Oracle® Fusion Middleware

Installation and Configuration Guide for Oracle Business Intelligence Applications

11*g* Release 1 (11.1.1)

E16814-07

November 2012

Provides the steps to set up Oracle BI Applications, after exiting Oracle Fusion Applications Provisioning; explains how to perform system set up and functional configuration using Oracle BI Applications Configuration Manager.

ORACLE®

Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications 11g Release 1 (11.1.1)

E16814-07

Primary Author: P Brownbridge

Contributors: Oracle Business Intelligence Applications development, product management, and quality assurance teams.

Contents

Part II Functional Configuration for Oracle Business Intelligence Applications

4 Overview of Functional Configuration in Oracle BI Applications

5 Performing Functional Configuration

6 Administering and Maintaining Functional Configuration Data

Part III   Appendices

A   Supporting Tasks

B   Functional Configuration Task Reference

C   Dialog Reference for Oracle BI Applications Configuration Manager

Index

Preface

Oracle Business Intelligence Applications are comprehensive prebuilt solutions that deliver pervasive intelligence across an organization, empowering users at all levels — from front line operational users to senior management — with the key information they need to maximize effectiveness. Intuitive and role-based, these solutions transform and integrate data from a range of enterprise sources and corporate data warehouses into actionable insight that enables more effective actions, decisions, and processes.

Oracle BI Applications are built on Oracle Business Intelligence Suite Enterprise Edition, a comprehensive next-generation BI and analytics platform.

Documentation Accessibility

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

Access to Oracle Support

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

Audience

This document is intended for managers and implementors of Oracle BI Applications.

Related Documents

For related information, see the following documents:

- The Oracle Business Intelligence Applications chapter in the *Oracle Fusion Middleware Release Notes* for your platform:
  http://docs.oracle.com/cd/E23943_01/relnotes.htm

- *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*

- *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Data Warehouse Administration Console*

> **Note:** For information about System Requirements and Supported Platforms for Oracle Business Intelligence products, see "System Requirements and Certification".

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in This Release

This section describes the new features in Oracle BI Applications 11g Release 1 (11.1.1) that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications*.

This section contains the following topics:

- "New Features in Release 11.1.1.6.4"
- "New Features in Release 11.1.1.6.3"
- "New Features in Release 11.1.1.6.2"
- "New Features in Release 11.1.1.6"
- "New Features in Release 11.1.1.5.2"
- "New Features in Release 11.1.1.5"
- "System Requirements and Certification"

New Features in Release 11.1.1.6.4

New features in Release 11.1.1.6.4 that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications* include the following:

Oracle BI Applications Upgrade Tool Updates

The Oracle BI Applications Upgrade Tool has been simplified. For more information, see Section 3.4.3, "Setup Step: Apply Patches".

New Features in Release 11.1.1.6.3

New features in Release 11.1.1.6.3 that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications* include the following:

BI Extender Updates

Updates have been made to the BI Extender.

New Features in Release 11.1.1.6.2

New features in Release 11.1.1.6.2 that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications* include the following:

Informatica PowerCenter V9.1 HotFix 2

Oracle BI Applications Version 11.1.1.6.2 requires Informatica PowerCenter V9.1 HotFix 2.

Warehouse Component Deployment Assistant (WCA) Installs Informatica PowerCenter Services

(On UNIX platforms) Warehouse Component Deployment Assistant (WCA) installs Informatica PowerCenter Services V9.1 HotFix 2.

Oracle BI Repository (RPD) Trimming

Oracle BI Repository (RPD) Trimming is now a process that is performed in the BI Provisioning phase of the Oracle Fusion Applications installation.

Oracle BI Repository (RPD) Restoration

You must restore the 'Golden' Metadata Repository (RPD) for Oracle BI Applications as part of the post-installation setup steps (for more information, see Step 14 in Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications").

New DAC Physical Data Sources Connection and Informatica Relational Connections

A new connection FUSION_V1 replaces the previous three connections FUSION_V1_FSCM, FUSION_V1_HCM, and FUSION_V1_CRM.

New Manual Patches for ATGLite and FSM

New patches are available for ATGLite and FSM.

New Features in Release 11.1.1.6

New features in Release 11.1.1.6 that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications* include the following:

New Warehouse Component Deployment Assistant (WCA)

The new Warehouse Component Deployment Assistant (WCA) streamlines the post-installation setup for Oracle BI Applications by automating the configuration of many components. For more information about using WCA, see Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

New Features in Release 11.1.1.5.2

New features in Release 11.1.1.5.2 that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications* include the following:

Improvements to Oracle BI Applications Configuration Manager

The following updates were made to Oracle BI Applications Configuration Manager, as follows:

- On the Overview page, the Alerts pane only displays data relating to enabled Offerings.

- On the Edit Domain Member Mappings dialog, the new **Sync To Source** option enables you to automatically synchronize target domains with source domain values.

- On the Edit Domain Member Mappings dialog, you can now use the new **Batch Edit** option to update multiple target domain members at the same time.

- On the Manage Data Load Parameters page, the functionality relating to Global Parameters and Application Specific Parameters has been improved. For example, table column names and values have been made more intuitive, and the **Edit All** option enables you to update all Fact Groups and Dimension Groups that are associated with a Data Load Parameter.

New Features in Release 11.1.1.5

New features in Release 11.1.1.5 that are documented in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications* include the following:

New Oracle BI Applications Configuration Manager

Functional Configuration is performed in a Task-based guided format using Oracle BI Applications Configuration Manager and Oracle Functional Setup Manager (FSM). For each Offering that you are deploying, Oracle Functional Setup Manager provides a list of tasks that must be performed to configure that Offering. For more information, see Chapter 5, "Performing Functional Configuration".

Maintenance and Administration of Functional Configuration Data is performed using Oracle BI Applications Configuration Manager. For more information, see Chapter 6, "Administering and Maintaining Functional Configuration Data".

System Requirements and Certification

Refer to the system requirements and certification documentation for information about hardware and software requirements, platforms, databases, and other information. Both of these documents are available on Oracle Technology Network (OTN).

The system requirements document covers information such as hardware and software requirements, minimum disk space and memory requirements, and required system libraries, packages, or patches:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-requirements-100147.html

The certification document covers supported installation types, platforms, operating systems, databases, JDKs, and third-party products:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html

1

Overview

This section provides an overview to *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications*.

Tip: To get started quickly with Functional Configuration, follow the high-level steps in Section 5.2, "Roadmap for Functional Configuration".

1.1 Overview

Part I provides pre-installation requirements and post-installation steps for Oracle Business Intelligence Applications with a Fusion Applications source system, as follows:

- Before installing Oracle Business Intelligence Applications, review the pre-installation requirements in Chapter 2, "Pre-installation and Pre-deployment Requirements for Oracle BI Applications".

- After installing Oracle Business Intelligence Applications using the Oracle Fusion Applications installer, follow the mandatory post-installation or upgrade steps in Chapter 3, "Setting Up Oracle Business Intelligence Applications". For a high-level list of post-installation setup and upgrade steps, see Section 3.1, "Roadmap for Installing, Setting Up, and Upgrading Oracle BI Applications".

Part II explains how to use Functional Setup Manager (FSM) to perform Functional Configuration for Oracle Business Intelligence Applications, and how to use Oracle BI Applications Configuration Manager to administer and maintain setup data, as follows:

- Before starting Functional Configuration, familiarize yourself with the concepts, tools, and workflows that are explained in Chapter 4, "Overview of Functional Configuration in Oracle BI Applications".

- Enable your Offerings and select Feature Choices, create an Implementation Project, and perform the configuration Tasks to configure your Offerings, as described Chapter 5, "Performing Functional Configuration".

- Monitor and maintain your Functional Configuration setup data, as described Chapter 6, "Administering and Maintaining Functional Configuration Data".

Part III includes reference information, and supporting information that is used to provide Help topics for Oracle Business Intelligence Applications Configuration Manager, and Functional Setup Manager (FSM) used with Oracle BI Applications.

Part I

Setting Up Oracle Business Intelligence Applications

Part I covers installation and deployment requirements, and post-installation setup instructions for Oracle BI Applications. It contains the following sections:

- Chapter 2, "Pre-installation and Pre-deployment Requirements for Oracle BI Applications"
- Chapter 3, "Setting Up Oracle Business Intelligence Applications"

2

Pre-installation and Pre-deployment Requirements for Oracle BI Applications

This section covers pre-installation and pre-deployment options for Oracle BI Applications.

You should review this information before you begin the installation and deployment process. You should also read the guidelines for setting up Oracle Business Analytics Warehouse and read the database guidelines for the source OLTP databases that you are using.

This section contains the following topics:

- Section 2.1, "Oracle-Specific Database Guidelines for Oracle Business Analytics Warehouse"

- Section 2.2, "Additional Suggestions for Optimizing Oracle Performance in Oracle Business Analytics Warehouse"

- Section 2.3, "Creating Custom Indexes in Oracle Fusion Applications Database for Incremental Load Performance"

- Section 2.4, "Code Page and Data Movement Requirements"

Notes

- For information about requirements and certification, see "System Requirements and Certification".

- Oracle Business Analytics Warehouse is a database that contains dimensional schemas. Although it is technically possible to put the Oracle Business Analytics Warehouse in the same database as the transactional database, for performance reasons it is not recommended. The transactional database is optimized for an online transaction processing (OLTP) database, whereas Oracle Business Analytics Warehouse is optimized for an online analytical processing (OLAP) database. Reasons for not combining the two databases include:

  - The analytical queries interfere with normal use of the transactional database, which is entering and managing individual transactions.

  - The data in a transactional database is normalized for update efficiency. Transactional queries join several normalized tables and will be slow (as opposed to pre-joined, de-normalized analytical tables).

  - Historical data cannot be purged from a transactional database, even if not required for current transaction processing, because you need it for analysis. (By contrast, the analytical database is the warehouse for historical as well as current data.) This causes the transactional database to further slow down.

- Transactional databases are tuned for one specific application, and it is not productive to use these separate transactional databases for analytical queries that usually span more than one functional application.

- The analytical database can be specifically tuned for the analytical queries and Extract-Transform-Load (ETL) processing. Analytical database requirements are different from transactional database requirements.

- The Informatica Repository stores all of the Informatica object definitions for the ETL mappings that populate the Oracle Business Analytics Warehouse. It is a series of repository tables that are stored in a database, which can be a transactional, analytical, or a separate database.

2.1 Oracle-Specific Database Guidelines for Oracle Business Analytics Warehouse

Oracle-specific guidelines for Oracle Business Analytics Warehouse:

- To configure Oracle Business Analytics Warehouse on Oracle databases more easily, refer to the parameter template file init11gR2_FusionApplications_ template.ora. This files is located in \biapps\etl under the Oracle Home for BI.

  The parameter template file provides parameter guidelines based on the cost-based optimizer for Oracle 11g, and the guidelines will help you set up Oracle Business Analytics Warehouse for performance and growth. Use these guidelines as a starting point. You typically make changes based on your specific database sizes, data shape, server size (CPU and memory), and type of storage.

  Copy the appropriate template file into your $ORACLE_HOME/dbs directory. Then, review the recommendations in the template file, and make the changes based on your specific database configuration. The database administrator should make changes to the settings based on performance monitoring and tuning considerations.

  Note: The NLS_LENGTH_SEMANTICS parameter enables you to define byte-length or character-length semantics. Oracle BI Applications supports BYTE and CHAR values for this parameter. If you are using MLS characters, then you can add this parameter to the init11gR2_FusionApplications_template.ora file.

- Oracle recommends that you use the following database parameters for provisioning:

```
Common
======
audit_trail NONE
plsql_code_type NATIVE
nls_sort BINARY
open_cursors 500
session_cached_cursors 500
_b_tree_bitmap_plans FALSE
query_rewrite_integrity TRUSTED
job_queue_processes 10
star_transformation_enabled TRUE
parallel_max_servers 16

Sizeable Parameters
===================
processes 2500
sga_target 8GB
pga_aggregate_target 4GB
```

```
Optimizer Stats Gathering
=========================
Auto Stats Gathering Job None
```

- At a minimum, separate the data and index tablespaces. Create more tablespaces to separate heavily used tables and their indexes.

- Use the maximum block and page size available for tablespaces (for example, 32K), because it provides good overall performance and also does not impose low limits on the maximum size to which the tablespace can grow, as compared to 4K, 8K, and 16K sizes.

- If you are using multiple disk storage systems, then stripe the tablespace containers and files across as many disks as possible.

- Unbuffered (or 'raw') devices for tablespaces provide better performance in comparison to 'cooked' file systems (or files that are buffered through the operating system).

- RAID-5 is known to give a good balance of performance and availability.

- For Oracle databases, size the buffer pools based on content and size (number of tables and their sizes) of tablespaces.

- Allocate about two-thirds of the total available server memory to the database, assuming that no other application is running on the same server.

- Use Oracle Automated Storage Management (ASM) option for tablespaces as this provides better performance and manageability as compared to OS based general purpose file systems.

2.2 Additional Suggestions for Optimizing Oracle Performance in Oracle Business Analytics Warehouse

This section contains additional suggestions for optimizing performance for Oracle databases.

- Oracle database supports only binary sorting. If you are running an Oracle client, do one of the following:

  – Set the NLS_SORT parameter to BINARY.

  – Choose a NLS_LANG setting that includes binary.

  These settings are required for adequate performance from the dedicated Web client.

- Make sure that cost-based optimization is enabled in the Oracle development, test, and production databases and that statistics are kept up to date. Otherwise, the rule-based optimizer may be used.

- Create foreign keys in the Oracle database, but configure Oracle to not enforce the foreign key relationship. The existence of foreign keys will allow Oracle to better optimize certain queries. By turning off enforcement, the database load should not be negatively affected.

- Analyze application for occurrences of highly skewed data that is indexed. Create histogram statistics for these indexes to enable the optimizer to better perform queries.

- To increase data throughput between Oracle BI Server and Oracle, change SDU and TDU settings in listener.ora. The default is 2 KB and can be increased to 8 KB.

- On the server side, edit the listener.ora file. Under the particular SID_LIST entry, modify SID_DESC as follows:

```
SID_LIST_LISTENER =
   SID_LIST =
      SID_DESC = (SDU=16384)(TDU=16384)
         ORACLE_HOME = /.....)
         SID_NAME = SOLAP)
   )
)
```

- Make sure the temporary tablespace has adequate space.

- Set the number of log file groups to 4.

- Set the size of each log file to 10 MB.

- On the client side, edit the tnsnames.ora file. Modify the TNS alias by adding SDU= and TDU= as follows:

```
myhost_orcl.world=
   DESCRIPTION=(SDU=16384)(TDU=16384)
   ADDRESS = (PROTOCOL = TCP)(HOST=myhost)(PORT=1521))
CONNECT_DATA=(SID=ORCL))
```

2.3 Creating Custom Indexes in Oracle Fusion Applications Database for Incremental Load Performance

Oracle Fusion Applications database tables contain mandatory LAST_UPDATE_DATE columns, which are used by Oracle BI Applications for capturing incremental data changes. Some Oracle Fusion Applications source tables used by Oracle BI Applications do not have an index on the LAST_UPDATE_DATE column because the presence of the index may impede performance of some application flows or processes for a small set of Fusion Applications implementations. Since LAST_UPDATE_DATE indexes on such tables are used by Oracle BI Applications only, customers should manually create them in their source Fusion Applications environments. Such customizations have been approved by Oracle Fusion Application Development.

To create the index, use the following syntax:

```
CREATE index AP.OBIEE_<TABLE_NAME> ON <TABLE_NAME>(LAST_UPDATE_DATE) tablespace
<IDX_TABLESPACE>;
```

> **Note:** Indexes created with the DDL in this section have the prefix OBIEE_. This prefix does not follow standard Fusion Applications index naming conventions. Therefore, Autopatch might fail during future upgrades. In such cases, the indexes with the OBIEE_ prefix should be dropped and Autopatch restarted. Use FND_STATS to compute statistics on the newly created indexes and update statistics on newly indexed table columns in the Oracle Fusion Applications database.

2.4 Code Page and Data Movement Requirements

Oracle Business Analytics Warehouse can be deployed in various code page environments and supports global deployments. Data movement is supported in the following source database and data warehouse configuration modes:

- Unicode to Unicode - **Note**: This is the only supported configuration mode for Oracle Business Intelligence Applications 11*g* Release 1 (11.1.1).

- Code page (multi-byte or single-byte) to Unicode.

- Code page to code page (where the code pages are the same).

Oracle BI Applications uses Informatica PowerCenter to perform extract, transform and load (ETL) routines to move data from source database(s) to Oracle Business Analytics Warehouse.

During the installation and configuration procedures described in this chapter, you will configure your environment to enable accurate data movement. Use the guidelines and references noted below to determine values for these settings that are appropriate for your environment:

- Determining the source to target configuration mode. Consult your database administrator to determine the code page that your source OLTP database uses. Based on the type of data that will be moved from one or more source databases to Oracle Business Analytics Warehouse, determine what code page to use for Oracle Business Analytics Warehouse. When determining what code page to use for Oracle Business Analytics Warehouse, consider future requirements for storing data.

- For accurate data movement from source database to target, the code page of Oracle Business Analytics Warehouse (target) must be a superset of the code page of the source database. Informatica considers a code page to be a superset of another code page when the code page contains all the characters encoded in the other code page and additional characters not encoded in the other code page.

  Note: To enable data movement from source(s) to Oracle Business Analytics Warehouse, you set relaxed Code Page Validation for the Integration Services. You must ensure that the target code page is a superset of the source code page for accurate data movement.

- Setting the SiebelUnicodeDB property. If your source to target configuration mode for data movement is Unicode to Unicode, then you will set a custom property called SiebelUnicodeDB on the Integration Services. Configuration modes of code page to Unicode or code page to code page do not require this property to be set.

- Determining the Data Movement Mode. Before you deploy Oracle BI Applications, you must determine what data movement mode to use (ASCII or Unicode) for the PowerCenter Integration Service.

  Note: Oracle BI Applications 11*g* Release 1 (11.1.1) with Oracle Fusion Applications only supports 'Unicode to Unicode' as a Data Movement mode.

  The Character Data Movement Mode is an Informatica PowerCenter Integration Service option that you choose based on whether you want to move single-byte or multi-byte data. Choose Unicode if non-ASCII characters have to be moved. Otherwise, choose ASCII. The Data Movement Mode option is configurable and can be reset after installation.

  To set the Data Movement Mode, log into Informatica Administrator, select the Integration Service, then click the Properties tab, then display the General Properties tab, and set the DataMovementMode value.

■ Determining code pages for Informatica PowerCenter components. In addition to source and target code pages, Informatica PowerCenter uses code pages for PowerCenter Client, the Integration Service, the Informatica Repository, and PowerCenter command line programs (pmcmd and pmrep, which are used by DAC to communicate with PowerCenter Services). Carefully review 'Chapter 22: Understanding Globalization,' in the Informatica PowerCenter Administrator Guide, particularly if your environment requires the Data Mode Movement mode to be set to UNICODE. The section discusses code page compatibility and code page requirements for the Informatica components.

■ Setting environment variables. You must manually set the appropriate environment variables for UNIX environments. In addition, the Informatica installer requires the appropriate locale to be set on UNIX machines. Use LANG, LC_CTYPE or LC_ALL to set the UNIX code page. For more information, see the topic titled 'Configuring Environment Variables,' in 'Chapter 2: Before You Install,' in the Informatica PowerCenter Installation Guide.

You also need to set the environment variable NLS_LANG.

3

Setting Up Oracle Business Intelligence Applications

This chapter explains how to set up Oracle BI Applications, after exiting Oracle Fusion Applications Provisioning.

This chapter primarily covers setting up Oracle BI Applications in a single BI domain with the WebLogic Administration Server and Managed Server in the same domain home. However, some setup tasks also apply to enterprise deployments (for more information, see Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications").

Note: For information about deploying Oracle BI Applications components for high availability in Oracle Fusion Applications environments, see "Configuring Oracle Business Intelligence Applications" in *Oracle Fusion Applications Customer Relationship Management Enterprise Deployment Guide*.

This chapter contains the following sections:

- Section 3.1, "Roadmap for Installing, Setting Up, and Upgrading Oracle BI Applications"

- Section 3.2, "Before You Set Up Oracle BI Applications"

- Section 3.3, "Overview of Setting Up Oracle BI Applications"

- Section 3.4, "Tasks for Setting Up Oracle BI Applications"

3.1 Roadmap for Installing, Setting Up, and Upgrading Oracle BI Applications

Do one of the following:

- If you have just installed Fusion Applications Release 5 (or you installed an earlier release of Fusion Applications but you did not deploy Oracle BI Applications), then you must set up Oracle BI Applications components, as described in Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications."

- If you have just upgraded Fusion Applications to Release 5, and you had Oracle BI Applications 11.1.1.6.0 or 11.1.1.6.2 deployed, then you must perform additional steps to upgrade Oracle BI Applications components to Release 11.1.1.6.3.

  For additional steps to upgrade Oracle BI Applications components to Release 11.1.1.6.3, refer to Tech Note `1471195.1` on My Oracle Support.

3.1.1 High-level Tasks for Setting Up Oracle BI Applications

Overview to Setting Up Oracle BI Applications

When you set up Oracle BI Applications, you can use Warehouse Component Deployment Assistant (WCA) to configure some components, or you can configure these components manually. In addition, there are other components that you must configure manually whether you use WCA or not. For details, refer to the **List of High Level Steps for Setting Up Oracle BI Applications** below.

Choose whether to use WCA as follows:

- You typically use WCA to configure some components if you are deploying Oracle BI Applications on a UNIX or Linux platform, and into a single BI domain (on UNIX) with the WebLogic Administration Server and Managed Server in the same domain home. If you use WCA to configure BI Applications, then you perform the steps listed below except for Steps 9 and 15.

- You typically choose not to use WCA if you are deploying Oracle BI Applications on a BI Domain on a Windows platform, or you are deploying Oracle BI Applications in an enterprise deployment. If you are not using WCA to deploy BI Applications, then you must perform the steps listed below except for Step 8.

