVM memory settings, depending on whether Profitability and Cost Management is being launched through a Windows service or from the command line. In either case, for 32-bit systems, Oracle recommends a maximum setting of 1024 MB. You can change the custom startup scripts.

For additional information on JVM memory settings, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

➤ To increase the JVM memory allocation if Profitability and Cost Management is being launched from Windows service:

1 **Navigate to** `HKEY_LOCAL_MACHINE\SOFTWARE\Hyperion Solutions\`*`Product Component`*`\`*`ServiceName`***.**

2. Using regedit, update the Windows registry for Windows services to set the JVMOptionX (where X is 1, 2, and so on) whose value starts with -Xmx. For 32-bit systems, Oracle recommends a maximum setting of 1024 MB.

3. Restart the Windows service for the product.

➤ To increase the JVM memory allocation if Profitability and Cost Management is being launched from the command line:

1. Edit your start script to specify the desired heap size. For 32-bit systems, Oracle recommends a maximum setting of 1024 MB.

2. Restart the script for the product.

Solving Connectivity Issues

Using the Embedded Connection Type

By default, Profitability and Cost Management uses APS mode to connect to Essbase, and APS uses many TCP ports while deploying the Essbase databases. This situation may cause a network error, which is displayed in the Profitability and Cost Management log file.

If this error occurs, switch the Connection Type to embedded mode and redeploy the database.

➤ To set the Connection Type to Embedded Mode:

1. In Profitability and Cost Management, from Task Areas, select **Manage Model**, and then **Model Summary.**

2. On the Model Summary screen, select the **Model Level Preference** tab.

3. Under **Essbase Information**, from the **Connection Type** drop-down list, select **Embedded.**

4. Click the Save icon.

Modifying TCPIP Parameters

If the database deployment fails in embedded mode, or if you must use APS mode, you can modify the following Windows registry TCPIP parameters, as recommended by the Microsoft Windows Operating System documentation:

- TcpTimedWaitDelay
- MaxUserPort

Caution! Before making changes to the Microsoft Windows Registry, consult the Microsoft Windows Operating System documentation for the correct settings and best practices to avoid any risk of corrupting the registry.

➤ To modify TCPIP parameters:

1. **From the Microsoft Windows Registry, navigate to** `KEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Services\TCPIP\Parameters.`
2. **Add or modify the following registry entries:**
   - TcpTimedWaitDelay—Set this entry to a low value (for example, 30) for the sockets to be released quickly. This DWORD value ranges between 30-300.
   - MaxUserPort—Set this entry to a high value (for example, 33000) to increase the total numbers of sockets that can be connected to the port. This is a DWORD value.

   See the Microsoft Windows Operating System documentation for more details before modifying these parameters.
3. **Reboot the system for the changes to take effect.**

Clustering for High Availability

A cluster is an array of servers or databases that behave as a single resource which share task loads and provide failover support. Clustering enables optimal use of multiple resources, but may require fine-tuning on a consistent basis. For example, if Oracle Enterprise Performance Management Workspace, Fusion Edition is installed as a distributed system, there may also be network issues to be considered, such as network latency.

- For information on clusters set up for Oracle Hyperion Enterprise Performance Management System services and applications, Oracle Essbase, Shared Services, and so on, see the *Oracle Hyperion Enterprise Performance Management System High Availability and Disaster Recovery Guide.* Additional information is available in the White Papers Library at Oracle Enterprise Performance Management/Business Intelligence White papers. (http:// www.oracle.com/ technology/products/bi/resource-library.html).
- For Java Application servers, cache clustering support is driven by TopLink Essentials, which is bundled with Oracle Hyperion Profitability and Cost Management, Fusion Edition. TopLink also offers cache coordination where changes made in one node can be synchronized, replicated, or invalidated across multiple nodes of the same application forming a cluster or grid.

  For more information, see these documents about TopLink:

  - `http://www.oracle.com/technology/tech/java/newsletter/articles/ toplink/ toplink_caching_locking.html`
  - `http://www.oracle.com/technology/products/ias/toplink/JPA/ essentials/toplink-jpaextensions. html#TopLinkCaching`

Additional Reference Documentation

Tuning an installation is an ongoing process. Apart from reviewing performance after the installation of a new release or service fix, there may be opportunities that arise during production or with modifications to the hardware and software to improve or make adjustments.