List of High Level Steps for Setting Up Oracle BI Applications

To install and set up Oracle BI Applications, you must complete the following tasks, in order. Each high-level task breaks down into a list of detailed steps.

Tip: Before you begin, obtain the BI repository and BI Domain details from the team that installed Oracle Fusion Applications, as described in Section 3.2, "Before You Set Up Oracle BI Applications."

1. Create databases for Oracle Business Analytics Warehouse, DAC Repository, Informatica Domain, and Informatica Repository, as described in Section 3.4.1, "Setup Step: Create Required Databases."

2. Create the following schemas using RCU:

   - Oracle BI Applications schemas (DAC Repository schema and Oracle Business Analytics Warehouse schema) by following the steps in Section 3.4.2.1, "Creating Oracle BI Applications Schemas Using Oracle BI Applications RCU".

   - If you have implemented the Sales Prediction Engine module of Oracle Fusion Customer Relationship Management, then run the Sales Prediction Engine RCU to create Oracle Business Analytics Warehouse objects by following the steps in Section 3.4.2.2, "Creating Oracle Business Analytics Warehouse Objects with the Oracle Sales Prediction Engine RCU."

3. Apply all required Oracle BI Applications patches for this release, as described in Section 3.4.3, "Setup Step: Apply Patches."

4. Install the Oracle BI Administration Tool, as described in Section 3.4.4, "Setup Step: Install Oracle BI Administration Tool."

5. Install and configure database connectivity software, as described in Section 3.4.5, "Setup Step: Install and Configure Database Connectivity Software."

6. Grant user access to Oracle BI Applications components, as described in Section 3.4.6, "Setup Step: Grant User Access to Oracle BI Applications Components."

7. Create a user for running ETL, as described in Section 3.4.7, "Setup Step: Create a User for ETL."

8. (Setup Step for WCA users only) If you are deploying on UNIX and are not deploying Oracle BI Applications in an enterprise deployment, then you may use the WCA utility to install Informatica PowerCenter Services and to configure Oracle BI Applications components, as follows:

a. Run WCA as described in Section 3.4.8.1, "Setup Step: Configure Oracle BI Applications components using Warehouse Component Deployment Assistant (WCA) in a UNIX Environment."

b. Validate the setup as described in Section 3.4.8.2, "Setup Step: Validate the Oracle BI Applications Component Configuration."

c. Restart the BI components, as described in Section 3.4.8.3, "Restart BI Components."

After completing Steps 8a, 8b, and 8c, skip to Step 10.

Note: WCA automatically configures the components that are configured manually in Steps 7 and 17. For more information about WCA, see Section A.6, "About Warehouse Component Deployment Assistant (WCA)."

9. (Manual Setup Step) If you are deploying Oracle BI Applications in a Windows environment or you are installing Oracle BI Applications in an enterprise deployment, then you must perform the following steps in Section 3.4.9, "Setup Step: Manually Set Up Oracle BI Applications Components" before proceeding to Step 10:

a. Install and Set Up Informatica PowerCenter Services by performing the following steps in:

i - Install and configure Informatica PowerCenter Services, as described in Section 3.4.9.1, "Install and Set Up Informatica PowerCenter Services Manually."

ii - Create the Repository Service, as described in Section 3.4.9.2, "Setup Step: Creating the Informatica Repository Service."

iii - Create the Integration Service, as described in Section 3.4.9.3, "Setup Step: Creating the Informatica Integration Service."

iv - Load the pre-built Informatica repository, as described in Section 3.4.9.4, "Setup Step: Load the Prebuilt Informatica Repository."

v - Copy source files to the Informatica PowerCenter Services machine, as described in Section 3.4.9.5, "Setup Step: Copying Source Files to the Informatica PowerCenter Services machine."

vi - Set the PowerCenter Relaxed Code Page Validation option, as described in Section 3.4.9.6, "Setup Step: Setting PowerCenter Integration Services Relaxed Code Page Validation."

vii - Set the PowerCenter Custom Properties, as described in Section 3.4.9.7, "Setup Step: Setting PowerCenter Integration Services Custom Properties."

b. Configure the Oracle BI Connection Pools, as described in Section 3.4.9.9, "Setup Step: Configuring the Oracle BI Connection Pools."

c. Extend the BI Domain, as described in Section 3.4.9.8, "Setup Step: Extending the BI Domain."

10. Configure Oracle HTTP Server, as described in Section 3.4.10, "Setup Step: Configure Oracle HTTP Server."

11. Configure single sign-on (SSO) for Oracle BI Applications Configuration Manager and Functional Setup Manager, as described in Section 3.4.11, "Setup Step: Configure SSO and Portlet Provider for Oracle BI Applications Configuration Manager and Functional Setup Manager."

12. Install and configure Informatica PowerCenter Client tools and set up the Informatica Domain and Repository, as follows:

    a. Install PowerCenter Client Tools, as described in Section 3.4.12.1, "Install PowerCenter Client Tools."

    b. Set up the Informatica Domain and Repository in PowerCenter Client Tools, as described in Section 3.4.12.2, "Setup Step: Accessing the Informatica Domain and Repository from Informatica Client Tools."

    c. Create the Repository Administrator User in the Native Security Domain, as described in Section 3.4.12.3, "Setup Step: Creating the Repository Administrator User in the Native Security Domain."

13. Install DAC Client on a Windows-based machine where the Informatica PowerCenter Client Tools are installed, as described in Section 3.4.13, "Setup Step: Install and Set Up DAC Client."

14. Create an ODBC DSN to the Oracle BI Server, as described in Section 3.4.14, "Setup Step: Create an ODBC DSN to the Oracle BI Server."

15. (Manual Setup Step) If you are using WCA to configure Oracle BI Applications components, then skip to Step 16. If you performed manual Step 9 and are configuring Oracle BI Applications components manually, then you must perform the following steps before proceeding to Step 16:

    a. Register Informatica in DAC and Configure DAC Integration Settings, as described in Section 3.4.15.1, "Setup Step: Register Informatica in DAC and Configure DAC Integration Settings".

    b. Configure Physical Data Source Connections in DAC, as described in Section 3.4.15.2, "Setup Step: Configuring Physical Data Source Connections in DAC".

    c. Configure Relational Connections in Informatica, as described in Section 3.4.15.3, "Setup Step: Configuring Relational Connections in Informatica".

    d. Set the SiebelUnicodeDB Property in Informatica Integration Services, as described in Section 3.4.15.4, "Setup Step: Setting the SiebelUnicodeDB Property in Informatica Integration Services".

    e. Enable User Currency Preference Settings, as described in Section 3.4.15.5, "Setup Step: Enabling User Currency Preference Settings".

    f. Register Source Systems in Oracle BI Applications Configuration Manager, as described in Section 3.4.15.6, "Setup Step: Registering Source Systems".

16. Enable Offerings in Oracle BI Applications Configuration Manager, as described in Section 3.4.16, "Setup Step: Enable Offerings for Deployment".

17. Set languages for ETL data load, as described in Section 3.4.17, "Setup Step: Set Languages for Data Load".

18. Edit Preferred Currency Display Names, as described in Section 3.4.18, "Setup Step: Editing Preferred Currency Display Names and Enable Document Currency".

19. Restore the BI Metadata Repository File (RPD), as described in Section 3.4.19, "Setup Step: Restore the BI Metadata Repository File for Oracle BI Applications".

20. Run Domains ETL, as described in Section 3.4.20, "Setup Step: Run Domains ETL."

21. Verify Duty Roles required for Oracle BI Applications, as described in Section 3.4.21, "Setup Step: Verifying Duty Roles required for Oracle BI Applications."

22. If you want to customize Oracle-delivered analyses or dashboards, then to find out how read Section 3.4.22, "About Customizing Oracle-Delivered Oracle BI Presentation Catalog Content."

23. Next Steps:

    After you complete the setup tasks described above, read Section 3.4.23, "Next Steps."

3.2 Before You Set Up Oracle BI Applications

Before You Begin

Before you can begin to set up Oracle Business Intelligence Applications, you need to obtain the following three values from the team who installed Oracle Fusion Applications:

- The password to access the Oracle BI repository.

- The host name where Oracle Business Intelligence is installed.

- The location of the BI Domain.

Obtaining the Oracle BI Repository and Host Name

Two Oracle Fusion Applications provisioning dialogs impact Oracle Business Intelligence Applications. Before you complete the setup tasks for Oracle Business Intelligence Applications, consult with the team who installed Oracle Fusion Applications to obtain the values for **Host** and **RPD password**.

- The Installation Location dialog in the Oracle Fusion Applications Provisioning Wizard

  At this dialog, a **RPD password** was entered to access the Oracle BI repository. Oracle Fusion Applications provisioning creates this password and makes it available so that Oracle Business Intelligence Applications can access the metadata in the repository in the Oracle Fusion Applications environment.

- The Oracle Business Intelligence Configuration dialog

  At this dialog, the **Host** where the Oracle Business Intelligence product was to be installed was entered.

Obtaining the Location of the Oracle BI Domain

At the conclusion of the Oracle Fusion Applications provisioning process for Oracle Business Intelligence, the following entities are available:

- WebLogic domain.

- Oracle BI Web application (J2EE) components.

- Oracle BI Server and BI Presentation Services, which are deployed on the machine that hosts the domain.

The resulting environment is referred to as the "Business Intelligence domain" or "BI Domain." Consult with the team who installed Oracle Fusion Applications for the location of the BI Domain.

The following default values are used:

- Admin Server port - 10201.

- Managed Server port - 10217.

- WLS Admin user - FAadmin.

Before you set up Oracle BI Applications, do the following:

- Review the Certification Document for Oracle BI Applications and ensure that your environment meets the listed requirements:

  http://www.oracle.com/technology/software/products/ias/files/
  fusion_certification.html

- Install Oracle BI Applications using the Oracle Fusion Applications installation and provisioning process. See *Oracle Fusion Applications Installation Guide*.

- Read Chapter 2, "Pre-installation and Pre-deployment Requirements for Oracle BI Applications."

- Read Section 3.3, "Overview of Setting Up Oracle BI Applications".

After you complete the tasks listed above, proceed to Section 3.4, "Tasks for Setting Up Oracle BI Applications" for detailed instructions on how to set up Oracle BI Applications.

3.3 Overview of Setting Up Oracle BI Applications

This section provides an overview for setting up Oracle BI Applications, and includes the following topics:

- Section 3.3.1, "About Setting Up Oracle BI Applications"

- Section 3.3.2, "About the Components in an Oracle BI Applications Environment"

- Section 3.3.3, "Summary of Home Directories"

3.3.1 About Setting Up Oracle BI Applications

Oracle Business Intelligence Enterprise Edition is installed and provisioned as part of the Fusion Applications installation and provisioning process. The BI Provisioning step creates a WebLogic domain, the BI Web application (J2EE) components, and the BI Server and BI Presentation Services, which are deployed on the machine that hosts the BI Domain. During the Fusion Applications installation, the software components of Oracle BI Applications are installed in the Business Intelligence Oracle Home but are not set up. To set up Oracle BI Applications, you must follow the steps in Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Warehouse Component Deployment Assistant (WCA) is a script that you can use to install and configure a number of components for Oracle BI Applications in UNIX environments. For example, WCA installs Informatica PowerCenter Services, and configures a Repository Service and an Integration Service. For guidance on deciding whether to use WCA, refer to Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

For a complete list of components that are configured by WCA, see Section A.6, "About Warehouse Component Deployment Assistant (WCA)."

The following Oracle Business Intelligence provisioning occurs during Fusion Applications installation and provisioning:

- Analytics View Objects and the BI Broker servlet are deployed to the appropriate Fusion Applications WebLogic domains.

- The BI metadata repository file (OracleBIApps.rpd) is trimmed during BI Provisioning to contain only those RPD projects for Oracle BI Applications that are relevant for the installed Fusion Applications Offerings.

- The following are configured in the BI metadata repository file (OracleBIApps.rpd):

  - WebLogic connection pools for the three applications pillars

  - SQL Bypass connections FSCM_OLTP, CRM_OLTP, HCM_OLTP, and FBI_OLTP

  - Static Variables

- The security roles (or Duty Roles) required by Oracle BI Applications and Oracle Fusion Transactional Business Intelligence are provisioned.

3.3.2 About the Components in an Oracle BI Applications Environment

The following figure shows the main tools and components that you set up in an Oracle BI Applications environment:

Figure 3–1 Tools and Components in an Oracle Business Intelligence Applications Environment
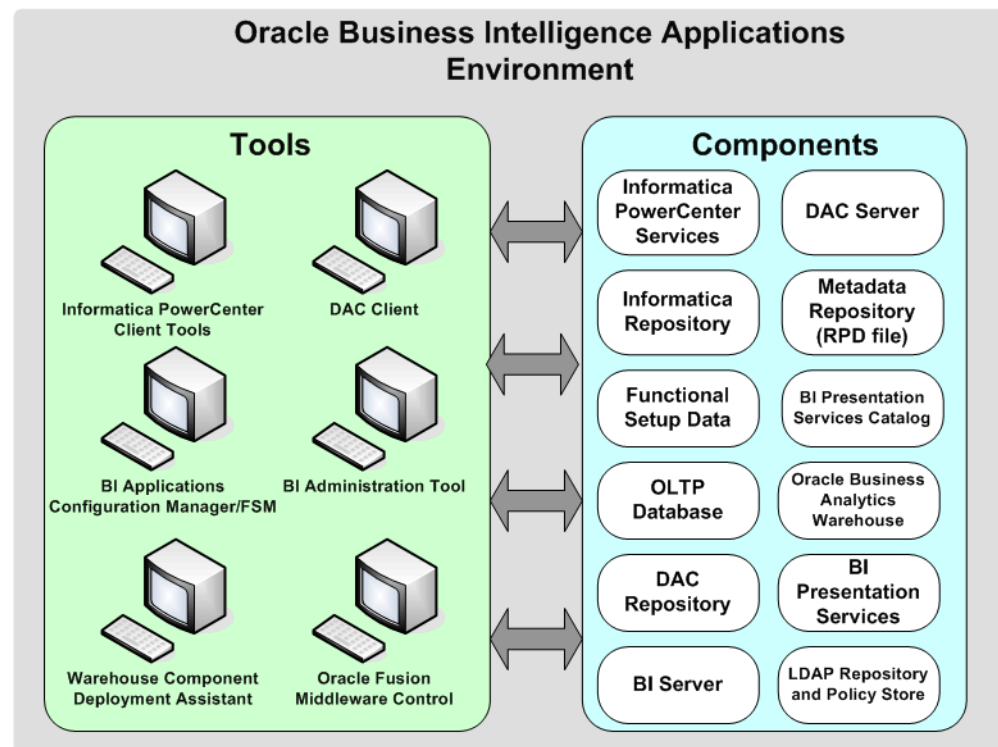
Tools

- **Informatica PowerCenter Client tools** - third-party Informatica PowerCenter Client tools that you use to set up the Informatica platform, and make ETL

customizations if required. For example, PowerCenter Client Tools include Informatica Administrator, and PowerCenter Designer.

- **Oracle BI Applications Configuration Manager** - a Web application used to perform system setup, and manage functional setup data.

  Functional Setup Manager (FSM) - a Web application leveraged by Oracle BI Applications Configuration Manager that is used to configure Oracle BI Applications Offerings using guided Tasks.

- **Warehouse Component Deployment Assistant (WCA)** - a tool that you can use to set up some Oracle BI Applications components (for example, DAC, Informatica PowerCenter, and Oracle BI Applications Configuration Manager). For guidance on deciding whether to use WCA, refer to Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications". For a complete list of components that are configured by WCA, see Section A.6, "About Warehouse Component Deployment Assistant (WCA)."

- **DAC Client** - orchestrates the ETL process and configures the DAC Server.

- **BI Administration Tool** - a Windows client tool used to manage BI metadata repositories (that is, RPD files).

- **Fusion Middleware Control** - a Web application that enables you to manage system components by performing tasks such as monitoring status, starting and stopping processes, scaling out, resolving issues, and configuring components.

Components

- **Informatica PowerCenter Services** - third-party Informatica PowerCenter server that is used to perform ETL. Informatica PowerCenter Services is managed by Informatica Administrator, which is a Web application.

- **Informatica Repository** - prebuilt ETL logic for each BI Application.

- **Functional Setup Data** - Functional Configuration choices made in Functional Setup Manager and Oracle BI Applications Configuration Manager.

- **OLTP database** - transactional data in the Fusion Applications source system.

- **DAC Repository** - stores the metadata that represents Oracle Business Analytics Warehouse processes.

- **Oracle BI Server** - provides the query and data access capabilities at the heart of Oracle Business Intelligence, and provides services for accessing and managing the enterprise semantic model (stored in a file with an .rpd extension).

- **DAC Server** - Data Warehouse Administration Console server that is used to orchestrate ETL processes.

- **Metadata Repository** - RPD file that stores the BI metadata.

- **BI Presentation Services Catalog** - prebuilt analyses and dashboards, common files for Oracle BI Applications.

- **Oracle Business Analytics Warehouse** - the target database into which source system data is loaded.

- **Oracle BI Presentation Services** - provides the framework and interface for the presentation of Business Intelligence data to Web clients.

- **LDAP Repository and Policy Store** - provide Users and Groups with appropriate security privileges to enable them to access dashboards and analyses (via Job Roles and Duty Roles).

3.3.3 Summary of Home Directories

The following table defines the main Home directories for the components in an Oracle BI Applications deployment:

Table 3–1   Summary of Home Directories for Oracle BI Applications

Home Directory	Description
DOMAIN_HOME	Location of the Oracle WebLogic BI Domain (for example, /instance/domains/*<hostname>*/BIDomain).
INFA_HOME	Location of the user-specified Informatica installation directory on the machine that hosts PowerCenter Client or PowerCenter Services.
	For PowerCenter Client, this is a directory on the Windows machine that also hosts DAC Client. For example, c:\informatica\91\.
	For PowerCenter Services, this is a directory on a UNIX or Windows machine that hosts PowerCenter Services (also referred to as the Informatica Server). For example, /informatica/91/.
MW_HOME	Location of the Oracle Fusion Middleware home (for example,\FMW_11g\). It includes the WebLogic home and one or more Oracle homes.
ORACLE_HOME (for Oracle Business Intelligence)	This is the root directory of Oracle Business Intelligence. For example, the Oracle home for Oracle Business Intelligence is typically *MW_HOME*/Oracle_BI1.
ORACLE_INSTANCE	Location of the Oracle Instance home (for example, *MW_HOME*\Oracle_BI1\instances\instance1).

3.4 Tasks for Setting Up Oracle BI Applications

This section provides detailed tasks for setting up Oracle BI Applications.

Note: You must perform the tasks in this section in the sequence described in Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Before you complete the tasks described in this section, you must ensure that your environment meets the prerequisites and that you understand the recommendations described in Section 3.2, "Before You Set Up Oracle BI Applications."

3.4.1 Setup Step: Create Required Databases

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Install Oracle Database 11.2.0.2 or later for Oracle Business Analytics Warehouse, the DAC Repository, the Informatica Repository, and the Informatica Domains Repository. For more information, including the list of supported database versions, see the Oracle Fusion Applications system requirements and supported platforms documentation.

Ensure that your Oracle Database instance is configured for the ALT32UTF8 character set. Also, this release supports the data movement mode Unicode to Unicode. For more information, see Section 2.4, "Code Page and Data Movement Requirements."

For the database that hosts Oracle Business Analytics Warehouse, perform the following additional steps:

- Apply database patches 9739315 and 9255542.

- For additional database guidelines for Oracle Business Analytics Warehouse, see Chapter 2, "Pre-installation and Pre-deployment Requirements for Oracle BI Applications".

For prerequisite information for the Informatica Repository and Informatica Domains Repository databases, refer to Informatica documentation.

3.4.2 Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Oracle BI Applications requires the following database schemas, which you create with the Oracle BI Applications edition of the Repository Creation Utility (RCU):

- Oracle BI Data Warehouse Administration Console schema

- Oracle Business Analytics Warehouse schema

To create these two schemas, follow the steps in Section 3.4.2.1, "Creating Oracle BI Applications Schemas Using Oracle BI Applications RCU".

If your organization has deployed the Sales Prediction Engine module of Oracle Fusion Customer Relationship Management, you must also create Oracle Business Analytics Warehouse database objects using the Sales Prediction Engine RCU. For information about creating the Sales Prediction Engine RCU, see Section 3.4.2.2, "Creating Oracle Business Analytics Warehouse Objects with the Oracle Sales Prediction Engine RCU".

3.4.2.1 Creating Oracle BI Applications Schemas Using Oracle BI Applications RCU

Use the Oracle BI Applications RCU to create the Oracle BI Data Warehouse Administration Console and Oracle Business Analytics Warehouse schemas, which are required by Oracle BI Applications.

The Oracle BI Applications RCU zip file named rcuHomeBIApps.zip is located in the Fusion Applications software repository under installers/biapps_rcu/<OS - either Linux or Window>. (The Fusion Applications software repository is obtained when you download and unzip the Oracle Fusion Applications 11g parts available in the Oracle Fusion Applications Media Pack on the Oracle Software Delivery Cloud.) Extract this zip file into a folder, for example, BIA_RCU_HOME.

Before you create Oracle BI Applications schemas using the Oracle BI Applications RCU, ensure that you understand the following:

- The Oracle BI Applications RCU supports Oracle Database 11.2 and later, as described in Section 3.4.1, "Setup Step: Create Required Databases." The database in which you create the RCU schemas must meet the minimum requirements specified in the Certification Document for Oracle BI Applications:

  http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

- To create schemas with RCU on Oracle databases, you must possess the DBA privilege and be logged in as `SYSDBA` (for example, as user `SYS`). Although you can use another database user that has been granted the SYSDBA role, this is not recommended. Using database users other than SYS requires additional steps, which are described in "RCU Requirements for Oracle Databases" in *Oracle Fusion Middleware System Requirements and Specifications*.

- When you run RCU, you create databases, schemas, and tablespaces using `CREATE ANY SCHEMA`, `ALTER ANY SCHEMA`, `CREATE TABLE`, `CREATE TABLESPACE`, and so on. RCU updates the `schema_version_registry` by writing entries to the `system.schema_version_registry$` system table.

Pre-requisites

Before you create Oracle BI Applications schemas using the Oracle BI Applications RCU, you must do the following:

- On the database where you intend to create the Oracle BI Applications schemas, ensure the following grant is given to `PUBLIC`:

  `grant execute on UTL_FILE to PUBLIC;`

- Before running the Oracle BI Applications RCU to create database schemas, you must transfer the following dump files (.dmp) to a directory with global write access (RCU writes log files to this directory) on the machine that hosts your database:

  - obia.dmp

    This file is required for Oracle Business Analytics Warehouse, and includes the Oracle BI Applications Configuration Manager repository and the Oracle Functional Setup Manager repository.

  - dac.dmp

    This file is required for Oracle Business Intelligence Data Warehouse Administration Console, and includes the Oracle Business Intelligence DAC repository.

  When you download and extract the Oracle Business Intelligence Applications RCU .zip file, the dump files are located in the following directory:

  `BIA_RCU_HOME/rcu/integration/biapps/schema`

  where `BIA_RCU_HOME` is the directory where you unzipped the Oracle BI Applications RCU installation package.

To create the Oracle BI Applications database schemas using the Oracle BI Applications RCU:

1. Access the `bin` directory in the `BIA_RCU_HOME`.

2. Start RCU, as follows:

   - UNIX:

     `./rcu`

   - Windows:

     `rcu.bat`

   The Welcome dialog opens.

3. In the Welcome dialog, click **Next**.

4. In the Create Repository dialog click **Create**, and then click **Next**.

5. In the Database Connection Details dialog, specify the connection details for the database in which you want to store the schemas.

Specify the following database connection details:

Table 3–2   Specifying Database Connection Details for Oracle BI Applications RCU

Field Name	Action
Database Type	Click **Oracle Database**. Other database types are not supported in this release.
Host Name	Specify the fully qualified host name of the machine hosting the database. For example, you might specify `machine12345.example.com`.
Port	Specify the number of the communication port for the database. The default port number for Oracle databases is 1521.
Service Name	Specify the database account name or the Service Name for the database. For example, you might specify `BIA11G`.
Username	Specify the username for accessing the machine that hosts the database. For example, you might specify `SYS`.
Password	Specify the password for the username specified in the **Username** field.
Role	If this value is not SYS, then you must select **SYS**. The default value is selected automatically when you enter a username in the **Username** field.

6. Click **Next** to display the Checking Prerequisites dialog.

7. After the prerequisite checks conclude with no errors, click **OK** to dismiss the dialog and proceed to the Select Components dialog.

8. In the Select Components dialog, near the top of the dialog, select **Create a new Prefix**.

The default prefix is DEV. You can provide another prefix, if preferred.

RCU automatically creates schema names in the format *prefix_schemaname*. For example, if you enter the prefix DEV11G, RCU creates the Oracle Business Analytics Warehouse schema with the name DEV11G_DW. Make a note of these schema names and the prefix values from this dialog.

9. Expand the **Oracle BI Applications Schemas** group, and then select the **Oracle Data Warehouse Administration Console** (DAC), and the **Oracle Business Analytics Warehouse** (DW) schemas only.

Note: Do not select the Oracle Transactional BI (OTBI) schema.

10. Click **Next** to display the Checking Prerequisites dialog.

After the prerequisite checks conclude with no errors, click **OK** to proceed to the Schema Passwords dialog.

11. In the Schema Passwords dialog, specify and confirm a password for the schemas, then click **Next** to proceed to the Custom Variables dialog.

12. In the Custom Variables dialog, enter the locations of the dump files in the **Value** field for each schema, then click **Next** to proceed to the Map Tablespaces dialog.

    For example, if the dump files are located in a Linux directory named `/tmp` on the database machine, enter `/tmp` in the **Value** field.

13. In the Map Tablespaces dialog, confirm the schema names.

    For more information, especially for tablespace recommendations, see Section 2.1, "Oracle-Specific Database Guidelines for Oracle Business Analytics Warehouse."

14. Click **Next** to create the tablespaces for the schemas.

    After the tablespaces are created with no errors, click **OK** to proceed to the Summary dialog.

15. In the Summary dialog, click **Create** to start the schema creation process and display the Create dialog.

    When the schemas are created with no errors, the Completion Summary dialog is displayed.

16. In the Completion Summary dialog, click **Close**.

3.4.2.2 Creating Oracle Business Analytics Warehouse Objects with the Oracle Sales Prediction Engine RCU

Use the Sales Prediction Engine RCU to create Oracle Business Analytics Warehouse database objects (ODM tables, views, packages, ILS Processing tables, and so on) in Oracle Business Analytics Warehouse. This is required if the Sales Prediction Engine module of Oracle Fusion Customer Relationship Management was deployed.

Before you create Oracle Business Analytics Warehouse database objects, ensure that you have a supported Oracle Business Analytics Warehouse that contains the Oracle BI Applications schemas. See Section 3.4.2.1, "Creating Oracle BI Applications Schemas Using Oracle BI Applications RCU."

To create the Sales Prediction Engine schema with the Sales Prediction Engine RCU, follow the steps in "Setting Up Sales Prediction Engine" in the Customer Relationship Management chapter of *Oracle Fusion Applications Post-Installation Guide*.

3.4.3 Setup Step: Apply Patches

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

You must use the Oracle BI Applications Upgrade Tool to apply the patches for Functional Setup Manager and ATGLite. The Oracle BI Applications Upgrade Tool is a command line utility that is available in the following Oracle Home for BI location:

```
ORACLE_HOME/biapps/tools/lib/biappsrepositoryutil.jar
```

Run the Oracle BI Applications Upgrade Tool, as follows:

1. Navigate to the $ORACLE_HOME/biapps/tools/lib folder.

2. Set the JAVA_HOME, ORACLE_HOME and PATH variables, as follows:

```
ORACLE_HOME=$MW_HOME/Oracle_BI1
JAVA_HOME=$ORACLE_HOME/jdk
PATH=$JAVA_HOME/bin:$PATH
```

3. Run the following command:

```
$ORACLE_HOME/jdk/bin/java -jar biappsrepositoryutil.jar upgradeALL
bi.oracle.home=$ORACLE_HOME work.dir= /workfolder dw.user=$DW_USER
dw.password=$DW_USER_PASSWORD dw.conn.url=$HOSTNAME:$DB_PORT/$GLOBAL_DB_NAME
atg.db.seed.url=$HOSTNAME:$DB_PORT/$GLOBAL_DB_NAME
fsm.db.seed.url=$HOSTNAME:$DB_PORT/$GLOBAL_DB_NAME
biacm.db.seed.url=$HOSTNAME:$DB_PORT/$GLOBAL_DB_NAME dac.repo.user=$DAC_USER
dac.repo.password=$DAC_USER_Password dac.conn.url=$HOSTNAME:$DB_PORT/$GLOBAL_
DB_NAME db.sys.user=sys db.sys.password=$SYS_PASSWORD
db.conn.sid.url=$HOSTNAME:$DB_PORT:$ORACLE_SID"
```

Where:

bi.oracle.home - BI Oracle Home full path.

work.dir - working directory to be used for the run. Log files will be written to this directory.

db.sys.user - Database System User Name for the Oracle Business Analytics Warehouse database.

db.sys.password - Database System User Password (if this is not provided in the command line then the tool will request this at runtime).

db.conn.sid.url - DAC database connection URL (Format - *host:port:SID*).

dac.repo.user - DAC Repository schema user name (User should be the DAC Schema Owner).

dac.repo.password - DAC Repository schema user password (if this is not provided in the command line then the tool will request this at runtime).

dac.conn.url - DAC Repository connection URL (Format: *host:port/serviceName*).

dw.user – Oracle Business Analytics Warehouse schema user name (User should be the DW Schema Owner).

dw.password – Oracle Business Analytics Warehouse schema user password (if this is not provided in the command line then the tool will request this at runtime).

dw.conn.url – Oracle Business Analytics Warehouse connection URL (Format: *host:port/serviceName*).

atg.db.seed.url - Oracle Business Analytics Warehouse connection URL (Format: *host:port/serviceName*).

fsm.db.seed.url - Oracle Business Analytics Warehouse connection URL (Format: *host:port/serviceName*).

biacm.db.seed.url - Oracle Business Analytics Warehouse connection URL (Format: *host:port/serviceName*).

3.4.4 Setup Step: Install Oracle BI Administration Tool

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

The Oracle BI Administration Tool enables you to manage the metadata repository and is required in the Oracle BI Applications set-up process. If the Oracle BI Administration Tool has not been installed, then you must install it on a Windows machine.

You install Oracle BI Administration Tool using the Oracle Business Intelligence Enterprise Edition Plus Client Installer. The Installer executable is named biee_client_install.exe and this file is available in the installed BI Middleware Home, in bi/clients/biserver.

Before you install the Oracle BI Administration Tool, ensure that the machine where you intend to perform the installation meets the following requirements:

- The machine is running a supported Microsoft Windows operating system.

- The machine has network connectivity with the machine that hosts your Oracle BI server.

For the instructions and considerations for installing the Oracle BI Administration Tool, see "Installing and Uninstalling Oracle Business Intelligence Client Tools" in *Oracle Fusion Middleware Installation Guide for Oracle Business Intelligence*.

3.4.5 Setup Step: Install and Configure Database Connectivity Software

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

You must install and configure the appropriate database connectivity software, Oracle Database Client, on the machines that host the Informatica PowerCenter Integration Service and Repository Service.

The PowerCenter Integration Services requires connectivity to Oracle Business Analytics Warehouse (that is, the target database). The Integration Service uses native connectivity to access Oracle Business Analytics Warehouse. This native connection is also used by Integration Services to connect to the Oracle BI Applications Configuration Manager tables, which reside in the $prefix\_DW$ schema.

The PowerCenter Repository Service requires native connectivity to communicate with the Informatica Repository database.

To install and configure the Oracle Database Client:

1. On the machine or machines that host the Integration Services and Repository Service, install the Oracle Database Client.

   Refer to the following topics in the *Informatica PowerCenter Configuration Guide*:

   - For Integration Service or Repository Service installed on Windows machines, see the section entitled "Configuring Native Connectivity" in Chapter 10, "Connecting to Databases from Windows."

   - For Integration Service or Repository Service installed on Linux or UNIX machines, see the section entitled "Configuring Native Connectivity" in Chapter 11, "Connecting to Databases from UNIX."

     Note: You must configure the environmental variables on Linux or UNIX as described in the Informatica documentation.

     ---

2. On the machine that hosts the Integration Service, create a tnsnames.ora entry for the $prefix\_DW$ schema. On the machine that will host the Repository Service, create a tnsnames.ora entry for the Informatica Repository.

   For additional information about database connectivity for Integration Services and Repository Services, refer to *Informatica PowerCenter Installation and Configuration Guide*.

3.4.6 Setup Step: Grant User Access to Oracle BI Applications Components

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

You must grant users access to Oracle BI Applications Configuration Manager, Functional Setup Manager, and DAC. Access to these applications is controlled through the following Duty Roles (Duty Roles are also referred to as Application Roles in Oracle APM):

- BI Applications Administrator Duty (BIA_ADMINISTRATOR_DUTY)

- BI Applications Implementation Manager Duty (BIA_IMPLEMENTATION_ MANAGER_DUTY)

- BI Applications Functional Developer Duty (BIA_FUNCTIONAL_DEVELOPER_ DUTY)

Examples of Setting Up Users For Oracle BI Applications

To set up security, you map Users to Enterprise Roles, and then you map Enterprise Roles to Duty Roles.

Work with your security administrator to grant Enterprise Roles (also known as Job Roles) to appropriate Duty Roles.

For example:

- In a small deployment, you might set up one user with an Enterprise Role named 'BIA_Admin_Role', which you map to the BI Applications Administrator Duty (that is, BIA_ADMINISTRATOR_DUTY). A user with the BIA_Admin_Role privileges can then access Oracle BI Applications components with required functionality.

- In a large deployment, you might set up the following users:

  - To manage the functional configuration, you set up one user with an Enterprise Role named 'BIA_Admin_Role', which you map to the BI Applications Administrator Duty (that is, BIA_ADMINISTRATOR_DUTY).

  - To set up an implementation project and assign tasks to functional developers, you set up two users with an Enterprise Role named 'BIA_Implementation_ Manager_Role', which you map to the BI Applications Implementation Manager Duty (that is, BIA_IMPLEMENTATION_MANAGER_DUTY).

  - To perform the Tasks for functional configuration in FSM, you set up ten users with an Enterprise Role named 'BIA_Functional_Developer_Role', which you map to the BI Applications Functional Developer Duty (that is, the BIA_ FUNCTIONAL_DEVELOPER_DUTY).

For more detailed information about security privileges required for Oracle BI Applications components, see Section 2.1.10 'Security Overview of Oracle BI Applications Configuration Manager and Functional Setup Manager' in *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*.

> **Note:** Many tasks in this chapter involve DAC and Oracle BI Applications Configuration Manager. The user who performs these tasks using these applications must possess the BI Applications Administrator Duty role.

3.4.7 Setup Step: Create a User for ETL

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

The Extract Transform and Load (ETL) process must be run by a user with appropriate data security privileges granted on the Fusion Application tables from which data is extracted into Oracle Business Analytics Warehouse. For this purpose, the Enterprise Role named FUSION_APPS_OBIA_BIEE_APPID is provisioned during install with the appropriate ETL security privileges.

Work with your Security Administrator to complete the steps below.

To create a user for ETL:

1. Use Oracle Identity Management (or an appropriate security tool) to create a new user and password.

   For example, you might create a new user named OBIA_ETL_USER.

2. Make the user a member of the Enterprise Role FUSION_APPS_OBIA_BIEE_ APPID.

3. Make a note of the user credentials.

   These credentials are used to configure connections in DAC and Informatica later in the setup process.

3.4.8 Setup Step: Run WCA to Configure Oracle BI Applications Components

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

If you are deploying on UNIX and are not deploying Oracle BI Applications in an enterprise deployment, then you may use WCA to install Informatica PowerCenter Services and to configure Oracle BI Applications components, as follows:

1. Run WCA as described in Section 3.4.8.1, "Setup Step: Configure Oracle BI Applications components using Warehouse Component Deployment Assistant (WCA) in a UNIX Environment."

2. Validate the setup as described in Section 3.4.8.2, "Setup Step: Validate the Oracle BI Applications Component Configuration."

3. Restart the BI components, as described in Section 3.4.8.3, "Restart BI Components."

For guidance on deciding whether to use WCA, refer to Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Notes

You must set the following environment variables before you attempt to start Informatica PowerCenter Services:

- Environment variables: TNS_ADMIN, INFA_HOME, INFA_DOMAINS_FILE.

- The PATH, NLS_LANG, LD_LIBRARY_PATH, and ODBCINI variables must be updated to include Informatica PowerCenter values.

- The ODBCINI variable points to the INFA odbc.ini file. The .ini file should include the DNS for the BI Server as described in Section 3.4.14, "Setup Step: Create an ODBC DSN to the Oracle BI Server."

3.4.8.1 Setup Step: Configure Oracle BI Applications components using Warehouse Component Deployment Assistant (WCA) in a UNIX Environment

Use WCA to install and configure a number of components for Oracle BI Applications in UNIX environments. If you are using WCA for the first time, Oracle recommends that before you start you read the sections Appendix A.6.1, "Overview of WCA" and Appendix A.6.2, "Notes on using WCA Configuration Parameters in wca_param.properties".

Note: WCA is not supported on Windows environments. For guidance on deciding whether to use WCA, refer to Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

To configure Oracle BI Applications components using WCA:

1. Locate the Informatica PowerCenter installation files.

   The Informatica PowerCenter installation files, license key and documentation are located in:

   ```
   <Fusion Applications Install Repository>\installers\PowerCenter
   ```

2. Copy the 910HF2_Server_Installer_<*OS.tar*> and License Key to the BI Domain machine, and extract the tar file into a folder, for example, /scratch/myfiles/infa9.

   Make a note of the directory path where you have extracted the Informatica files. When you configure the wca_params.properties file in Step 6, you specify the value of the INFA_INSTALL_UNZIP_LOC parameter as this path. For example, if you extracted the Informatica files into /scratch/myfiles/infa9, then you set the value of INFA_INSTALL_UNZIP_LOC to /scratch/myfiles/infa9.

3. Set the following environment variables for WCA:

   ```
   #Env Variable Setting in .cshrc file [Profile file of linux user] related to
   WCA. For example :
   setenv WCA_HOME "/scratch/aime/work/APPTOP/fusionapps/bi/dwtools/scripts/wca"
   setenv ANT_HOME /scratch/aime/work/APPTOP/fusionapps/modules/org.apache.ant_
   1.7.1
   setenv PATH "${ANT_HOME}/bin:${PATH}"
   setenv JAVA_HOME /scratch/aime/work/APPTOP/fusionapps/jdk6
   ```

4. Configure WebLogic Server so that it does not prompt for an Administrator password when starting up or shutting down, as follows:

   a. Create a file 'boot.properties' with the WebLogic Server Administrator's username password and store the file in $DOMAIN_HOME/servers/[server_name]/security.

   b. Add the following parameters to the boot.properties file:

   ```
   username=<Superuser with Admin role such as WebLogic>
   password=<password of user>
   ```

   c. Repeat steps 4.a and 4.b for Admin Server and all Managed Servers.

   For example, you might create the following files:

   MW_HOME/user_projects/domains/bifoundation_domain/servers/AdminServer/security/boot.properties.

   MW_HOME/user_projects/domains/bifoundation_domain/servers/bi_server1/security/boot.properties.

For more information about setting up a boot.properties file, see the topic 'Boot Identity Files' in the chapter 'Starting and Stopping Servers, in *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server*. Alternatively, locate this topic in WebLogic Server Administration Console Online Help.

5. Shut down BI domain WebLogic Admin server, managed server, and OPMN services.

Note: Node Manager should be running (or 'up').

6. Edit the wca_params.properties file, and specify a value for each parameter that you want to use.

Tip: For a full list of parameter descriptions, refer to the inline comments marked with # in the file wca_params.properties.

For example:

```
...
#This parameter represents the Informatica Installation home directory.
#Supported Values: Give the absolute path to Informatica Installation home
directory
INFA_HOME=/scratch/aime/work/Informatica/PowerCenter9.1

#This parameter represents host name of the machine where Informatica is
installed
INFA_PCS_HOST=localhost

#This parameter represents the Informatica Domain User Name that is created
when Informatica is installed.
#Supported Values: The domain user name provided when Informatica is installed.
#We can also use the default user provided by Informatica: Administrator
INFA_DOM_USER=Administrator

...
```

Note: For reference information about configuring the wca_params.properties file, see Appendix A.6.2, "Notes on using WCA Configuration Parameters in wca_param.properties".

7. Using a text editor, edit the file <BI_Oracle_Home>/dwtools/wca/wca_passwords.properties and specify a password for each component.

Note: If you have set the value of CONFIGURE_INFA to 'y', then you must specify appropriate passwords for the Informatica-related components.
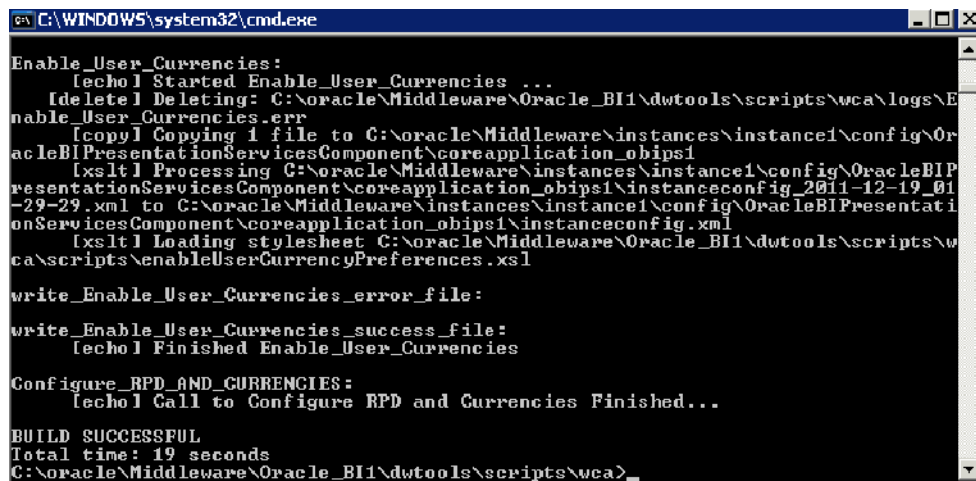
```
...
INFA_DOM_PASSWRD=admin
DW_DB_PASSWRD=biacm
DW_ETL_USER_PASSWRD=abc
INFA_DOM_DB_PASSWRD=password
INFA_REP_DB_PASSWRD=password
INFA_REP_PASSWRD=Administrator
DAC_DB_PASSWRD=biacm
MDS_DB_PASSWRD=fusion_mds
WL_PASSWRD=fusionfa1
RPD_ENCRYPTION_PASSWRD=welcome1
...
```

> **Note:** To maintain security, Oracle recommends that after you have
> completed configuration with WCA, you remove the passwords from
> the passwords.properties file.

8. Start WCA with the values specified in the properties files, by executing the
   following command:

   ```
   <MW_HOME>\Oracle_BI1\dwtools\scripts\wca\ant
   ```

   Use the progress report displayed in the console and associated log files to
   determine whether all components were configured correctly. If all components
   were configured correctly, then the message 'BUILD SUCCESSFUL' is displayed in
   the console and is recorded in the log file.

   If WCA fails to configure an Oracle BI Applications component, then the WCA
   script is stopped, an error message is displayed in the console, and an ERR file is
   created.

   If WCA fails to configure an Oracle BI Applications component, then when the
   WCA script stops, do the following:

   a. Attempt to correct the fault that is reported.