Tuning is usually specific to the operating system, hardware, and software release. The performance of Oracle Hyperion Profitability and Cost Management, Fusion Edition is also sensitive to the volume and size of data, as well as the actual model structure.

Refer to the following documentation, available from the Oracle Technology Network (`http://www.oracle.com/technology/documentation/epm.html`) for additional reference documents that may provide tuning suggestions that apply to your system.

Table 29 Tuning Reference Documentation

Component	Documentation Library Tab	Documentation
Installation and Configuration	Deployment	● *Oracle Hyperion Enterprise Performance Management System Installation Start Here* ● *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*
Oracle's Hyperion® Shared Services	Foundation Services	● *Oracle Hyperion Enterprise Performance Management System Security Administration Guide* ● *Oracle Hyperion Enterprise Performance Management System User and Role Security Guide*
Oracle Hyperion EPM Architect, Fusion Edition	Foundation Services	● *Oracle Hyperion Enterprise Performance Management Architect, Fusion Edition Administrator's Guide* ● *Oracle Hyperion Enterprise Performance Management Architect, Fusion Edition Batch Client User's Guide (Release 11.1.1 and 11.1.1.3)* ● *Oracle Hyperion Enterprise Performance Management Architect, Fusion Edition File Generator User's Guide*

Glossary

! See bang character.

#MISSING See missing data.

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 A property that determines 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.

active service A service whose Run Type is set to Start rather than to Hold.

active-active high availability system A system in which all the available members can service requests, and no member is idle. An active-active system generally provides more scalability options than an active-passive system. Contrast with active-passive high availability system.

active-passive high availability system A system with active members, which are always servicing requests, and passive members that are activated only when an active member fails. Contrast with active-active high availability system.

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 that an end user creates dynamically.

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.

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 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 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 requirements, reporting requirements, or both.

application administrator A person responsible for setting up, configuring, maintaining, and controlling an application. Has all application privileges and data access permissions.

application currency The default reporting currency for the application.

Application Migration Utility A command-line utility for migrating applications and artifacts.

application server cluster A loosely joined group of application servers running simultaneously, working together for reliability and scalability, and appearing to users as one application server instance. See also vertical application cluster and horizontal application cluster.

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.

asymmetric topology An Oracle Fusion Middleware Disaster Recovery configuration that is different across tiers on the production site and standby site. For example, an asymmetric topology can include a standby site with fewer hosts and instances than the production site.

attribute A 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 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.

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 1) In Financial Reporting, a container that holds a group of similar documents. Books may specify dimension sections or dimension changes; 2) In Data Relationship Management, a collection of exports that can be run together as a group. Export results can be combined together or output separately.

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 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 details information such as system availability, notification of application refresh, or application backups.

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 Accounts Accounts with formulas that you cannot alter. These formulas are fixed to maintain the accounting integrity of the model that you are building. For example, the formula for Net Income, a Calculated Account, is modeled into Strategic Finance and cannot be changed in historical or forecast periods.

calculated member in MaxL DML A member designed for analytical purposes and defined in the optional WITH section of a MaxL DML query.

Calculation Manager A module of Enterprise Performance Management Architecture (EPMA) that Planning and Financial Management users can use to design, validate, and administrate business rules in a graphical environment. c

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 An area that 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 A map that 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.

CDM See custom-defined macro.

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 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 in which 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.

cluster interconnect A private link used by a hardware cluster for heartbeat information, to detect node failure.

cluster services Software that manages cluster member operations as a system. With cluster services, you can define a set of resources and services to monitor through a heartbeat mechanism between cluster members and to move these resources and services to a different cluster member as efficiently and transparently as possible.

clustered bar charts Charts in which categories are viewed side-by-side; 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 In Data Relationship Management, a field of data associated with an import source or the results of a query, compare, validation, or export.

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.

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) 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 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 eanbles users to browse and select content to be placed on 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 groups Groupings 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 Reporting that 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 by the dollar (1* 0.923702). After conversion, the European euro amount is .92.

Currency Overrides A feature allowing the selected input method for any input period to be overridden to enable input of that period's value as Default Currency/Items. To override the input method, enter a pound sign (#) 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.

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.

cycle through 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 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, one FDM data load location exists 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 A feature that prevents changes to data according to specified criteria, such as a 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 A 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 analysis types 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 The unit scale of data. For example, If you select to define your analysis in thousands and enter 10, this unit 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) In Business Rules, a block of the database where calculated values are stored; 2) In Profitability and Cost Management, the association of a source and destination in the allocation model that controls the direction of allocated costs or revenue flow.

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 the 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, if they have been updated).

Disaster Recovery The ability to safeguard against natural or unplanned outages at a production site by having a recovery strategy for applications and data to a geographically separate standby site.

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 In Profitability and Cost Management, an allocation method that describes the mathematical relationship between the sources that use the driver and the destinations to which those sources allocate cost or revenue. For Business Modeling, see also cost driver and activity driver.

duplicate alias name A name that occurs more than once in an alias table and 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 Multiple occurrences 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 Members 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 Members 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, 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.

EPM Oracle home A subdirectory of Middleware home containing the files required by EPM System products. The EPM Oracle home location is specified during installation with EPM System Installer.

EPM Oracle instance A directory containing active, dynamic components of EPM System products (components that can change during run-time). You define the EPM Oracle instance directory location during configuration with EPM System Configurator.

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 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 a specific 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 EPM System products with user information stored outside the application. The user account is maintained by the EPM System, but password administration and user authentication are performed by an external service, using a corporate directory such as Oracle Internet Directory (OID) or Microsoft Active Directory (MSAD).

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.

failover The ability to switch automatically to a redundant standby database, server, or network if the primary database, server, or network fails or is shut down. A system that is clustered for failover provides high availability and fault tolerance through server redundancy and fault-tolerant hardware, such as shared disks.

Favorites gadget A gadget that contains links to Reporting and Analysis documents and URLs. See also gadget.

file delimiter A character, such as a comma or tab, that separates 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.

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 string 1) In Essbase, a method for transforming the way cell values are displayed; 2) In Data Relationship Management, a parameter of a Format or Formatted Date derived property that indicates the format in which a property value should be returned.

formula In Data Relationship Management, business logic used by a derived property to dynamically calculate a property value.

frame An area on the desktop. 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 In Data Relationship Management, a syntactic element of a derived property formula that accepts parameters and returns dynamic values.

gadget A simple, specialized, lightweight application that provides easy viewing of EPM content and enables access to core Reporting and Analysis functionality.

geneology 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 it is 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

hardware cluster a collection of computers that provides a single view of network services (for example, an IP address) or application services (such as databases and Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

high availability A system attribute that enables an application to continue to provide services in the presence of failures. This is achieved through removal of single points of failure, with fault-tolerant hardware, as well as server clusters; if one server fails, processing requests are routed to another server.

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.

horizontal application server cluster A cluster with application server instances on different machines.

host A server on which applications and services are installed.

host properties Properties pertaining to a host, or if the host has multiple Oracle EPM homes, to an Oracle EPM home.

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, a 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 A status that indicates changes in child entities consolidating into parent entities.

implied share A member with one or more children but only one that is consolidated, so the parent and child share a value.

import format In FDM, the definition of the structure of the source file that 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 status A status that indicates entities deactivated from consolidation for the current period.

inactive user A user whose account was 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. An index page contains pointers to data blocks.

input data Data loaded from a source rather than calculated.

installation assemblies Product installation files that plug in to EPM System Installer.

integration A process that is run to move data between Oracle's Hyperion applications using Shared Services. Data integration definitions specify the data moving between a source application and a destination application, and they 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, and so on), database software, the database server network address, and database user name. Administrators create and publish Interactive Reporting connection (.oce) files.

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 whether the accounts are in 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 nonintersecting dimensions, the nonintersecting 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.

intrastage assignment An assignment in the financial flow to an object 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 application server cluster An active-active application server cluster of Java Virtual Machines (JVMs).