      Use the log files created by WCA in the wca\logs folder to diagnose the error
      and attempt to correct the fault that is reported. Log files are named with a
      timestamp.

   b. Re-start WCA.

> **Note:** If WCA cannot configure a component, you must manually configure that component, as described in "Diagnosing WCA Errors" below, then re-start WCA. If you manually configure a component and want to re-start WCA, then you must change the ERR extension to a SUCC extension for the appropriate error file. For example:
>
> 1. WCA reports an error creating the Informatica repository, and creates the file create_Repository_service.err.
>
> 2. If you cannot resolve this error to enable WCA to create the Informatica repository, then you manually create the Informatica repository as described in Section 3.4.9.2, "Setup Step: Creating the Informatica Repository Service."
>
> 3. You rename the file create_Repository_service.err to create_Repository_service.succ, then re-start WCA.
>
>   If you do not rename this file, then the WCA script will stop when it detects the create_Repository_service.err file.
>
> **Note**: Do not delete other ERR files or SUCC files created by WCA.

9. (Recommended) Remove the passwords from the passwords.properties file.

> **Note:** To maintain security, Oracle recommends that after you have completed configuration with WCA, you remove the passwords from the passwords.properties file.

Diagnosing WCA Errors

Table 3–3, " Diagnosing WCA Errors" explains how to manually rectify WCA configuration errors. For example, if WCA fails to create an Informatica PowerCenter Repository Service, then you must analyze and attempt to fix the problem reported in the WCA log. If you cannot fix the problem, then you must create a Repository Service manually using Informatica Administrator, as specified in the **Action Required If You Cannot Resolve The Error** column.

Log files are available under `MW_HOME/ORACLE_BI_HOME/dwtools/scripts/wca/logs`. The file wca_<timestamp>.log should be used to for debugging failures that happen during WCA script execution.

If an error cannot be fixed, use the manual task in the **Action Required If You Cannot Resolve The Error** column to configure the component.

Table 3–3   Diagnosing WCA Errors

WCA Configuration Error	Action Required If You Cannot Resolve The Error
Failure to create Informatica PowerCenter Repository Service	Manually create the Informatica PowerCenter Repository Service, by following the steps in Section 3.4.9.2, "Setup Step: Creating the Informatica Repository Service." Error files: create_Repository_service.err

Table 3–3 (Cont.) Diagnosing WCA Errors

WCA Configuration Error	Action Required If You Cannot Resolve The Error
Failure to create Informatica PowerCenter Integration Service	Manually create the Informatica PowerCenter Integration Service, by following the steps in Section 3.4.9.3, "Setup Step: Creating the Informatica Integration Service." Error files: create_integration_service.err create_integration_service.succ
Failure to restore the Informatica PowerCenter Repository	Manually restore the Informatica PowerCenter Repository, by following the steps in Section 3.4.9.4, "Setup Step: Load the Prebuilt Informatica Repository." Error files: restoreInfaRep_restoreInfaRep.err restoreInfaRep_changeOpModeToNormal.err
Failure to copy source files to the Informatica PowerCenter machine	Manually copy source files to the Informatica PowerCenter machine, by following the steps in Section 3.4.9.5, "Setup Step: Copying Source Files to the Informatica PowerCenter Services machine." Error files: copySrcFile_copyZipFile.err copySrcFile_unzipCopiedZipFile.err
Failure to configure Informatica PowerCenter Properties	Manually configure Informatica PowerCenter properties, by following the steps in: Section 3.4.9.6, "Setup Step: Setting PowerCenter Integration Services Relaxed Code Page Validation." Section 3.4.9.7, "Setup Step: Setting PowerCenter Integration Services Custom Properties."
Failure to create a repository Administrator user	Manually create a repository Administrator user, by following the steps in Section 3.4.12.3, "Setup Step: Creating the Repository Administrator User in the Native Security Domain." Error files: creuser_creRepAdminUser.err creuser_assignRepAdminRole.err
Failure to create Informatica Relational Connections	Manually create Informatica Relational Connections, by following the steps in Section 3.4.15.3, "Setup Step: Configuring Relational Connections in Informatica." Error files: creconn_connToRepInAuthenticatedMode.err creconn_creDWConn.err creconn_creBIACMConn.err creconn_creFV1CRMConn.err creconn_creFV1HCMConn.err creconn_creFV1FSCMConn.err

Table 3–3   (Cont.) Diagnosing WCA Errors

WCA Configuration Error	Action Required If You Cannot Resolve The Error
Failure to create DAC Integration Settings	Manually create DAC integration settings, by following the steps in Section 3.4.15.2, "Setup Step: Configuring Physical Data Source Connections in DAC." and Section 3.4.15.1, "Setup Step: Register Informatica in DAC and Configure DAC Integration Settings." Error files: configure_DAC.err
Failure to register Source Systems	Manually register Source Systems, by following the steps in Section 3.4.15.6, "Setup Step: Registering Source Systems." Error files: registerSrcInCm.err
Failure to extend the BI Domain	Manually extend the BI Domain, by following the steps 1 and 2 in Section 3.4.9.8, "Setup Step: Extending the BI Domain." Error files: Extending_BI_Domain.err If WCA fails to extend the BI Domain and generates the error message '[ERROR]: The Weblogic Admin Server Failed to Start', then do the following: 1. Start WebLogic Admin Server. 2. Re-start WCA.
Failure to configure the BI Domain	Manually configure the BI Domain, by following the steps 3 and 4 in Section 3.4.9.8, "Setup Step: Extending the BI Domain." Error files: Configuring_BI_Domain.err
Failure to configure RPD connection pools	Manually configure RPD connection pools, by following the steps in Section 3.4.9.9, "Setup Step: Configuring the Oracle BI Connection Pools." Error files: Configure_Connection_Pools.err
Failure to enable currencies	Manually enable currencies, by following the steps in Section 3.4.15.5, "Setup Step: Enabling User Currency Preference Settings." Error files: Enable_User_Currencies.err

3.4.8.2 Setup Step: Validate the Oracle BI Applications Component Configuration

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

To validate the Oracle BI Applications component configuration, do the following:

1. In a shell, run bi-init.sh or bi-init.cmd.

2. In the same shell, set all environment variables that are suggested for running Domains ETL.

3. Navigate to the folder <WCA_HOME>/validations/.

4. In the same shell, execute the ValidateWCA.py script, for example:
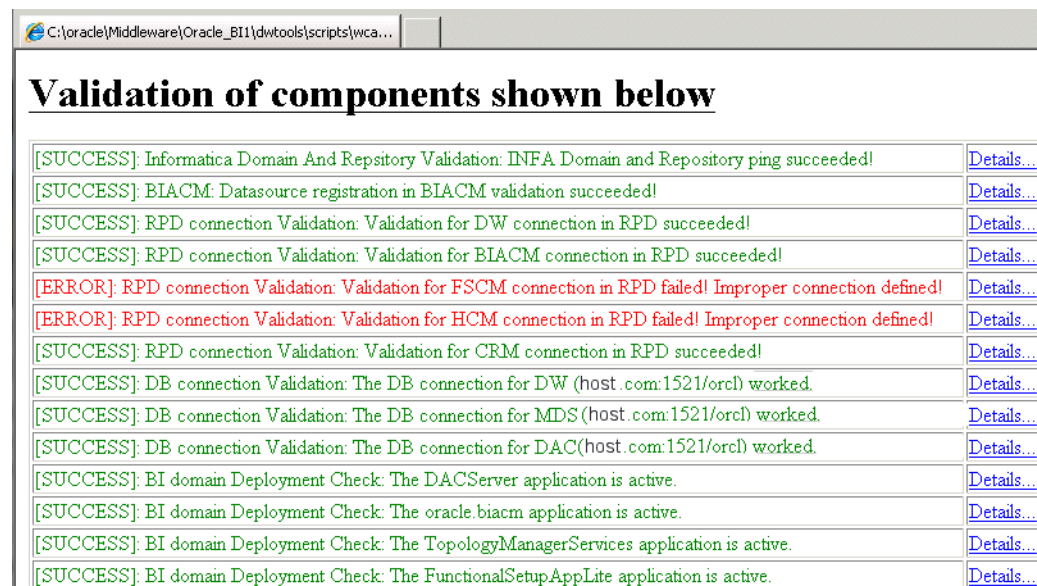
```
$BI_ORACLE_HOME/common/bin/wlst.sh <WCA_HOME>/validations/ValidateWCA.py
```

Notes:

Before you execute the ValidateWCA.py script, you must do the following:

- Make sure that the DAC server is in the 'Active' state in Managed Server.

- Make sure that DAC Client can connect to the DAC Server.

- Assign the execute permission to the wca/validations/validateDAC.sh script.

The ValidateWCA.py script produces a Web report containing an analysis of the configuration of each component, with a 'Details' link to more detail for each component. For example:

Note: If you are deploying Oracle BI Applications in a Windows environment (that is, not supported by WCA), then to use the ValidateWCA.py script you must manually edit the wca_parameter.properties file to specify values for all components that you have manually configured. For example, if you manually created the Informatica PowerCenter Repository Service and Integration Service, then you must manually set values for the INFA_RS_NAME and INFA_IS_NAME parameters that match the service names that you used.

3.4.8.3 Restart BI Components

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

After running WCA, you must start or restart the following components:

- Node Manager

- Admin Server

- Managed Server

- BI Services (that is, OPMN)

- DAC Server (for more information about restarting DAC Server, see Section A.7, "How to Restart DAC Server"

3.4.9 Setup Step: Manually Set Up Oracle BI Applications Components

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

If you are configuring some Oracle BI Applications components manually (that is, you are not using WCA), then you must perform the following steps:

1. Install and configure Informatica PowerCenter Services, as described in Section 3.4.9.1, "Install and Set Up Informatica PowerCenter Services Manually."

2. Create the Repository Service, as described in Section 3.4.9.2, "Setup Step: Creating the Informatica Repository Service."

3. Create the Integration Service, as described in Section 3.4.9.3, "Setup Step: Creating the Informatica Integration Service."

4. Load the pre-built Informatica repository, as described in Section 3.4.9.4, "Setup Step: Load the Prebuilt Informatica Repository."

5. Copy source files to the Informatica PowerCenter Services machine, as described in Section 3.4.9.5, "Setup Step: Copying Source Files to the Informatica PowerCenter Services machine."

6. Set the PowerCenter Relaxed Code Page Validation option, as described in Section 3.4.9.6, "Setup Step: Setting PowerCenter Integration Services Relaxed Code Page Validation."

7. Set the PowerCenter Custom Properties, as described in Section 3.4.9.7, "Setup Step: Setting PowerCenter Integration Services Custom Properties."

8. Extend the BI Domain, as described in Section 3.4.9.8, "Setup Step: Extending the BI Domain."

9. Configure the Oracle BI Connection Pools, as described in Section 3.4.9.9, "Setup Step: Configuring the Oracle BI Connection Pools."

3.4.9.1 Install and Set Up Informatica PowerCenter Services Manually

In a UNIX environment, WCA automatically installs and sets up Informatica PowerCenter Services V9.1. In a Windows environment or in an enterprise deployment, you must follow all of the steps in this section to manually install and set up Informatica PowerCenter Services.

To manually install and set up Informatica PowerCenter for an Oracle BI Applications environment, complete the following tasks, in order:

1. Read the installation considerations in Section 3.4.9.1.2, "Before Installing Informatica PowerCenter for Oracle BI Applications Environments" and the prerequisites in Section 3.4.9.1.1, "Prerequisites for Installing Informatica PowerCenter for Oracle BI Applications Environments."

2. Install Informatica PowerCenter Services, as described in Section 3.4.9.1.4, "Installing Informatica PowerCenter Services and Hotfix 2."

   Note: If you have licensed Informatica PowerCenter separately from Oracle BI Applications, skip this step and continue with step 3.

3. You must set the following environment variables before you attempt to start Informatica PowerCenter Services:

   – Environment variables: TNS_ADMIN, INFA_HOME, INFA_DOMAINS_FILE.

   – The PATH, NLS_LANG, LD_LIBRARY_PATH, and ODBCINI variables must be updated to include Informatica PowerCenter values.

- The ODBCINI variable points to the INFA odbc.ini file. The .ini file should include the DNS for the BI Server as described in Section 3.4.14, "Setup Step: Create an ODBC DSN to the Oracle BI Server."

Set environment variables, as follows:

a. Run the bi-init-sh or bi-init-cmd script, which sets the ORACLE_HOME environment variable.

b. Set the following environment variables:

INFA_HOME

INFA_DOMAINS_FILE

JAVA_HOME

JRE_HOME

LD_LIBRARY_PATH

NLS_LANG

ODBCINI

ORACLE_HOME

PATH

TNS_ADMIN

For example (with example values):

```
TNS_ADMIN=/scratch/aime/app/aime/product/11.2.0/dbhome_1/network/admin
INFA_HOME=/infa
INFA_DOMAINS_FILE=/infa/domains.infa
LD_LIBRARY_
PATH=/dev/shm/infa/ODBC5.2/lib:/dev/shm/infa/server/bin:/dev/shm/infa/java/
lib:/dev/shm/infa/java/jre:/scratch/aime/work/APPTOP/fusionapps/bi/bifounda
tion/server/bin:/scratch/aime/work/APPTOP/fusionapps/bi/bifoundation/odbc/l
ib:/scratch/aime/work/APPTOP/dbclient/lib:$LD_LIBRARY_PATH
PATH=/dev/shm/infa/server/bin:/dev/shm/infa/ODBC5.2/bin:/dev/shm/infa/serve
r/tomcat/bin:/dev/shm/infa/java/jre:/scratch/aime/work/APPTOP/fusionapps/bi
/bifoundation/server/bin:/scratch/aime/work/APPTOP/fusionapps/bi/bifoundati
on/server/bin:/scratch/aime/work/APPTOP/dbclient/bin
ODBCINI=/infa/ODBC5.2/odbc.ini
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
```

Note: The odbc.ini file specified by the ODBCINI= parameter must include an entry Driver=<*Driver location and name*>. For example, the odbc.ini file might contain the following parameters and values:

```
...
Driver=/scratch/aime/work/APPTOP/fusionapps/bi/bifoundation/server/bin/libn
qsodbc64.so
Description = Oracle BI Server
...
```

3.4.9.1.1   Prerequisites for Installing Informatica PowerCenter for Oracle BI Applications Environments   Oracle BI Applications Version 11.1.1.6.2 requires Informatica PowerCenter 9.1 with Hotfix 2. For information on where to obtain the installation files, see Section 3.4.9.1.3, "Locations of Installation Files for Informatica PowerCenter and Hotfix 2."

> **Note:** For information about additional hotfixes or emergency bug fixes that are required to support the current version of Informatica, see the Certification Document for Oracle BI Applications:
>
> http://www.oracle.com/technology/software/products/i as/files/fusion_certification.html

If you license Informatica PowerCenter separately and you do not have Informatica PowerCenter 9.1 and Hotfix 2, then before deploying Oracle BI Applications you must upgrade your Informatica PowerCenter license to 9.1 and Hotfix 2.

For more information about supported operating systems for Informatica PowerCenter, see the Certification Document for Oracle BI Applications:

http://www.oracle.com/technology/software/products/ias/files/fus ion_certification.html

3.4.9.1.2 Before Installing Informatica PowerCenter for Oracle BI Applications Environments

Before you install and configure Informatica PowerCenter for an Oracle BI Applications environment, consider the following:

- Determine which machines will host Informatica PowerCenter Client Tools and Informatica PowerCenter Services. Informatica PowerCenter Services can be installed on UNIX or Windows. Informatica PowerCenter Client Tools must be installed on Windows.

- DAC uses the Informatica `pmrep` and `pmcmd` command line programs to communicate with the Informatica Integration Service and the Repository Service. For these reasons, you must do *one* of the following:

    - Co-locate the DAC Server with PowerCenter Services, and DAC Client with the Informatica Clients.

    - Make the Informatica binaries and infa.domains file available to DAC by placing them on a shared network location, by either installing Informatica on a common NFS mount to which DAC has access, or by installing a dummy Informatica service on the same machine as the DAC server and copying the domains.infa file from a functional Informatica installation on another machine.

- DAC produces parameters files that are consumed by Informatica. By default, the parameter files are produced in the following location:

    DOMAIN_HOME\dac\Informatica\parameters

    You must either share this location so that Informatica PowerCenter Services has read and write access, or configure a shared location to which both the DAC Server and Informatica PowerCenter Services have read and write access.

- The Informatica PowerCenter Integration Service connects to the Oracle BI Server (for access to the Oracle Fusion Applications source system) using the BI Server ODBC. If the Informatica PowerCenter Services Integration Service is not installed on the same machine as the BI Server on Linux and UNIX machines, then the Integration Service user account must have access to the shared Middleware Home for BI. The Integration Service must be able to access the ODBC library files for the BI Server.

- You must meet all requirements and complete preinstallation tasks as described in Section 3.4.9.1.1, "Prerequisites for Installing Informatica PowerCenter for Oracle BI Applications Environments" and in the chapter "Before You Install" of

Informatica PowerCenter Installation and Configuration Guide. For information on the location of the Informatica documentation, see Section 3.4.9.1.3, "Locations of Installation Files for Informatica PowerCenter and Hotfix 2."

■ When you install PowerCenter Services and Client Tools version 9.1 and Hotfix 2, use the Informatica PowerCenter installation documentation in conjunction with the steps provided in the following sections. This chapter summarizes the steps to install Informatica PowerCenter and assumes a single-machine installation for the PowerCenter Services components. For information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica, see Certification Document for Oracle BI Applications:

http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

■ The database user account that you use to access the Informatica PowerCenter repository must have DBA privileges.

3.4.9.1.3 Locations of Installation Files for Informatica PowerCenter and Hotfix 2 The Fusion Applications software repository is obtained when you download and unzip the Oracle Fusion Applications 11g parts available in the Oracle Fusion Applications Media Pack on the Oracle Software Delivery Cloud.

The Informatica PowerCenter install files, license key and documentation are located in:

```
<Fusion Applications Install Repository>\installers\powercenter
```

The powercenter folder contains the following files:

■ OS-specific Server install file - 910HF2_Server_Installer_<OS.zip or tar>

■ 910HF2_Client_Installer_win32-x86.zip (Windows only)

■ Informatica Documentation (Infa91Docs.zip)

■ Informatica License Key

Note: The Informatica PowerCenter installer for Services and Client installs both version 9.1 and HotFix 2.

3.4.9.1.4 Installing Informatica PowerCenter Services and Hotfix 2 This section explains how to install Informatica PowerCenter Services 9.1 and Hotfix 2 for an Oracle BI Applications deployment. For detailed generic information about installing Informatica 9.1, see *Informatica PowerCenter Installation and Configuration Guide*. For the locations of the installation files for Informatica PowerCenter and Hotfix 2, see Section 3.4.9.1.3, "Locations of Installation Files for Informatica PowerCenter and Hotfix 2." For information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica, see the Certification Document for Oracle BI Applications:

http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

> **Note:**
>
> - If you have purchased the 'Informatica OEM PowerCenter ETL Server and PowerConnect Adapters' license from Oracle, use the Oracle_All_OS_Prod.key license key file. This file is located in the \powercenter\ folder as described in Section 3.4.9.1.3, "Locations of Installation Files for Informatica PowerCenter and Hotfix 2."
>
> - On Windows, do not install Informatica in a directory that has a space in the directory name. For example, do not install in D:\Program Files\INFA. If you install into a directory with a space in the name, you will cause errors in DAC.

To install Informatica PowerCenter Services 9.1 and Hotfix 2:

1. Start the Informatica PowerCenter Services installer, depending on your operating system:

   - To start the installer on Windows, run the file install.bat from the root directory where you unzipped the PowerCenter Services 9.1 installation package.

   - To start the installer on UNIX or Linux, use a shell command line to run install.sh from the root directory where you untarred the PowerCenter Services 9.1 installation package.

     On UNIX or Linux, select the option to use the GUI mode installation.

2. Follow the instructions on the Informatica PowerCenter Services installation wizard.

   For more information about installing PowerCenter Services, refer to *Informatica PowerCenter Installation and Configuration Guide*.

3. Stop any Informatica services that are running.

For more information, see the Informatica PowerCenter Version 9.1 HotFix Release Notes.

3.4.9.2 Setup Step: Creating the Informatica Repository Service

Follow this procedure to create the Informatica Repository Service.

Note: Before you log into Informatica Administrator, make sure that the Informatica service is running. To start the Informatica Service, use the Start Informatica Services program. On Windows, you can also use the Windows Services dialog to start the 'Informatica 9.1.0' service. On UNIX, use the infaservice [startup | shutdown] command.

For detailed information about creating a Repository Service, refer to the Informatica documentation.

To create the Informatica Repository Service:

1. Open Informatica Administrator and display the Domain tab.

   For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Domain Navigator, select the root Domain, then click Actions, then New, then PowerCenter Repository Service, to display the New PowerCenter Repository Service wizard.

3. Use the 'Specify the properties for this new PowerCenter Repository Service' dialog to specify a Name (for example, BIA_RS), Location, License, and Node, then click Next.

4. Use the 'Specify the database properties for this new PowerCenter Repository Service' dialog to specify a Database Type, Username, Password, Connection String, and Code Page.

   Notes:

   - The Username and Password are for the database account that you created for the Informatica repository

   - If you specify the wrong code page, you cannot correct this later. If the code page is incorrect, then you must create a new Repository Service.

5. In the 'Specify the creation options for the new PowerCenter Repository Service' area, select the 'No content exists under specified connection string. Create new content' radio button. For detailed information about creating a Repository Service, refer to the Informatica documentation.

6. Click Finish.

3.4.9.3 Setup Step: Creating the Informatica Integration Service

Follow this procedure to create the Informatica Integration Service.

Note: Before you begin, make sure that you have set up the environment variables for Informatica PowerCenter, as described in Section 3.4.9.1.4, "Installing Informatica PowerCenter Services and Hotfix 2".

For detailed information about creating an Integration Service, refer to the Informatica documentation.

To create the Informatica Integration Service

1. Open Informatica Administrator and display the Domain tab.

   For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Domain Navigator, select the root Domain, then click Actions, the New, then PowerCenter Integration Service, to display the New PowerCenter Integration Service wizard.

3. Use the 'Specify the properties for this new PowerCenter Integration Service' dialog to specify a Name (for example, BIA_IS), Location, License, and Node, then click Next.

4. Use the 'Specify the PowerCenter Integration Service and its login credentials' dialog to select the Repository Service that you created in a previous step, and specify the repository username and password (for example, Administrator\Administrator).

5. In the 'Select the data movement mode' area, select the appropriate Code Page setting from the Data Movement Mode drop down list.

6. Click Finish.

   The Integration Service will only start if the Repository Service is operating with the Operating Mode set to 'Normal'.

3.4.9.4 Setup Step: Load the Prebuilt Informatica Repository

During the Oracle BI Applications installation, an Informatica Repository file named Oracle_BI_DW_Base.rep is installed into the *ORACLE_HOME* directory.

To load the prebuilt Informatica Repository:

- In an English environment, follow the steps in Section 3.4.9.4.1, "Restoring the Prebuilt Informatica Repository for Environments in English."

- In an non-English environment, follow the steps in Section 3.4.9.4.2, "Restoring the Prebuilt Informatica Repository on a Non-English Operating System."

3.4.9.4.1   Restoring the Prebuilt Informatica Repository for Environments in English You use the Restore option in Informatica Administrator to load the prebuilt Oracle_BI_DW_Base repository.

To load the prebuilt Oracle_BI_DW_Base.rep repository into Informatica:

1. Copy the file ORACLE_HOME/biapps/admin/provisioning/dac/infabaseline/Oracle_BI_DW_Base.rep file on the installation machine to the following directory on the machine where Informatica PowerCenter Services is installed:

   - On Windows, copy the file to the *INFA_HOME*\server\infa_shared\Backup directory.

   - On UNIX or Linux, copy the file to *INFA_HOME*/server/infa_shared/Backup directory.

   Note: for more information about Oracle home locations, see Section 3.3.3, "Summary of Home Directories."

2. In Informatica Administrator, select the Repository Service.

   For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

3. In the General Properties area of the Properties tab, make sure the OperatingMode value is Exclusive.

   To change the OperatingMode value, click Edit, and then select a new value from the drop-down list. Click OK to exit edit mode.

4. In the Actions menu at the top right-hand side, choose Actions, then Repository Contents, then Delete.

5. At the Delete Contents for <repository name> dialog box, enter the repository username and password (for example, Administrator\Administrator), then click OK.

6. In the Actions menu at the top right-hand side, choose Actions, then Repository Contents, then Restore.

7. At the Restore Contents for <repository name> dialog, enter the repository username and password (for example, Administrator\Administrator), then click OK.

8. Choose Actions, then Repository Contents, then Restore.

9. At the Restore Contents dialog, select Oracle_BI_DW_Base.rep from the Select Backup File drop-down list.

10. Select the Restore as New check box.

11. Click OK to start the restore process.

12. When the restore process is complete, click Close (or click Save first to save the logging information).

When the restore is complete (the process typically takes approximately ten to twenty minutes depending on repository size and network speed), you will see a 'Success' message.

13. When a repository is restored, the repository becomes a standalone repository. After restoring the repository, you need to promote it to a global repository.

14. Change the OperatingMode value to Normal.

To change the Operating Mode, display the Properties tab, in the General Properties area, click Edit, then click the OperatingMode drop-down list, and select Normal.

Note: If you do not set the Operating Mode to Normal, then the Integration Service will not restart.

15. If prompted, enter the repository username and password.

3.4.9.4.2 Restoring the Prebuilt Informatica Repository on a Non-English Operating System If Informatica PowerCenter Services is installed on a non-English version of the operating system, you must use the command line to restore the prebuilt Informatica Repository provided with Oracle BI Applications.

To restore the prebuilt Informatica Repository on a non-English operating system:

1. Copy the ORACLE_ HOME/biapps/admin/provisioning/dac/infabaseline/Oracle_BI_DW_Base.rep file on the installation machine to the following directory on the machine where Informatica PowerCenter Services is installed:

- On Windows copy the file to the *INFA_HOME*\server\infa_ shared\Backup directory.

- On UNIX or Linux copy the file to *INFA_HOME*/server/infa_ shared/Backup directory.

Note: For more information about Oracle home locations, see Section 3.3.3, "Summary of Home Directories"

---

2. Open a command window.

3. Enter the following command to connect to the repository:

```
Pmrep connect -r Repository_Name -d Domain
```

4. Enter the following command to restore the repository:

```
PmRep restore -u Domain_User_Name -p Domain_User_Password -i Input_File_Name -n
```

where the *Input_File_Name* is the name of the prebuilt repository file.

3.4.9.5 Setup Step: Copying Source Files to the Informatica PowerCenter Services machine

You need to copy source files from the Oracle BI Applications installation directory to the Informatica directory on the Informatica PowerCenter Services machine.

Note: The following instructions assume the default Informatica directory locations for source files.

■ The default Informatica directory for source files is *INFA_HOME*\server\infa_shared\SrcFiles. You can confirm the source file directory for your environment by launching the PowerCenter Administration Console and going to the General Properties area of the Processes tab. The parameter name for the source file directory is $PMSourceFileDir.

Note: The value of the $PMSourceFileDir parameter must match the DAC system property called InformaticaParameterFileLocation. You will set this property later in the set up process, using the instructions in .

Before you start

After installation, the Source Files and other ETL-related files required for Informatica PowerCenter are located in the following ZIP file:

ORACLE_HOME\biapps\etl\biapps_etl.zip

Extract the files in biapps_etl.zip into the following location:

ORACLE_HOME\biapps\etl\biapps_etl\

You extract the following directory:

ORACLE_HOME\biapps\etl\biapps_etl\data_files\src_files

Copying the source files

To copy source files to Informatica PowerCenter Services:

1. Copy the source files as follows:

   From:

   ORACLE_HOME\biapps\etl\biapps_etl\data_files\src_files

   To:

   INFA_HOME\server\infa_shared\SrcFiles

   Note: If Informatica PowerCenter Services is installed on a UNIX or Linux machine, convert the target paths above from Windows to UNIX/Linux. For example, covert the Windows target path:

   From:

   INFA_HOME\server\infa_shared\SrcFiles

   To:

   INFA_HOME/server/infa_shared/SrcFiles

3.4.9.6 Setup Step: Setting PowerCenter Integration Services Relaxed Code Page Validation

PowerCenter Integration Services must be configured for relaxed code page validation. For more information, refer to section Understanding Globalization Relaxed Code Page Validation in *Informatica Administrator Guide.*

To configure Informatica PowerCenter Integration Services for relaxed code page validation:

1. Open Informatica Administrator and display the Domain tab.

For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Domain Navigator pane, select the **Integration Service**.

3. Display the **Properties** page.

4. In the Configuration Properties area, click **Edit to display the Edit Configuration Properties page**.

5. Clear the **ValidateDataCodePages** check box.

3.4.9.7 Setup Step: Setting PowerCenter Integration Services Custom Properties

Follow this procedure to set PowerCenter Integration Services custom properties.

To set Informatica PowerCenter Integration Services Custom Properties:

1. Open Informatica Administrator and display the Domain tab.

   For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Domain Navigator pane, select the **Integration Service**.

3. Display the **Properties** page.

4. In the Custom Properties area, click **Edit to display the Edit Custom Properties page**.

5. Use the New option to display the New Custom Property dialog, and add the following Custom Properties:

Table 3–4   Custom Properties for Informatica PowerCenter Integration Services

Custom Properties Name	Custom Properties Value	Notes
overrideMpltVarWithMapVar	Yes	Enables Informatica to evaluate parameters within mapplets.

6. Click **OK** to save the details.

7. Ensure that the Integration Service and Repository Service that you created during the Informatica PowerCenter installation are running.

3.4.9.8 Setup Step: Extending the BI Domain

During the Fusion Applications installation and provisioning process, the BI Domain is created. However, none of the Oracle BI Applications components (Oracle BI Applications Configuration Manager, DAC Server, and Functional Setup Manager) are deployed in the WebLogic BI domain. Therefore, you must extend the WebLogic BI domain to include these components. You deploy these components in WebLogic using the following python configuration scripts:

- install_dwtools.py

- configure_dwtools.py

The install_dwtools.py and configure_dwtools.py scripts, along with the other Oracle BI Applications scripts, can be found in *ORACLE_HOME*/dwtools/scripts. However, the version of the WebLogic Scripting Tool (`wlst`) that must be used can be found in *ORACLE_HOME*/common/bin.

The install_dwtools.py and configure_dwtools.py scripts also deploy ATG, which is a J2EE component used by Oracle BI Applications Configuration Manager and Functional Setup Manager.

Oracle BI Applications Configuration Manager, Functional Setup Manager, and ATG are deployed to the WebLogic Administration Server. The DAC Server is deployed to the WebLogic Managed Server. To deploy DAC into a Managed Server in a different domain home from the Administration Server, or to deploy the DAC Server for high availability, see "Configuring Oracle Business Intelligence Applications" in *Oracle Fusion Applications Customer Relationship Management Enterprise Deployment Guide*.

> **Note:** Only one DAC Server application can be run against the DAC repository at one time. Therefore, the DAC Server application should be deployed to only one Managed Server. If more than one DAC Server application is run against the DAC repository, the repository will be corrupted.
>
> For information about accessing the DAC Server through the WebLogic Server Administration Console, see Section 10.9, "Monitoring the DAC Server Using WebLogic Server" in the *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Data Warehouse Administration Console*.

To extend the BI domain:

1. Stop the Administration Server.

2. Run install_dwtools.py using the following syntax:

   WLST_PATH install_dwtools.py '*DOMAIN_HOME*' '*INFA_HOME*' '*INFA_ DOMAIN_FILE*' '*DW_DB_CONNECT*' '*DW_DB_SCHEMA*' '*MDS_DB_CONNECT*' '*MDS_ DB_SCHEMA*' '*DAC_DB_CONNECT*' '*DAC_DB_SCHEMA*' '*DAC_TARGET*'

   where:

   - *WLST_PATH* is the path to WebLogic Scripting Tool.

   - *DOMAIN_HOME* is the location of the Oracle BI domain (for example, /instance/domains/<*hostname*>/BIDomain).

   - *INFA_HOME* is the location of the user-specified Informatica installation directory on the machine that hosts Informatica PowerCenter Services (for example,\informatica\9.1\).

   - *INFA_DOMAIN_FILE* is the location and name of the Informatica domains fine infa.domain on the machine that hosts Informatica PowerCenter Services (for example,\informatica\8.6.1\domains.infa).

   - *DW_DB_CONNECT* is the connection string for the database that stores the DW schema that you created when you performed the setup steps in Section 3.4.2, "Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications". For example, machine12345.example.com:1521/BIA11G_DW.

   - *DW_DB_SCHEMA* is the name of the DW schema that you created when you performed the setup steps in Section 3.4.2, "Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications". For example, you might have specified the name DEV11G_DW.

- *MDS_DB_URL* is the connection string for the Oracle BI EE platform MDS schema (that is, not the Fusion Applications schema). This schema was created during the provisioning of Oracle Business Intelligence, as part of the Oracle Fusion Applications installation process. For example, `machine12345.example.com:1521/BIA11G_MDS`.

- *MDS_DB_SCHEMA* is the name of the Oracle BI EE platform MDS schema (that is, not the Fusion Applications MDS schema). This schema was created during the provisioning of Oracle Business Intelligence, as part of the Oracle Fusion Applications installation process. For example, `BIA11G_MDS`.

- *DAC_DB_URL* is the connection string for the database that stores the DAC schema that you created when you performed the setup steps in Section 3.4.2, "Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications". For example, `machine12345.example.com:1521/BIA11G_DAC`.

- *DAC_SCHEMA* is the name of the DAC schema that you created when you performed the setup steps in Section 3.4.2, "Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications". For example, you might have specified the name `BIA11G_DAC`.

- *DAC_TARGET* is the name of the Manager Server into which the DAC Server is deployed.

For example:

```
/mw_home/Oracle_BI1/common/bin/wlst.sh install_dwtools.py '/mw_home/user_
projects/domains/bifoundation_domain' '/informatica' '~/informatica.infa'
'jdbc:oracle:thin:@localhost:1521/orcl' 'BIA11G_DW'
'jdbc:oracle:thin:@localhost:1521/orcl' 'BIA11G_MDS'
'jdbc:oracle:thin:@localhost:1521/orcl' 'BIA11G_DAC' 'bi_server1'
```

> **Note:** When you run the script, you are also prompted for passwords.

3. Start the Administration Server.

4. Run configure_dwtools.py using the following syntax:

WLST_PATH configure_dwtools.py '*WL_USERNAME*' '*ADMIN_HOST*' '*ADMIN_PORT*'

where:

- *WLST_PATH* is the path to the WebLogic Scripting Tool.

- *WL_USERNAME* is the WebLogic username

- *ADMIN_HOST* is the machine that hosts the Administration Server

- *ADMIN_PORT* is the port number for the Administration Server

For example:

```
/mw_home/Oracle_BI1/common/bin/wlst.sh configure_dwtools.py 'weblogic'
'localhost' '7001'
```

> **Note:** When you run the script, you are also prompted for passwords.

3.4.9.9 Setup Step: Configuring the Oracle BI Connection Pools

Use the Oracle BI Applications configuration script (configure_rpd.py) to configure the following connection pools for Oracle BI Applications in the BI metadata repository file:

- Oracle Business Analytics Warehouse data source connection pools:
    - Oracle Data Warehouse Connection Pool
    - Oracle Data Warehouse Repository InitBlock Connection Pool
- Connection pool for Oracle BI Applications Configuration Manager
- Oracle DB InitBlock Initialization Connection Pool (uses the same values as those used for the HCM_OLTP connection)

In addition, the OLAPTBO repository static variable needs to be configured manually.

Note that the following repository files are provided on installation:

- OracleBIApps.rpd - the full repository file containing all Oracle BI Applications and Oracle Fusion Transactional Business Intelligence projects
- OracleBIAppsCRM.rpd - a partial repository file containing Oracle BI Applications and Oracle Fusion Transactional Business Intelligence projects for CRM modules
- OracleBIAppsERP.rpd - a partial repository file containing Oracle BI Applications and Oracle Fusion Transactional Business Intelligence projects for ERP modules

The OracleBIApps.rpd file is configured during Fusion Applications installation and provisioning, and is deployed to the BI Instance (for example, BI_INSTANCE\bifoundation\OracleBIServerComponent\coreapplication_obis1\repository). The configure_rpd.py script used in this procedure configures the repository file deployed in the BI Instance. The configurations specific to Oracle BI Applications made by the configure_rpd.py script must be performed on the repository file that was configured during Fusion Applications installation and provisioning, namely OracleBIApps.rpd.

To configure connection pools in the Oracle BI repository file (OracleBIApps.rpd):

1. Access *ORACLE_HOME*/dwtools/scripts and run configure_rpd.py using the following syntax:

   WLST_PATH configure_rpd.py '*DOMAIN_HOME*' '*DW_DB_CONNECT*' '*DW_DB_SCHEMA*' '*BI_INSTANCE_HOME*' '*ADMIN_HOST*' '*ADMIN_PORT*' '*WL_USERNAME*'

   where:

   - *WLST_PATH* is the path to the WebLogic Scripting Tool. Use the version of wlst that is found in *ORACLE_HOME*/common/bin.

   - *DOMAIN_HOME* is the location of the Oracle BI domain. For example,\MW_HOME\user_projects\domains\bifoundation_domain.

   - *DW_DB_CONNECT* is the connection string for the database that stores the DW schema that you created when you performed the setup steps in Section 3.4.2, "Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications". For example:

     (DESCRIPTION=(ADDRESS_

```
LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521)))(CONNECT_
DATA=(SERVICE_NAME=orcl)))
```

- *DW_DB_SCHEMA* is the name of the DW schema that you created when you performed the setup steps in Section 3.4.2, "Setup Step: Create Schemas and Oracle Business Analytics Warehouse Objects for Oracle BI Applications". For example, you might have specified the name DEV11G_DW.

- *BI_INSTANCE_HOME* is the home of your BI instance.

- *ADMIN_HOST* is the machine that hosts the Administration Server

- *ADMIN_PORT* is the port number for the Administration Server

- *WL_USERNAME* is the WebLogic username

For example:

```
/mw_home/Oracle_BI1/common/bin/wlst configure_rpd.py '/mw_home/user_
projects/domains/bifoundation_domain' '(DESCRIPTION=(ADDRESS_
LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521)))(CONNECT_
DATA=(SERVICE_NAME=orcl)))' 'DEV1_DW' '/mw_home/instances/instance1'
'localhost' '7001' 'weblogic'
```

> **Note:** When you run the script, you are also prompted for passwords.

The script sets the following static variables:

- OLAP_USER is set to the value of the Oracle Business Analytics Warehouse table owner (*prefix*_DW schema).

- OLAP_DSN is set to the connect string for Oracle Business Analytics Warehouse.

- CM_USER is set to the value of the Oracle Business Analytics Warehouse table owner (*prefix*_DW schema).

- CM_DSN is set to the connect string for Oracle Business Analytics Warehouse.

The script also sets the password of the *prefix*_DW schema on the Oracle Data Warehouse Connection Pool, the Oracle Data Warehouse Repository Initblocks Connection Pool, and the Connection Pool for Oracle BI Applications Configuration Manager. The Oracle DB InitBlock Initialization Connection Pool is configured to the same values as those for the HCM_OLTP connection.

2. Open the repository file OracleBIApps.rpd in the BI Administration Tool, select Manage, then Variables to display the Variable Manager, select the OLAPTBO variable in the Static list, and set the variable to the Oracle Business Analytics Warehouse table owner (*prefix*_DW schema).

3. Using OPMN, restart the BI Server.

3.4.10 Setup Step: Configure Oracle HTTP Server

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Configuring the Oracle HTTP Server

Work with your Web administrator to configure Oracle HTTP Server so that URLs for Oracle BI Applications Configuration Manager, Functional Setup Manager, and DAC Server point to the virtual host and port for the BI domain. Ensure that you configure the same Oracle HTTP Server for Oracle BI Applications Configuration Manager, Functional Setup Manager, and DAC Server.

To configure Oracle HTTP Server:

1. Access the directory *APPTOP*`/instance/CommonDomain_`
   `webtier/config/OHS/ohs1/moduleconf`

   where *APPTOP* is the Application home folder.

2. Open the file FusionVirtualHost_bi.conf.

3. Under the #BIEE section, add the following entries for Oracle BI Applications Configuration Manager, Functional Setup Manager, and DAC Server:

   - Oracle BI Applications Configuration Manager entry:

     ```
     <LocationMatch ^/biacm >
                 SetHandler weblogic-handler
                     WeblogicCluster host:port
             </LocationMatch>
     ```

     where *host* is the fully-qualified domain name of the WebLogic host where Oracle BI Applications Configuration Manager is deployed and *port* is the Administration Server port.

   - Functional Setup Manager entry:

     ```
     <LocationMatch ^/setup >
                 SetHandler weblogic-handler
                     WeblogicCluster host:port
             </LocationMatch>
     ```

     where *host* is the fully-qualified domain name of the WebLogic host where Functional Setup Manager is deployed and *port* is the Administration Server port.

   - DAC Server entry:

     ```
     <LocationMatch ^/DACServer >
                 SetHandler weblogic-handler
                     WeblogicCluster host:port
             </LocationMatch>
     ```

     where *host* is the fully-qualified domain name of the WebLogic host where the DAC Server is deployed and *port* is the Managed Server port.

   Note: For each LocationMatch entry, make sure that you have a space character between the value (for example, DACServer) and the closing '>' character.

4. Restart Oracle HTTP Server by accessing the directory *APPTOP*`/instance/CommonDomain_webtier/bin` and running the following commands:

   ```
   ./opmnctl stopall
   ./opmnctl startall
   ```

5. Provide the URL for Oracle BI Applications Configuration Manager to your end users.

Configuring the DAC Server URL

You must configure the DAC Server URL to point to the virtual host and port for the BI domain that you specified for the DAC Server in the previous task 'Configuring the Oracle HTTP Server', as follows:

1. In DAC Client, navigate to **Tools** menu, click **DAC Server Management**, and then click **Repository Configuration**.

2. In the Repository Configuration dialog window, enter the following information:

   - **Mode:** Web

   - **URL:** The URL for the DAC Server enterprise application deployed in WebLogic Server. For example, http://*WLS_host*:9704/DACServer.

3.4.11 Setup Step: Configure SSO and Portlet Provider for Oracle BI Applications Configuration Manager and Functional Setup Manager

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Oracle BI Applications Configuration Manager contains a link that launches Functional Setup Manager. Work with your Oracle BI Applications administrator and your Security Administrator to configure single sign-on (SSO) between Oracle BI Applications Configuration Manager and Functional Setup Manager using Oracle Access Manager. Without SSO configuration, no single sign-on capabilities are provided, however users with the appropriate Oracle BI Application duty roles can still access Oracle BI Applications Configuration Manager and Functional Setup Manager.

To configure Oracle BI Applications and FSM, do the following:

1. Configure SSO as described in Section 3.4.11.1, "How to configure SSO for Oracle BI Applications Configuration Manager and FSM".

2. Configure Portlet Provider as described in Section 3.4.11.2, "How to configure Portlet Provider for Oracle BI Applications Configuration Manager and FSM".

3.4.11.1 How to configure SSO for Oracle BI Applications Configuration Manager and FSM

To configure SSO for Oracle BI Applications Configuration Manager and Functional Setup Manager using Oracle Access Manager:

1. In a supported Web browser, login to Oracle Access Manager.

   For example, using the following URL:

   ```
   http://<host>:<port>/oamconsole/
   ```

   where *host* is the host name and *port* is the port number for your Oracle Access Manager application.