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 Page.

KeyContacts gadget A gadget that 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. See also gadget.

latest A spreadsheet keyword 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 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: accrued expenses, accounts payable, and long-term debt.

lifecycle management The process of migrating an application, a repository, or individual artifacts across product environments.

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 taskflow, 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.

load balancer Hardware or software that directs the requests to individual application servers in a cluster and is the only point of entry into the system.

load balancing Distribution of requests across a group of servers, which helps to ensure optimal end user performance.

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.

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, 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 A data model 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, one or more logic accounts generated after a source file is loaded into FDM. Logic accounts are calculated accounts derived from the source data.

logical Web application An aliased reference used to identify the internal host name, port, and context of a Web application. In a clustered or high-availability environment, this is the alias name that establishes a single internal reference for the distributed components. In EPM System, a nonclustered logical Web application defaults to the physical host running the Web application.

LRO See linked reporting object.

managed server An application server process running in its own Java Virtual Machine (JVM).

manual stage A stage that requires human intervention.

Map File A file that stores 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 The 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 The data definition language used by Essbase for batch or interactive system-administration tasks.

MaxL DML The 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) A language used for querying and calculation in multidimensional-compliant databases.

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 members Jan, Feb, and Qtr1.

member list A named system- or user-defined group that references members, functions, or member lists within a dimension.

member load In Essbase 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 Essbase Integration Services, a template containing the structure and rules for creating an Essbase outline from an OLAP model.

Middleware home A directory that includes the Oracle WebLogic Server home and can also include the EPM Oracle home and other Oracle homes. A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.

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 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 (Multipurpose Internet Mail Extension) type is determined by the file extension or HTTP header. Plug-ins tell browsers which MIME types they support and which 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 nondimensional 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.

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 Customizable Workspace Pages created by users. They are marked specially so that they can be easily accessed from one single place without having to navigate the repository.

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.

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 Essbase 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 Essbase 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. See also online analytical processing (OLAP).

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.

Oracle home A directory containing the installed files required by a specific product, and residing within the directory structure of Middleware home. See also Middleware home.

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. P&L 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 An 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 entities associated with at least one node, they have entity, node, and parent information associated with them.

partition area A subcube 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 The extent to which an entity is controlled within the context of its group.

percent ownership 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).

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 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 A user who 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. Planners comprise the majority of users.

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 before 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 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 Oracle's Hyperion 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.

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.

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 governor An Essbase Integration Server parameter or Essbase Server configuration setting that controls the duration and size of queries made to data sources.

reciprocal assignment An assignment in the financial flow that also has the source as one of its destinations.

reconfigure URL A URL that is 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. A regular journal can be balanced, balanced by entity, or unbalanced.

Related Accounts Accounts related to the main account and grouped under the same main account number. 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 Storage location for 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.

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, or Forecast2.

schema In relational databases, a logical model that represents the data and the relationships between the data.

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. The levels, from highest to lowest: the entire system (Essbase Server), applications on Essbase Server, 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 A 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.

scraping An inspection of a data source to derive the most basic metadata elements from it. Contrast with introspection.

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, Oracle Access Manager, Oracle Single Sign-On, or CA SiteMinder) that protects corporate Web resources.

security platform A framework enabling Oracle 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 disks See shared storage.

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 The part of the Shared Services repository that manages EPM System deployment information for most EPM System products, including installation directories, database settings, computer names, ports, servers, URLs, and dependent service data.

shared storage A set of disks containing data that must be available to all nodes of a failover cluster; also called shared disks.

Shared Workspace Pages Workspace Pages shared across an organization that 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.

silent response files Files providing data that an installation administrator would otherwise be required to provide. Response files enable EPM System Installer or EPM System Configurator to run without user intervention or input.

single point of failure Any component in a system that, if it fails, prevents users from accessing the normal functionality.

single sign-on (SSO) The ability to log on once and then access multiple applications without being prompted again for authentication.

smart tags Keywords in Microsoft Office applications that are associated with predefined actions available from the Smart Tag menu. In Oracle EPM System products, smart tags can also be used to import Reporting and Analysis content and to access Financial Management and Essbase functions.