2. Display the Policy Configuration tab.

3. Create a policy for Oracle BI Applications Configuration Manager, as follows:

a. In the Browse pane, navigate to Application Domains\bi.

b. Double-click the **Resources** node, to display the bi Resources page.

c. Use the New Resource option to display the Create Resource page.

d. Specify the following details:

Resource type: `http`

Host Identifier: `OraFusionApp`

Resource URL: /biacm/.../*

e. Save the details.

4. Determine whether there is an existing policy for Functional Setup Manager (for example, there might be an existing policy for the Resource URL /setup* in the Application Domains\fs domain).

   If there is an existing policy for Functional Setup Manager, then you can skip step 5 below and use this existing policy. If there is not an existing policy for Functional Setup Manager, then follow step 5 below to create a policy.

5. Create a policy for Functional Setup Manager, as follows:

   a. In the left hand Browse pane, navigate to Application Domains\bi.

   b. Double-click the **Resources** node, to display the bi Resources page.

   c. Use the New Resource option to display the Create Resource page.

   d. Specify the following details:

   Resource type: `http`

   Host Identifier: `OraFusionApp`

   Resource URL: /setup/.../*

   e. Save the details.

3.4.11.2 How to configure Portlet Provider for Oracle BI Applications Configuration Manager and FSM

To configure Portlet Provider for Oracle BI Applications Configuration Manager and Functional Setup Manager using Oracle Access Manager:

1. In a supported Web browser, login to Oracle Access Manager.

   For example, using the following URL:

   ```
   http://<host>:<port>/oamconsole/
   ```

   where `host` is the host name and `port` is the port number for your Oracle Access Manager application.

2. Display the Policy Configuration tab.

3. Create a policy for Oracle BI Applications Configuration Manager, as follows:

   a. In the Browse pane, navigate to Application Domains\bi.

   b. Select the **Resources** node.
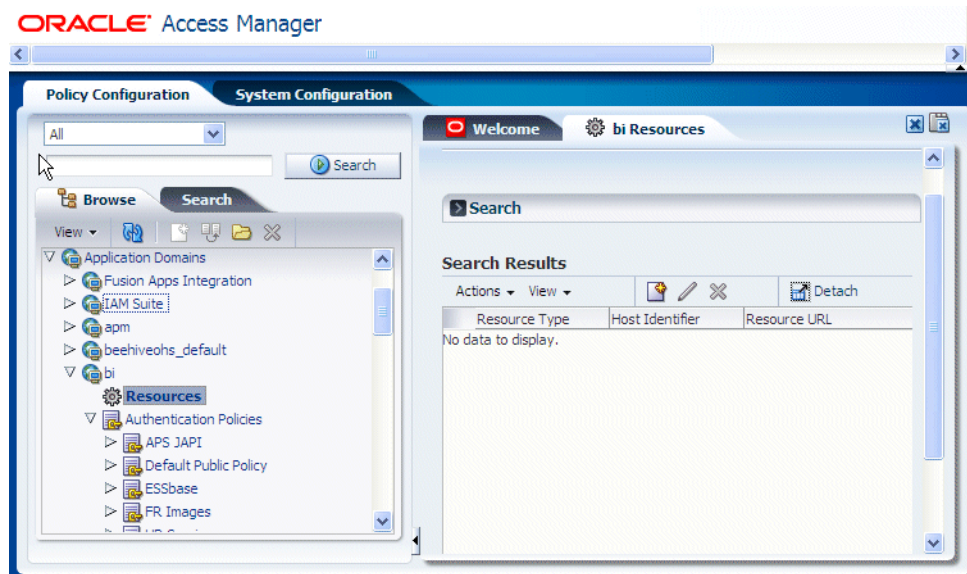
    c. Use the New Resource option to display the Create Resource page.

    d. Specify the following details:

        Resource type: `http`

        Host Identifier: `OraFusionApp`

        Resource URL: /setup/portlets/…/*

        Protection Level: `Excluded`
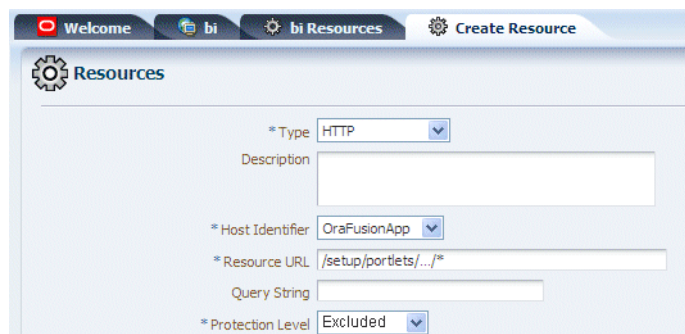
    e. Save the details.

3.4.12 Setup Step: Installing and Setting Up Informatica PowerCenter Client Tools (Windows)

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

3.4.12.1 Install PowerCenter Client Tools

This section explains how to install Informatica PowerCenter V9.1 Hotfix 2 Client Tools for an Oracle BI Applications deployment. For detailed generic information about installing Informatica V9.1, refer to *Informatica PowerCenter Installation and Configuration Guide*.

You must install Informatica PowerCenter Client Tools on the same Windows machine as the DAC Client.

For information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica, see the Certification Document for Oracle BI Applications:

http://www.oracle.com/technology/software/products/ias/files/fus ion_certification.html

To install Informatica PowerCenter Client Tools on Windows:

1. Run the program install.bat from the root directory where you unzipped the PowerCenter Client Tools V9.1 install package.

   For information about the location of the Informatica installation files, see Section 3.4.9.1.3, "Locations of Installation Files for Informatica PowerCenter and Hotfix 2".

2. Follow the instructions in the Informatica PowerCenter Client 9.1 installation wizard. For more information about installing PowerCenter Client, refer to *Informatica PowerCenter Installation and Configuration Guide*.

3. Copy the Windows pmcmd program file into the PowerCenter Client bin directory, as follows:

   a. Obtain the pmcmd.exe file by doing one of the following:

   On the PowerCenter client machine, obtain the pmcmd.exe file from the following directory:

   `INFA_HOME\clients\PowerCenterClient\CommandLineUtilities\PC\server\bin`

   Alternatively, if you have access to a Windows version of PowerCenter Services, then the pmcmd.exe file is located in the following directory on the machine where PowerCenter Services is installed:

   `INFA_HOME\server\bin`

   b. Copy the pmcmd.exe file to the following directory on the PowerCenter Client/DAC Client machine:

   `INFA_HOME\clients\PowerCenterClient\client\bin`

   Note: PowerCenter Client and DAC Client should be installed on the same Windows machine.

3.4.12.2 Setup Step: Accessing the Informatica Domain and Repository from Informatica Client Tools

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

When you first log into any of the Informatica PowerCenter Client tools (for example, Designer, Workflow Manager), you must specify the Informatica Domain and Repository that you want to use.

Note: The following tasks describe how to use Informatica PowerCenter Designer to specify an Informatica PowerCenter Domain and Informatica PowerCenter Repository. However, you can use any of the Informatica PowerCenter Client tools.

To set up the Informatica Domain:

1. Start Informatica PowerCenter Designer.

2. Click **Repository**, and then **Configure Domain**, to display the Configure Domains dialog.
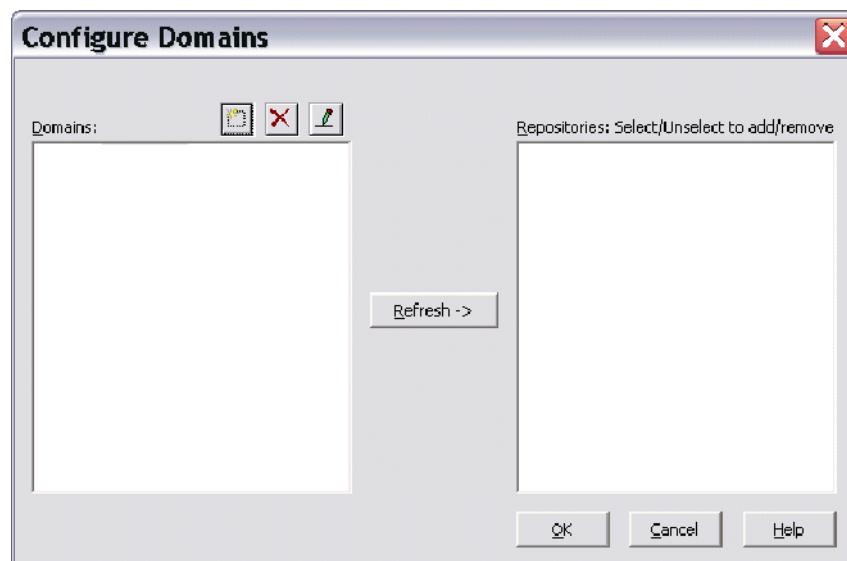
3. Click the **Add a new Domain** icon to display the Add Domain dialog.

4. Use the Add Domain dialog to specify the following:

Table 3–5   Add Domain dialog fields

Field Name	What to enter
Domain Name	Specify the name of the domain that you created when you installed Informatica PowerCenter Services. The default domain name is 'Domain_<HOSTNAME>'. For example, if the hostname of the Informatica PowerCenter Services machine is abc.com, then the domain name is Domain_abc.
Gateway Host	Specify the hostname or IP address of the Informatica Server machine that hosts the Gateway node (for example, machineA).
Gateway Port	Specify the gateway port number. For example, 6001. You can obtain the port number using Informatica Administrator.

5. Click **OK**.

When you specify a valid set of Domain details, the new domain is displayed in the Domains list, and any available Repositories are displayed in the Repositories list.
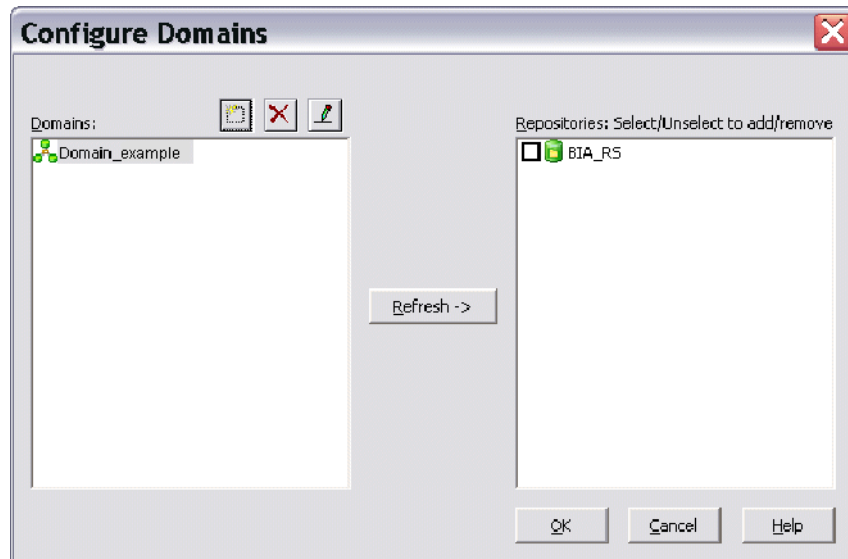
6. In the **Repositories** list, select the check box for the Repository that you want to use.
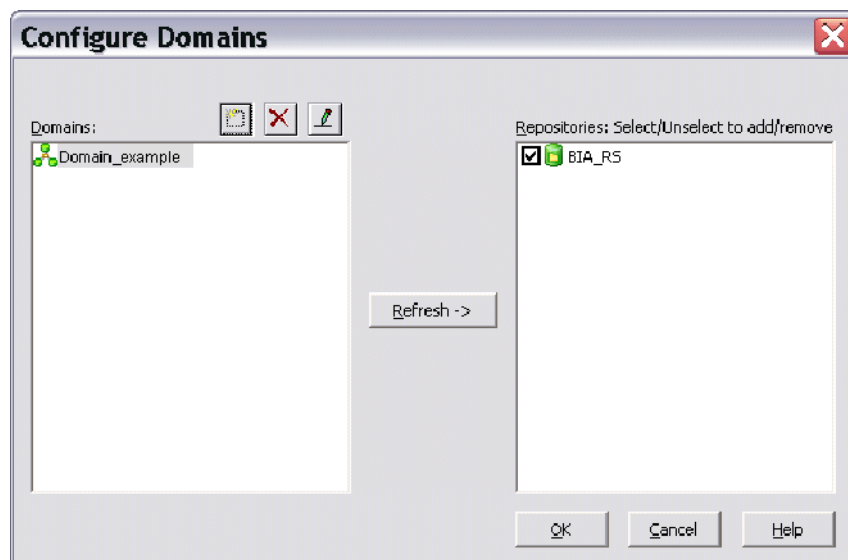
To set up the Informatica Repository:

1. Start Informatica PowerCenter Designer.

2. Select the Repository, and then click **Add**, to display the Add Repository dialog.
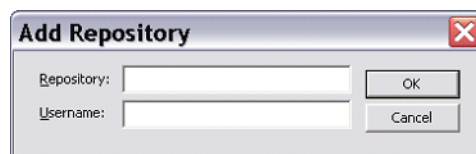
3. In the Repository field, type the name of the Repository Service that was created by the WCA script or created manually (for example, BIA_RS), and specify Administrator as the user name.

4. Click **OK**, to save the details and to return to the Start Page.

5. In the Repositories tree, double-click the Repository that you specified in the Step 3 (for example, BIA_RS).
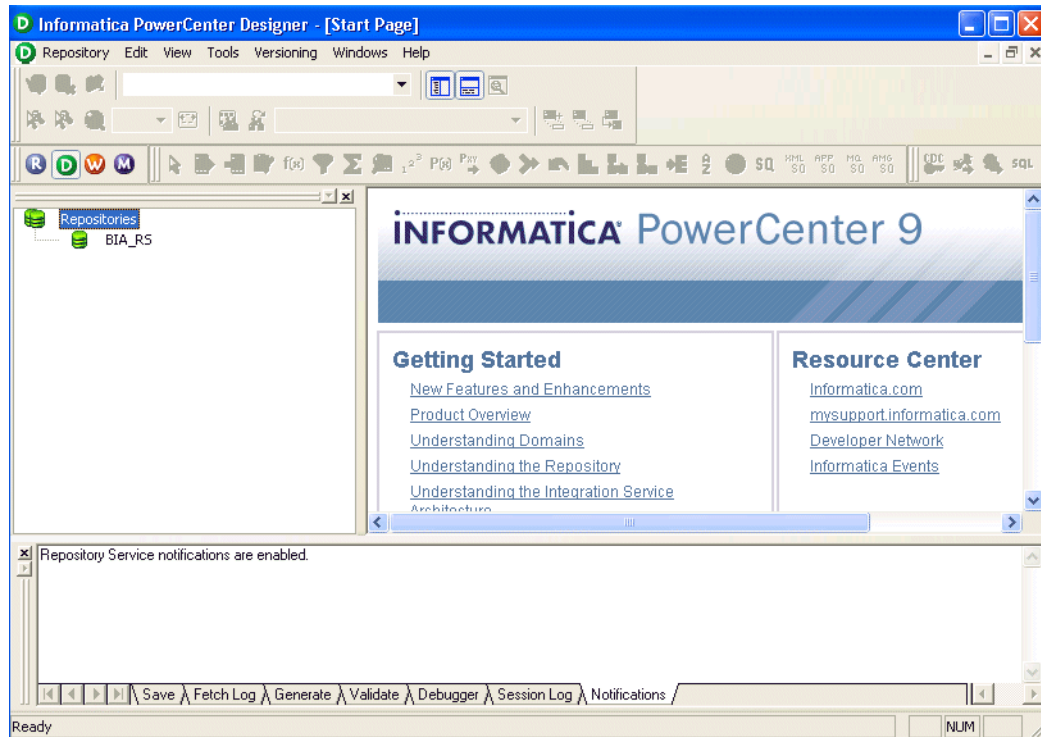
6. In the Connection Settings part of the Connect to Repository dialog, ensure that the **Domain** field displays the domain that you specified in the previous task.

7. Specify Administrator as the user name and password, and then click **Connect**.
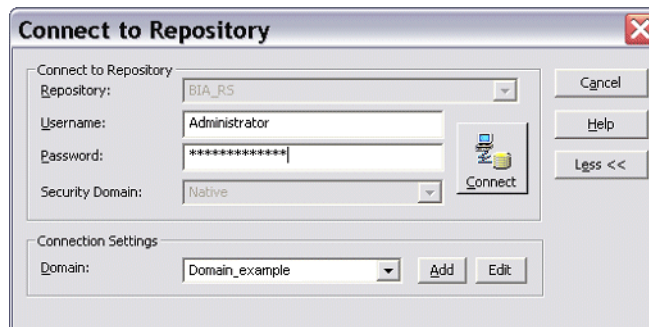
8. You can now expand the repository in the Repositories list to display the contents.
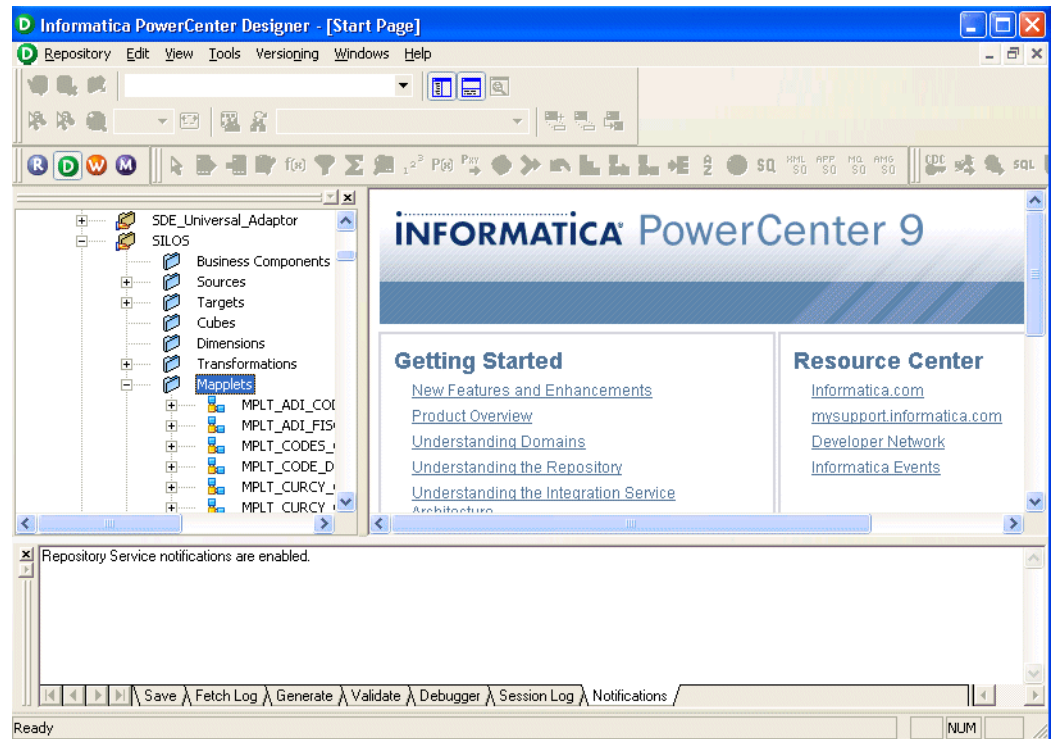
3.4.12.3 Setup Step: Creating the Repository Administrator User in the Native Security Domain

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

For DAC to access Informatica and perform tasks in pmcmd and pmrep command line programs, DAC must log in to Informatica as an Informatica Repository Administrator user. This user must be configured in the native security domain.

You must create such a Repository Administrator user, or, if your security policies permit, then you can use the default Administrator user (whose privileges include Repository administration) for DAC connectivity to Informatica.

For more information on creating users and Informatica security domains, refer to the Informatica documentation.

To use the default Administrator user in the native security domain:

1. Open Informatica Administrator and display the Security tab.

    For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Users area, expand the Native directory and select Administrator.

    Note: Do not select an existing user from an LDAP domain.

3. Display the Privileges tab, and use the make sure that the correct Domain and Repository Service are selected.
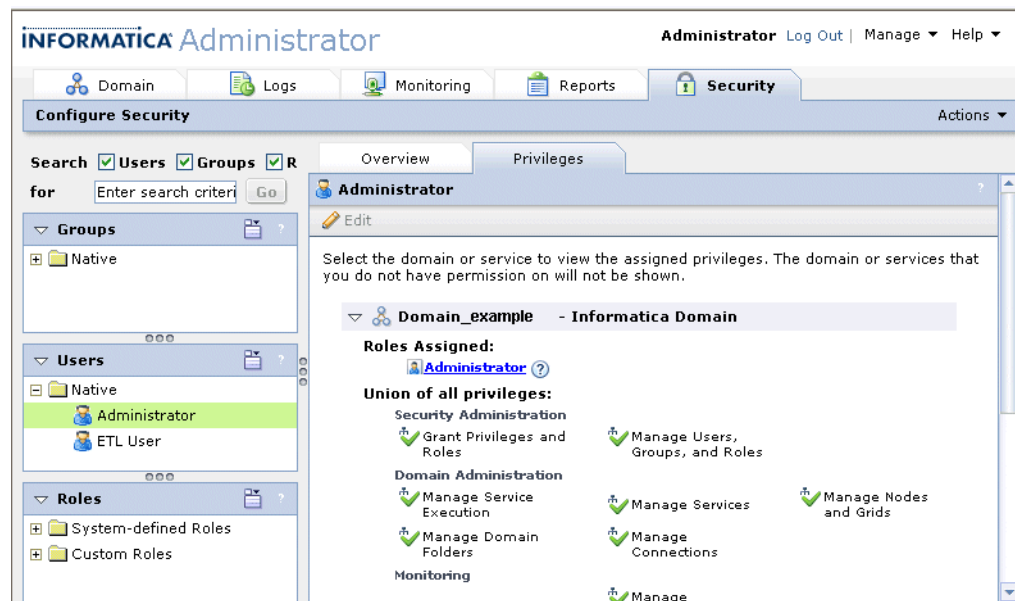
4. Click OK.

To create a new Repository Administrator defined in the native security domain:

1. Open Informatica Administrator and display the Security tab.

    For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Users area, click Actions, then Create User to display the Create User dialog.

    Note: Do not create this user in an LDAP domain.

3. Use the Create User dialog to specify the user details, then click OK.

4. In the Users area, select the new user, display the Privileges tab, then click Edit to display the Edit Roles and Privileges dialog.

5. Use the Roles tab and Privileges tab to select the appropriate Domain and Repository Service.
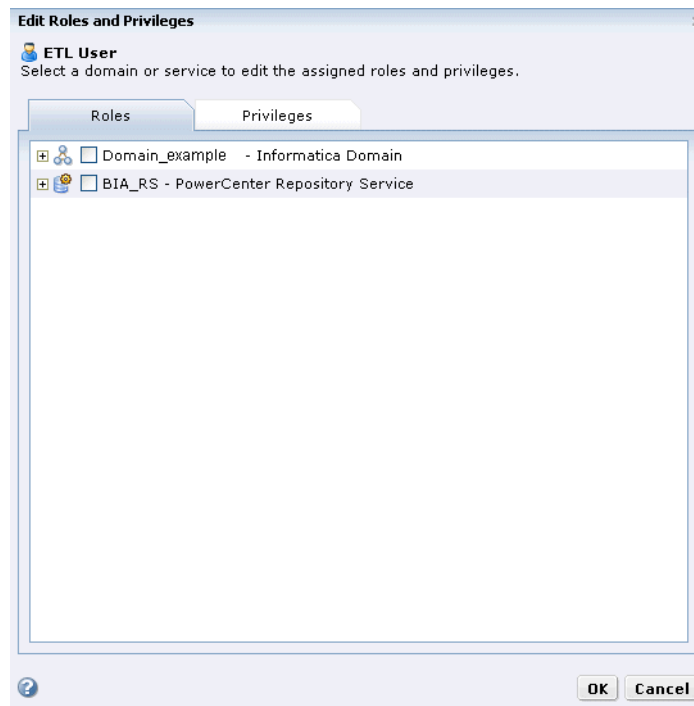
6. Click OK.

3.4.13 Setup Step: Install and Set Up DAC Client

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

To install and set up DAC Client, do the following:

- Install the DAC Client, as described in Section 3.4.13.1, "How to Install DAC Client".

  Note: PowerCenter Client and DAC Client should be installed on the same Windows machine.

- Copy the authentication file to the machine that hosts the DAC Client, as described in Section 3.4.13.2, "How to copy the DAC Authentication File to the DAC Client Machine".

- Log into DAC Client for the first time, as described in Section 3.4.13.3, "How to log into DAC Client for the first time".

For information about the DAC Client installer, see Section 3.4.13.4, "About DAC Configuration Handled by the Oracle BI Applications Client Installer".

3.4.13.1 How to Install DAC Client

DAC administrators must install DAC Client on the Windows-based machine where the Informatica PowerCenter Client Tools were installed. If Informatica PowerCenter Client Tools are not collocated on the same machine, then DAC Client must have access to the Informatica PowerCenter binaries and the domains.infa file on a shared network location.

You can obtain the Oracle BI Applications Client Installer executable from the folder `<Oracle Home for BI>\clients\biappsclients`.

For more information on DAC Client requirements, see Section 2.2, "DAC Client Installation Requirements" in the *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Data Warehouse Administration Console*.

To install the DAC Client using the Oracle BI Applications Client installer:

1. Start DAC Client installer by double-clicking the biapps_client_install.exe file.

    The installation wizard opens to the Introduction page.

2. In the Introduction dialog, click **Next** to proceed. The Informatica Server Location page opens.

3. In the Informatica Server Location page, enter the location of the Informatica PowerCenter Server folder (that is, the *INFA_HOME*), or click **Choose...** to navigate to the folder location. For example, c:\Informatica\PowerCenter9.1.

    Click **Next** to continue. The Informatica Domain File page opens.

4. In the Informatica Domain File page, enter the location of the Informatica PowerCenter domains file domains.infa, or click **Choose...** to navigate to the file. For example, c:\Informatica\PowerCenter9.1\domains.infa.

    Note: Informatica creates the domains.infa file in this location when you set up Informatica Client tools to access the Informatica Domain and Repository, as described in Section 3.4.12.2, "Setup Step: Accessing the Informatica Domain and Repository from Informatica Client Tools".

    Click **Next** to continue. The Choose Install Folder page opens.

5. In the Choose Install Folder page, enter the absolute path for the folder where you want to install DAC Client, or click **Choose...** to navigate to and specify the folder. For example, you might specify C:\BIA_11g_Clients.

    The installation directory path can contain alphanumeric, underscore (_), hyphen (-) or dot (.) characters and must begin with an alphanumeric character.

    The directory in which you install DAC is referred to as the Oracle Home location.

    Click **Next** to continue. The Choose Shortcut Folder page opens.

6. In the Choose Shortcut Folder page, use the radio buttons to specify where you would like to create product icons on the Windows machine. For example, you might want to add a DAC Client icon to the Windows Start Menu.

    Click **Next** to continue. The Pre-Installation Summary page opens.

7. In the Pre-Installation Summary page, verify your selections. To return to previous install options, click **Previous**.

    To continue with the installation, click **Install**.

8. When the installation is complete, click **Done**.

3.4.13.2 How to copy the DAC Authentication File to the DAC Client Machine

DAC Client uses an authentication file (cwallet.sso) to access the DAC Repository. The authentication file is automatically generated when WCA extends the BI domain to include the DAC Server (as described in Section 3.4.8.1, "Setup Step: Configure Oracle BI Applications components using Warehouse Component Deployment Assistant (WCA) in a UNIX Environment"). The authentication file can also be created by a DAC user at the time of the first connection to the DAC repository.

The authentication file is saved by default to the following directory:

```
DOMAIN_HOME\dac\conf-shared\security\repository
```

You must copy the cwallet.sso authentication file to the machine where DAC Client is installed. An administrator must also distribute this authentication file to all DAC Client users who intend to access the DAC Repository.

For more information about the authentication file, see Section 4.2, "About the DAC Authentication File" in the Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Data Warehouse Administration Console.

3.4.13.3 How to log into DAC Client for the first time

To log into DAC Client for the first time:

1. Start the DAC Client by navigating to the \dac directory and double-clicking the startclient.bat file.

   The Login dialog opens.

2. In the Login dialog, select Configure.

3. In the Configuring dialog, select Create Connection, and then click Next.

4. Enter the appropriate connection information:

   - Name: Specify a short unique name for the connection to the DAC repository.

     This name is displayed in the list of connections available at the DAC login dialog.

   - Connection type: Choose Oracle (Thin).

   - Service Name: Enter the service name for the DAC Repository (for example, XE).

   - Database Host: Enter the name of the computer that hosts the DAC Repository database.

   - Database Port: Enter the port number for the database.

5. Click in the **Authentication File** field, and then do the following:

   a. In the Authentication File dialog, click Choose existing authentication file, and then click OK.

   b. Navigate to the appropriate directory, select the cwallet.sso file that you copied from the DAC Server computer, and click OK.

   c. In the Configuring dialog, click Test Connection to confirm the connection works.

   d. Click Apply.

   e. Click Finish.

6. In the **Connection** field of the Login dialog, select the Connection that you created.

7. Use the **User name** and **Password** fields to specify the credentials of a user that has been granted the role 'BI Applications Administrator Duty'.

   Note: Users are set up with appropriate roles in step Section 3.4.6, "Setup Step: Grant User Access to Oracle BI Applications Components".

8. In the **Authentication Type** field, choose FMW, then click Login.

3.4.13.4 About DAC Configuration Handled by the Oracle BI Applications Client Installer

The Oracle BI Applications Clients installer performs the following configuration:

- In the config.bat file, the DAC installer configures the JAVA_HOME and DAC_ HOME variables.

- In the dac_env.bat file, the DAC installer creates an environment variable named INFA_DOMAINS_FILE and sets the value to the directory path of the domans.infa file.

- The DAC installer adds the directory path to Informatica PowerCenter binaries to the PATH environment variable.

> **Note:** If you reinstall or upgrade Informatica PowerCenter without reinstalling DAC, you need to manually set these environment variables.

3.4.14 Setup Step: Create an ODBC DSN to the Oracle BI Server

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

The Informatica PowerCenter Integration Service connects to the Oracle BI Server (for access to the Oracle Fusion Applications source system). The Integration Service accesses the Oracle BI Server using the Oracle BI ODBC Data Source Name (DSN). You must install and configure the Oracle BI ODBC DSN on the machine that hosts Informatica PowerCenter Services.

The procedure for creating the ODBC DSN is specific to the operating system (Windows, Linux, or UNIX) of the machine that hosts Informatica PowerCenter Services.

Procedure for Windows

To create an ODBC DSN to the Oracle BI Server:

1. On the machine that hosts Informatica PowerCenter Services, display the ODBC Data Source Administrator dialog.

   For example, open the Windows Control Panel, select **Administrative Tools**, and then select **Data Sources (ODBC)**.

2. Display the System DSN tab, and then click **Add** to display the Create New Data Source dialog.

3. Select the driver **Oracle BI Server** *<id_number>* (where *id_number* is a number specific to an installed Oracle home, for example '1'), and then click **Finish**.

   The first page of the Oracle BI DSN Configuration wizard is displayed.

   > **Note:** If the Oracle BI Server 11*g* driver is not installed on the machine, then you can install the BI Administration Tool using the Oracle Business Intelligence Client installer. For more information about installing Oracle Business Intelligence Client Tools, see Section 3.4.4, "Setup Step: Install Oracle BI Administration Tool".

4. Type a name for the data source in the **Name** field. For example, `AnalyticsWeb`.

   Note: Do not change the name of the centrally managed default DSN for Presentation Services and the Administration Tool.

5. In the **Server** field, select the machine that hosts the Oracle BI Server that will be used to run ETL.

If the server name does not appear in the list, then type the machine name in the **Server** field.

6. To connect to a set of clustered Oracle BI Servers through the Cluster Controllers, do the following:

a. Select **Clustered DSN**.

b. Enter information for the primary and secondary Cluster Controllers, as follows:

– In the **Primary Controller** field, enter the name of the machine that is specified as the primary Cluster Controller. Then, enter the port number for the Primary Controller in the appropriate **Port** field.

– If a secondary Cluster Controller has been set up, then type the name of the host where the Secondary Controller is running in the **Secondary Controller** field. Then, enter the port number for the Secondary Controller in the appropriate **Port** field.

You can find information about the Primary and Secondary Controller hosts and ports on the Availability tab of the Capacity Management page in Fusion Middleware Control.

c. To test the connection to the Cluster Controllers, click **Test Cluster Connect**.

If the test is not successful, then correct any errors identified in the message and test the connection again.

Note: Do not change the Primary Controller, Secondary Controller, and Port parameters of the centrally managed default DSN for Presentation Services.

7. Select **Route Requests To Physical Layer** to create a dedicated physical connection to the Oracle BI Server for clients using this DSN. All SELECT queries over this connection will be treated as SELECT_PHYSICAL queries.

SELECT_PHYSICAL queries directly query objects in the Physical layer of the metadata repository, bypassing the Presentation layer and the Business Model and Mapping layer.

8. Select **Use Forward Only Cursor** to change the ODBC cursor from its default scrollable mode to forward only.

9. Click **Next** to display the second page of the wizard.

10. Use the **Login ID** and **Password** fields to specify credentials for the Oracle BI Server (for example, weblogic, welcome1).

11. Enter a port number for the Oracle BI Server in the **Port** field. This option is enabled when **Clustered DSN** has not been selected on the first page of the wizard.

You can find information about the Oracle BI Server port on the Availability tab of the Capacity Management page in Fusion Middleware Control.

12. Click **Next** to display the third page of the wizard.

13. Select **Use Regional Settings when outputting dates and times**. When this option is enabled, data in Date, Time, or DateTime format is displayed using Windows regional settings, rather than the default format for the Oracle BI Server. To see the Windows regional settings, open the Control Panel and double-click Regional and Language Options.

14. Click **Finish** to save the DSN configuration.

Procedure for Linux and UNIX

> **Note:** The Integration Service user account must have access to the shared Middleware Home for BI. The Integration Service must be able to access the ODBC library files for the BI Server.

To create an ODBC DSN to the Oracle BI Server:

1. On the machine that hosts the Integration Service, configure the Informatica ODBC.ini file (in I*NFA_HOME*/ODBC5.2/ or *INFA_HOME*/ODBC6.0/) with an entry for the BI Server ODBC.

For a single BI Server, the ODBC.ini file entries should appear similar to the following:

```
[ODBC_DSN_NAME]
Driver= BI_SERVER_ODBC_PATH
Description=Oracle BI Server
ServerMachine= BI_SERVER_HOST
Repository=
Catalog=
UID=
PWD=
Port=PORT
ForwardOnlyCursor=Yes
SelectPhysical=Yes
IsClusteredDSN=No
Regional=Yes
PrimaryCCSPort=
SSL=No
PrimaryCCS=
```

where:

- *ODBC_DSN_NAME* is the name you are assigning to the data source. For example, `Analytics Web ETL`.

  Note: If you set up Oracle BI Applications using WCA, then you must provide the same name here for the ODBC DSN as you specified in the WCA properties file for the BI_SERVER_DSN parameter.

- *BI_SERVER_ODBC_PATH* is the path to the ODBC library file for the BI Server. For example, *ORACLE_BI_ HOME*/bifoundation/server/bin/libnqsodbc64.so.

- *BI_SERVER_HOST* is the fully qualified name of the machine hosting the BI Server.

- *PORT* is the port number of the BI Server. For example, `9703`.

For a BI Server cluster, the ODBC.ini file entries should appear similar to the following:

```
[ODBC_DSN_NAME]
Driver=BI_SERVER_ODBC_PATH
Description=Oracle BI Server
ServerMachine=BI_SERVER_HOST
Repository=
Catalog=
```

```
UID=
PWD=
Port=
ForwardOnlyCursor=Yes
SelectPhysical=Yes
IsClusteredDSN=Yes
Regional=Yes
PrimaryCCSPort=PORT_CLUSTER
SSL=No
PrimaryCCS=BI_SERVER_CLUSTER_CONTROLLER
```

where:

- *ODBC_DSN_NAME* is the name you are assigning to the data source. For example, `Analytics Web ETL`.

- *BI_SERVER_ODBC_PATH* is the path to the ODBC library file for the BI Server. For example, *ORACLE_BI_HOME*/bifoundation/server/bin/libnqsodbc64.so.

- *BI_SERVER_HOST* is the fully qualified name of the machine hosting the BI Server.

- *PORT_CLUSTER* is the port number of the primary controller for the BI Server cluster. For example, `9706`.

- *BI_SERVER_CLUSTER_HOST* is the name of the machine hosting the BI Server cluster controller.

2. Set the following Environment Variables for Linux and UNIX:

- INFA_HOME

- INFA_DOMAINS_FILE

- JAVA_HOME

- JRE_HOME

- LD_LIBRARY_PATH

- ODBCINI

- ORACLE_HOME

- PATH

- TNS_ADMIN

For example, (use appropriate shell commands for your platform):

```
export TNS_ADMIN=/oracle/db_1/network/admin
export ORACLE_HOME=/oracle/db_1/
export PATH=/oracle/db_1/bin:$PATH
run bi-init.sh
. ./bi-init.sh
export PATH=.:/INFA/server/tomcat/bin:/INFA/server/bin:/INFA/java/
jre:/INFA/ODBC5.2/bin:$PATH
export LD_LIBRARY_PATH=/oracle/db_1/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_
PATH=/INFA/server/bin:/INFA/ODBC5.2/lib:/INFA/java/lib:/INFA/java/jre:$LD_
LIBRARY_PATH
export JRE_HOME=/INFA/java/jre
export JAVA_HOME=/INFA/java/jre
export ODBCINI=/INFA/ODBC5.2/odbc.ini
export INFA_HOME=/INFA
```

```
export INFA_DOMAINS_FILE=/INFA/domains.infa
```

3. In Informatica Administrator, restart the Integration Service (that is, choose Disable, then Enable).

3.4.15 Setup Step: Configure Oracle BI Applications Components Manually

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

If you are configuring some Oracle BI Applications components manually (that is, you are not using WCA), then you must perform the following steps:

- Section 3.4.15.1, "Setup Step: Register Informatica in DAC and Configure DAC Integration Settings"

- Section 3.4.15.2, "Setup Step: Configuring Physical Data Source Connections in DAC"

- Section 3.4.15.3, "Setup Step: Configuring Relational Connections in Informatica"

- Section 3.4.15.4, "Setup Step: Setting the SiebelUnicodeDB Property in Informatica Integration Services"

- Section 3.4.15.5, "Setup Step: Enabling User Currency Preference Settings"

- Section 3.4.15.6, "Setup Step: Registering Source Systems"

3.4.15.1 Setup Step: Register Informatica in DAC and Configure DAC Integration Settings

This section explains how to configure DAC integration settings. It contains the following topics:

- Section 3.4.15.1.1, "Registering Informatica Services in DAC"

- Section 3.4.15.1.2, "Setting DAC System Properties"

3.4.15.1.1 Registering Informatica Services in DAC This section explains how to register the Informatica PowerCenter Integration Services and the Informatica PowerCenter Repository Service in DAC.

When you register Informatica Services in DAC, note the following:

- You must register one or more Integration Services service.

- You must register one Repository Service.

To register Informatica Services in DAC:

1. In DAC, navigate to the Setup view, and then display the **Informatica Servers** tab.

2. For the Informatica Integration Service that you want to register, do the following:

   a. Modify the record with Name = INFORMATICA_DW_SERVER by entering the following information in the Edit subtab:

Field	Enter or select
Name	Logical name for the Integration Service.
Type	Informatica

Field	Enter or select
Service	Specify the name of the Integration Service that was created manually or was created by the WCA script (for example, BIA_IS). **Tip**: To find out the name of the Integration Service, log into Informatica Administrator and navigate to the Domain that you created, and note down the name of the Integration Service within that Domain.
Server Port	No longer used.
Domain	Informatica domain name.
Login	Informatica Repository user name that has appropriate privileges to execute workflows (for example, Administrator). **Note:** DAC must log in to Informatica as an Informatica Repository Administrator user that is configured in the native security domain. See Section 3.4.12.3, "Setup Step: Creating the Repository Administrator User in the Native Security Domain."
Password	Informatica Repository user password.
Maximum Sessions	Maximum number of workflows that can be executed in parallel on the Informatica PowerCenter Integration Services service. If the number of sessions is zero or is not specified, the DAC Server assigns the default value of 10. Adjust this value based on the topology and configuration of your deployment. Allowing too many sessions can create excessive loads, which may result in failures.
Repository Name	Name of the Repository Service. For example, Oracle_BIA_RS.
Inactive	Indicates whether the PowerCenter Integration Services service will participate in the ETL process.

    b. Click **Test Connection** to ensure that the connection works.

        Note: Integration Services must be running.

    c. Click **Save** to save the details.

3. For the Informatica Repository Service you want to register, do the following:

    a. Modify the record with Name = INFORMATICA_REP_SERVER by entering the following information in the Edit subtab:

Field	Enter
Name	Logical name for the Repository Service. Specify the name of the Repository Service that was created manually or was created by the WCA script (for example, BIA_RS).
Type	Repository.
Hostname	Repository Service host name or IP address.
Server Port	Informatica Gateway port number. For example, 6001.
Domain	(Read only) The Domain for the Hostname specified by the **Hostname** field. If the correct Domain is not displayed, click Refresh at the top of the Informatica Servers tab to refresh the field.

Field	Enter
Login	Informatica Repository user name that has appropriate privileges to execute workflows (for example, Administrator).
Password	Informatica Repository user password (for example, Administrator).
Maximum Sessions	Maximum number of workflows that can be executed in parallel on the Informatica PowerCenter Integration Services service. If the number of sessions is zero or is not specified, the DAC Server assigns the default value of 10.
Repository Name	Name of the Repository Service. For example, BIA_RS.
Inactive	Indicates whether the Repository Service will participate in the ETL process.

    b. Click **Test Connection** to ensure that the connection works.

      Note: The Repository Service must be running.

    c. Click **Save** to save the details.

3.4.15.1.2 Setting DAC System Properties This section describes how to set DAC System Properties to ensure proper integration between DAC Client, the DAC Server, and Informatica.

To set DAC System Properties:

1. Start DAC Client by navigating to the \dac directory and double-clicking the startclient.bat file. The Login dialog opens.

2. In the Login dialog, select the Connection.

3. Enter the User Name and Password for a user with the Oracle BI Applications Administrator Duty Role.

For more information about assigning the Oracle BI Applications Administrator Duty Role, see Section 3.4.6, "Setup Step: Grant User Access to Oracle BI Applications Components".

4. From the **Views** menu, click **Setup**, and then click **DAC System Properties**.

5. Set the value for the InformaticaParameterFileLocation property.

DAC produces parameters files that are consumed by Informatica and writes them to the location specified by this property. By default, the parameter files are created in the following directory:

DOMAIN_HOME\dac\Informatica\parameters

You must either share this location so that Informatica PowerCenter Services has read and write access, or configure an alternate shared location to which both the DAC Server and Informatica PowerCenter Services have read and write access.

3.4.15.2 Setup Step: Configuring Physical Data Source Connections in DAC

Follow this procedure to configure physical data source connections in DAC. You must configure the following data sources:

- DataWarehouse - For the connection to Oracle Business Analytics Warehouse.

- BIACM - For the connection to Oracle BI Applications Configuration Manager.

■ FUSION_V1 - For connection to the Oracle BI Server for the Fusion Application domains.

> **Note:** You must configure connections for the Fusion Applications domains listed above, even if you have not deployed the associated applications.

To configure physical data source connections in DAC:

1. Start DAC Client by navigating to the \dac directory and double-clicking the startclient.bat file. The Login dialog opens.

2. In the Login dialog, select the Connection.

3. Enter the User Name and Password. This user must be granted the Oracle BI Applications Administrator Duty Role. See Section 3.4.6, "Setup Step: Grant User Access to Oracle BI Applications Components."