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. Contrast with dense dimension. For example, not all customers have data for all products.

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 1) A task description that forms one logical step within a taskflow, usually performed by an individual. A stage can be manual or automated; 2) For Profitability, logical divisions within the model that represent the steps in the allocation process within your organization.

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 RDBMS.

staging table 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, and then use the template as the basis for many regular journals.

Status bar The bar at the bottom of the screen that 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 nonsequential 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, a chart that rolls up detail charts shown below in the same column, plotting metrics at the summary level at the top of each chart column.

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 A setting that 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 simultaneously connect to an single instance.

symmetric topology An Oracle Fusion Middleware Disaster Recovery configuration that is identical across tiers on the production site and standby site. In a symmetric topology, the production site and standby site have the identical number of hosts, load balancers, instances, and applications. The same ports are used for both sites. The systems are configured identically and the applications access the same data.

sync Synchronization of 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 A feature that transfers data from application 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 Business processes in the taskflow management system that consist 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 A single instance of a taskflow including its state and associated data.

taskflow management system A system that defines, creates, and manages the execution of a taskflow, including definitions, user or application interactions, and application executables.

taskflow participant The resource that 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 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.

text measure 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 The time period that the data represents, such as fiscal or calendar periods.

time events Triggers for job execution.

time scale A scale that displays metrics by a specific time span, 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).

Timeline Viewer An FDM feature that enables users to view dates and times of completed process flow steps for specific locations.

Title bar A bar that displays the Strategic Finance name, the file name, and the scenario name Version box.

toast message A message that fades in the lower-right corner of the screen.

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. If a hierarchical relationship exists, the top-level member name is generally the same as the dimension name.

trace allocations A Profitability feature that enables you to visually follow the flow of financial data, either forwards or backwards, from a single intersection throughout the model.

trace level The level of detail captured in a 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, measures, and, ultimately, to its physical source elements. See also lineage.

traffic lighting Color-coding of report cells, or pins based on a comparison of two dimension members, or on fixed limits.

transformation 1) A process that transforms artifacts so that they function properly in the destination environment after application migration; 2) In data mining, the modification of 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 A process that 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 through a third common currency. For example, to convert balances from the Danish krone to the British pound, balances could be converted from the krone to the euro and from the euro to the pound.

triggers An Essbase feature whereby data is monitored according to user-specified criteria that, when met, cause Essbase to alert the user or system administrator.

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.

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.

unique member name A nonshared 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 deploying a new software release and moving applications, data, and provisioning information from an earlier deployment to the new deployment.

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. Popular user directories include Oracle Internet Directory (OID), Microsoft Active Directory (MSAD), and Sun Java System Directory Server.

user variable A variable that dynamically renders data forms based on a user's member selection, displaying only the specified entity. For example, a user variable named Department displays specific departments and employees.

user-defined attribute (UDA) An attribute, associated with members of an outline to describe a characteristic of the members, that can be used to return lists of members that have the specified associated UDA.

user-defined member list A named, static set of members within a dimension defined by the user.

validation The process of checking a business rule, report script, or partition definition against the outline to ensure that the object being checked is valid.

validation rules Rules used in FDM to enforce data integrity. For example, in FDM, validation rules ensure that certain conditions are met after data is loaded from FDM to the target application.

value dimension A dimension that is used to define input value, translated value, and consolidation detail.

variance The difference between two values (for example, between planned and actual values).

version A 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.

vertical application server cluster A cluster with multiple application server instances on the same machine.

view A year-to-date or periodic display of data.

visual cue A formatted style, such as a font or a color, that highlights specific data value types. 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.

WebLogic Server home A subdirectory of Middleware home containing installed files required by a WebLogic Server instance. WebLogic Server home is a peer of Oracle homes.

weight A 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 reusable 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.

workbook An entire spreadsheet file with many worksheets.

workflow 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 that 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).

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.

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 A 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 A software tool that identifies version number of the most up-to-date plug-in on the server.

ZoomChart A tool for viewing detailed information by enlarging a chart. A ZoomChart enables you to see detailed numeric information on the metric that is displayed in the chart.

Index