4. From the Views menu, click **Setup**, and then click **Physical Data Sources**.

   The Physical Data Sources tab displays pre-created records for the data sources listed above.

   Note: In DAC, Data Source Number must be unique, even though this restriction is not enforced when a Data Source Number is specified in DAC Client.

5. Update the records for the following data sources and configure them as specified in the following table:

Name	Type	Connection Type	Table Owner and Password
DataWarehouse	Warehouse	Oracle (Thin)	The table owner and password for Oracle Business Analytics Warehouse.
BIACM	Source	Oracle (Thin)	The table owner and password for Oracle Business Analytics Warehouse. Note: The *prefix*_DW schema contains the Oracle BI Applications Configuration Manager tables.
FUSION_V1	Source	BI Server	The user name and password for the ETL user specified in Section 3.4.7, "Setup Step: Create a User for ETL." The port number for FUSION_V1 can be found in BI Fusion Middleware Control console (see the port for coreapplication_obis1).

6. Click **Test Connection** for each connection to ensure the connection works.

7. Click **Save**.

3.4.15.3 Setup Step: Configuring Relational Connections in Informatica

This section describes how to configure relational connections in Informatica.

Note: Unicode to Unicode is the only supported configuration mode for Oracle Business Intelligence Applications 11g Release (11.1.1).

Before you configure relational connections in Informatica, ensure that Repository Services and Integration Services are enabled and running.

To configure relational connections in Informatica:

1. In Informatica PowerCenter Workflow Manager, select **Connections**, then **Relational,** to display the Relational Connection Browser.

2. Create a Relational Connection of type **Oracle** for DataWarehouse with the following settings:

   - Name - DataWarehouse. The name must match the name of the physical data source connection in DAC. See Section 3.4.15.2, "Setup Step: Configuring Physical Data Source Connections in DAC."

   - Type - **Oracle**

   - User Name - Table owner for the Oracle Business Analytics Warehouse schema (`prefix_DW`).

   - Password - Table owner password

   - Connect String - The tnsnames.ora entry for Oracle Business Analytics Warehouse. This is the tnsnames.ora entry from Section 3.4.5, "Setup Step: Install and Configure Database Connectivity Software."

   - Code Page - UTF-8 encoding of Unicode

   - Attributes - Enable the **Enable Parallel Mode** option.

3. Create a Relational Connection of type **Oracle** for BIACM with the following settings:

   - Name - BIACM. The name must match the name of the physical data source connection in DAC. See Section 3.4.15.2, "Setup Step: Configuring Physical Data Source Connections in DAC."

   - Type - **Oracle**

   - User Name - Table owner for the Oracle Business Analytics Warehouse schema (`prefix_DW`).

   - Password - Table owner password.

   - Connect String - The tnsnames.ora entry for Oracle Business Analytics Warehouse. This is the tnsnames.ora entry from Section 3.4.5, "Setup Step: Install and Configure Database Connectivity Software."

   - Code Page - UTF-8 encoding of Unicode.

   - Attributes - Enable the **Enable Parallel Mode** option.

4. Create a Relational Connection of type **ODBC** for FUSION_V1 with the following settings:

   - Name - FUSION_V1. The name must match the name of the physical data source connection in DAC. See Section 3.4.15.2, "Setup Step: Configuring Physical Data Source Connections in DAC."

   - Type - **ODBC**

   - User Name - The user name for the ETL user specified in Section 3.4.7, "Setup Step: Create a User for ETL."

- Password - ETL user password.

- Connect String - The BI Server ODBC DSN (for example, AnalyticsWebETL) that was created on the machine that hosts the Integration Services. See Section 3.4.14, "Setup Step: Create an ODBC DSN to the Oracle BI Server."

- Code Page - UTF-8 encoding of Unicode.

> **Note:** You must configure connections for the Fusion Applications domains listed above, even if you have not deployed the associated applications.

3.4.15.4 Setup Step: Setting the SiebelUnicodeDB Property in Informatica Integration Services

For data movement mode Unicode to Unicode to work correctly, you must set the custom property SiebelUnicodeDB on Informatica Integration Services. Set the property in the following format:

user_OLTP@ConnectString_OLTP user_OLAP@ConnectString_OLAP.

To set SiebelUnicodeDB property in Informatica Integration Services:

1. Open Informatica Administrator and display the Domain tab.

   For information about how to access Informatica Administrator, see Section A.1, "How to Log Into Informatica Administrator".

2. In the Domain Navigator pane, select the **Integration Service**.

3. In the **Properties** tab, scroll down to the Custom Properties area, and click **Edit**, then click **New**.

4. In the **Name** field, enter the following:

   SiebelUnicodeDB

5. In the **Value** field, enter connect strings for the OLTP and OLAP data sources in the following format:

   user_OLTP@ConnectString_OLTP user_OLAP@ConnectString_OLAP

   where:

   - *user_OLTP* is the user name for the ETL user specified in Section 3.4.7, "Setup Step: Create a User for ETL" It must match exactly the value you entered for the User Name field when creating the relational connection for the OLTP in Informatica Workflow Manager.

   - *ConnectString_OLTP* is the connect string for the BI Server ODBC DSN created in Section 3.4.14, "Setup Step: Create an ODBC DSN to the Oracle BI Server." It must match exactly the value you entered for the Connect String field when creating the relational connections for the OLTP (FUSION_V1_*) in Informatica Workflow Manager.

   - *user_OLAP* is the database user for Oracle Business Analytics Warehouse. It must match exactly the value you entered for the User Name field when creating the relational connection for Oracle Business Analytics Warehouse in Informatica Workflow Manager.

   - *ConnectString_OLAP* is the connect string for the Oracle Business Analytics Warehouse database. It must match exactly the value you entered for the

Connect String field when creating the relational connection for Oracle Business Analytics Warehouse in Informatica Workflow Manager.

Note:

- Ensure that you enter a space between the values for `ConnectString_OLTP` and `user_OLAP`. For example:

  `oltp@db204007.host.com olap@db204008.host.com`

- You must enter the user names and connection strings in the same upper and lower case letters as you used for the relational connections in Informatica Workflow Manager.

3.4.15.5 Setup Step: Enabling User Currency Preference Settings

This section provides instructions for enabling user currency preference settings for Oracle BI Applications. This section contains the following topics:

- Section 3.4.15.5.1, "Enabling the Currency User Preference File for Oracle BI Applications"

3.4.15.5.1 Enabling the Currency User Preference File for Oracle BI Applications The Currency User Preference file used by Oracle BI Applications is userpref_currencies_OBIA.xml. The file instanceconfig.xml used by Oracle BI Presentation Services must point to this file. To do this, you modify the UserprefCurrenciesConfigFile parameter in instanceconfig.xml to point to userpref_currencies_OBIA.xml. If you have deployed a cluster, then you must do this for every instance home in the cluster.

Both instanceconfig.xml and userpref_currencies_OBIA.xml file are located in the following directory:

`INSTANCE_HOME\config\OracleBIPresentationServicesComponent`

To enable user currency preference file for Oracle BI Applications:

1. Access the following directory on the Oracle BI instance:

   `INSTANCE_HOME\config\OracleBIPresentationServicesComponent`

2. Open the file instanceconfig.xml for editing.

3. Modify the UserprefCurrenciesConfigFile parameter in instanceconfig.xml to point to userpref_currencies_OBIA.xml.

4. Save and close instanceconfig.xml.

5. Using OPMN, restart Presentation Services.

3.4.15.6 Setup Step: Registering Source Systems

Register your Oracle Fusion Applications source system using Oracle BI Applications Configuration Manager. Specify a source system to match your Oracle Fusion Applications configuration.

To register a Source System:

1. Log in to Oracle BI Applications Configuration Manager as a user with the Business Intelligence Applications Administrator duty role.

   For more information about how to log in and use Oracle BI Applications Configuration Manager, see Appendix A.2, "How to Start Oracle BI Applications Configuration Manager."

2. Select the **Define BI Applications Instance** link to display the Define Business Intelligence Applications Instance dialog.

3. Display the Source Systems tab.

4. Specify a Source System as follows.

   a. Click the **Add** icon to display the Register Source dialog.

   b. Use the Register Source dialog to specify the following details:

Element	Description
Source Instance Name	Specify a Source Instance name to identify the Fusion Applications source system. For example, 'Oracle Fusion Applications - Production'. The Source Instance name appears in the Oracle BI Applications Configuration Manager dialogs that are available to functional developers.
Description	(Optional) Additional information to assist BI implementors and functional developers.
Data Source Number	A unique ID to identify data from the Oracle Fusion Applications Transactional Database. The Data Source Number entered here must match the DSN specified in the Physical connection in DAC for the Fusion Applications Source of type BI Server. The default Data Source Number for Fusion Applications source is 200. Enter 200 unless you changed this value in DAC. **Note**: In DAC, Data Source Number must be unique, even though this restriction is not enforced when a Data Source Number is specified in DAC Client.
Select Fusion Applications	Select the **Select** check box for each application that is deployed in the Oracle Fusion Applications database instance. For example, if you have all three Fusion Applications deployed in one Fusion Applications instance, select all three check boxes.

   c. Save the details.

> **Note:** When you register your source system in Oracle BI Applications Configuration Manager, the appropriate setup seed data is enabled based on your Oracle Fusion Applications selections. For example,

3.4.16 Setup Step: Enable Offerings for Deployment

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Use Oracle BI Applications Configuration Manager to enable the Oracle BI Applications Offerings that you have purchased.

To enable Oracle BI Applications Offerings for deployment:

1. Login to Oracle BI Applications Configuration Manager as a user with the Business Intelligence Applications Administrator duty role.

   For more information about how to log in and use Oracle BI Applications Configuration Manager, see Section A.2, "How to Start Oracle BI Applications Configuration Manager."

2. Select the **Manage BI Applications** link to display the Manage Business Intelligence Applications dialog.

3. Display the BI Application Offerings tab.

4. Select the **Enabled** check box next to each Offering you want to deploy.

   Enabling Offerings enables the applicable setup seed data in Oracle BI Applications Configuration Manager.

   > **Note:** Enable only the Offerings that have been purchased so that you do not display data for other Offerings. You can enable additional Offerings as you purchase them.

5. Click **Save** to save your changes.

6. To exit this dialog, click **Done**.

3.4.17 Setup Step: Set Languages for Data Load

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Oracle BI Applications 11*g* Release 1 (11.1.1) loads data from a Fusion Applications source to Oracle Business Analytics Warehouse in multiple languages.

In Oracle BI Applications Configuration Manager, you specify the language or languages in which data is extracted from the source and loaded into Oracle Business Analytics Warehouse.

To specify Oracle Business Analytics Warehouse languages:

1. Login to Oracle BI Applications Configuration Manager as a user with the Business Intelligence Applications Administrator duty role.

   For more information about how to log in and use Oracle BI Applications Configuration Manager, see Appendix A.2, "How to Start Oracle BI Applications Configuration Manager."

2. Select the **Manage BI Applications** link to display the Manage Business Intelligence Applications dialog.

3. Display the **Warehouse Languages** tab.

4. Specify the languages from which data is to be extracted from the list of languages displayed in the table by selecting **Installed** from the Installed drop down. American English is the default installed language. All other languages are disabled.

   > **Note:** When you mark a language as Installed, the Data Load Parameter LANGUAGE_LIST is populated internally with the list of selected languages. This parameter list is used during data extraction to extract data in the selected languages.

5. Set the Base Language by selecting the **Language** record and clicking the **Set Base Language** icon in the table toolbar. By default, American English is the Base Language.

> **Note:** The base language is used to generate pseudo-translation records in Oracle Business Analytics Warehouse for any languages missing in the Source system.

6. Click **Save** to save your changes.

3.4.18 Setup Step: Editing Preferred Currency Display Names and Enable Document Currency

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Editing Preferred Currency Display Names

Oracle Business Intelligence is installed with a set of preferred currencies with pre-configured Preferred Currency Codes and Preferred Currency Names. You can use the default Currency Display Names, or you can specify new Currency Display Names as described in this section. You edit Preferred Currency Name values to change the currency labels that are displayed in all modules associated with BI dashboards. For example, you might want to change the 'Local Currency' label from 'Ledger Currency' to 'Local Currency'.

To edit Currency Display Names:

1. Login to Oracle BI Applications Configuration Manager as a user with the Business Intelligence Applications Administrator duty role.

   For more information about how to log in and use Oracle BI Applications Configuration Manager, see Section A.2, "How to Start Oracle BI Applications Configuration Manager."

2. From the **Tasks** bar, click **Manage Preferred Currencies** to display the "Manage Preferred Currencies dialog".

3. Select a currency in the Preferred Currencies list. Selecting the currency displays the associated modules in the bottom table.

4. Click the value in the **Preferred Currency Name** column (or click the Edit icon) to display the Preferred Currency Name dialog.

5. In the Preferred Currency Name field, specify a currency name. This is the name that will appear for end users in the Currency drop-down list, located in Preferences tab of the My Accounts dialog of Oracle Business Intelligence.

6. Click **Save and Close**.

Enabling Document Currency

By default, Document currency is excluded from the Currency drop-down list (located in the Preferences tab of the My Accounts dialog in Oracle Business Intelligence). To include Document currency in the Currency drop-down list, you must remove a specific filter from all of the following security groups in the BI metadata repository (RPD):

- OBIA_PROJECT_CURRENCY_PREFERENCES

- OBIA_HCM_CURRENCY_PREFERENCES

- OBIA_SCM_CURRENCY_PREFERENCES

- OBIA_FINANCIAL_CURRENCY_PREFERENCES

- OBIA_PROCUREMENT_CURRENCY_PREFERENCES

- OBIA_MARKETING_CURRENCY_PREFERENCES

- OBIA_PARTNER_CURRENCY_PREFERENCES

- OBIA_CRM_CURRENCY_PREFERENCES

- OBIA_SALES_CURRENCY_PREFERENCES

Note: The security groups listed above are displayed in Oracle BI Administration Tool on the Application Roles tab on the Identity Manager dialog. The example screenshot below shows the security group named OBIA_PROJECT_CURRENCY_ PREFERENCES selected on the Application Roles tab.
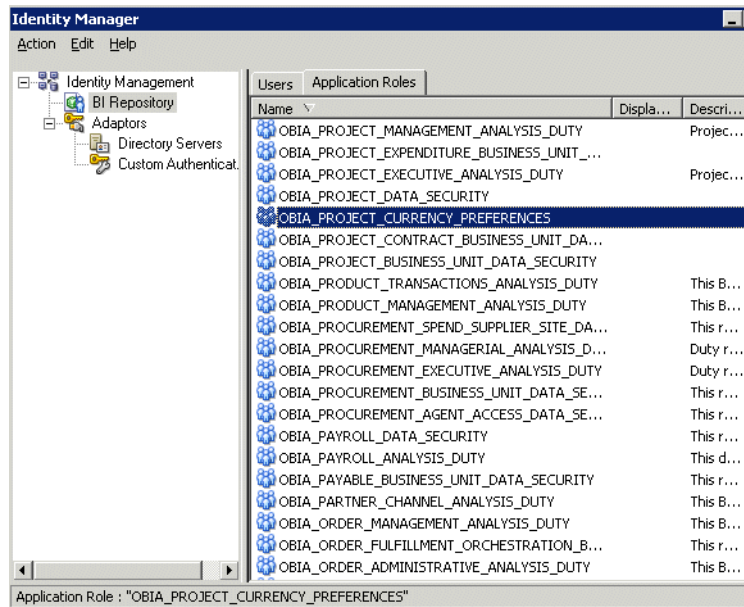
To enable document currencies:

1. In Oracle BI Administration Tool, click **Manage**, then **Identity,** to display the Identity Manager dialog.

2. In the Identity Manager dialog, click **BI Repository**.

3. Display the **Application Roles** tab.

4. For each of the security groups in the RPD list above, do the following:

   a. Double-click the security group to display the **Application Role - <Name>** dialog.

   b. Click **Permissions**, and then display the **Data Filters** tab.

   c. In the **Data Filter** column, remove the following filter:

   ```
   AND "Core"."Fact - Preference List"."Currency Preference
   Code" <> 'Document Currency'
   ```

   For example, the filter before editing:

   ```
   "Core"."Fact - Preference List"."Module Code" = 'PROJECT_
   AN' AND "Core"."Fact - Preference List"."Currency
   Preference Flag" in ('W', 'B') AND   "Core"."Fact -
   Preference List"."Currency Preference Code" <> 'Document
   Currency'
   ```

For example, the filter after editing:

```
"Core"."Fact - Preference List"."Module Code" = 'PROJECT_
AN' AND "Core"."Fact - Preference List"."Currency
Preference Flag" in ('W', 'B')
```

5. Save the changes.

   If you edited the metadata repository in Offline mode, then you must use Oracle Fusion Middleware Control to upload the RPD file.

   Tip: In Oracle Fusion Middleware Control, use the Upload BI Server Repository area on the Business Intelligence\coreapplication\Deployment\Repository tab. For detailed instructions on how to upload an RPD file, see Section 10.2 'Using Fusion Middleware Control to Upload a Repository and Set the Oracle BI Presentation Catalog Location' in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

3.4.19 Setup Step: Restore the BI Metadata Repository File for Oracle BI Applications

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

The BI metadata repository is trimmed during the BI Provisioning phase in a Fusion Applications installation to contain only RPD projects for Oracle Fusion Transactional Business Intelligence (OTBI). Oracle BI Applications requires a full (that is, untrimmed) RPD for ETL. To deploy Oracle BI Applications, you must restore the Golden RPD and merge it with updates and customizations in the Runtime RPD.

> **Note:** The trimmed BI metadata repository is referred to as the Runtime RPD, and is located in:
>
> ```
> BI_INSTANCE\bifoundation\OracleBIServerComponent\coreapplication_
> obis1\repository\OracleBIApps.rpd
> ```
>
> The full (that is, untrimmed) BI metadata repository is referred to as the Golden RPD, and is located in:
>
> ```
> BI_ORACLE_
> HOME/biapps/admin/provisioning/OracleBIServerComponent/OracleBIApps
> .rpd
> ```

To restore the BI Metadata Repository file for Oracle BI Applications:

1. Create an appropriate directory structure, as described in Section 3.4.19.2, "Create Directories and Copy In the RPD Files".

2. Open the Runtime RPD and obtain the list of Projects, as described in Section 3.4.19.3, "Open the Runtime RPD and Obtain a List of Projects".

3. Extract the Project metadata from the Golden RPD, as described in Section 3.4.19.4, "Extract the Trimmed Metadata from the Golden RPD".

4. Equalize the Runtime RPD and the Golden RPD, as described in Section 3.4.19.5, "Equalize the Runtime RPD and the Golden RPD".

5. Create a patch for the new RPD, as described in Section 3.4.19.6, "Create a Patch for the New RPD".

6. Apply the patch containing the new RPD, as described in Section 3.4.19.7, "Apply the Patch for the New RPD".

7. Upload the new RPD, as described in Section 3.4.19.8, "Upload the New RPD".

3.4.19.1 Before you start

Before you start, you need the following:

- a copy of the Runtime RPD.

- the password for the Runtime RPD (to obtain the password, go to the LDAP's Credential Store, and navigate to cn=oracle.bi.system, cn=CredentialStore, cn=FusionDomain, cn=JPSContext, cn=FAPolicies, and obtain the password).

- a copy of the Golden RPD.

- the password for the Golden RPD (the default is Admin123).

3.4.19.2 Create Directories and Copy In the RPD Files

Create the following directory structure and set up the RPD files.

1. Create the following directories:

   Work\

   Work\GoldenRPD

   Work\RuntimeRPD

   Work\ExtractRPD

   Work\Patch

   Work\RestoredRPD

   Work\EqualizeRPD

2. Copy the Golden RPD and the Runtime RPD to the corresponding folders:

   a. Copy the Golden RPD (OracleBIApps.rpd) to the \Work\GoldenRPD directory and rename it GoldenRPD.rpd.

      Note: The Golden RPD is located in the following location:

      ```
      BI_ORACLE_
      HOME/biapps/admin/provisioning/OracleBIServerComponent/OracleBIApps.rpd
      ```

   b. Copy the Runtime RPD (OracleBIApps.rpd) to the \Work\RuntimeRPD directory, and rename it RuntimeRPD.rpd.

      Note: The Runtime RPD is located in the following location:

      ```
      BI_INSTANCE\bifoundation\OracleBIServerComponent\coreapplication_
      obis1\repository
      ```

3.4.19.3 Open the Runtime RPD and Obtain a List of Projects

1. In Oracle BI Administration Tool, open the Runtime RPD in Offline mode.

   For example, open C:\Work\RuntimeRPD\RuntimeRPD.rpd.

2. From the menu, choose Manage, then Projects, to display the Projects Manager dialog.

3. Note down or copy into a file the project names in this list, using the high/low case letters and spaces exactly as displayed.

For example, "Fusion Incentive Compensation Transactional Analysis", "Fusion Marketing Transactional Analysis". You will use this list of project names in the next task.

3.4.19.4 Extract the Trimmed Metadata from the Golden RPD

1. Open a command prompt.

2. Navigate to the bin directory of the Oracle BI Administration Tool installation, and open the bi_init.bat file.

   For example, C:_CLIENT_INSTALL_DIR_ \oraclebi\orahome\bifoundation\server\bin\bi_init.bat.

3. Edit the bi_init.bat file, and copy into memory the sections where the environment variables are being set.

   For example, copy the following commands:

   ```
   set ORACLE_HOME=C:\RUP3\oraclebi\orahome
   set ORACLE_INSTANCE=C:\RUP3\oraclebi\orainst
   set ORACLE_BI_APPLICATION=coreapplication
   ```

4. In the command prompt, execute the environment commands by pasting in the commands from Step 3 and pressing Enter.

5. In the command prompt, execute the extractprojects command using the following parameters and values:

   -B *<Path>*\GoldenRPD.rpd

   -O *<Path>*\ExtractRPD.rpd

   -I "Project 1" -I "Project 2" -I "Project *<n>*"

   Note: You must specify the Project names that you obtained in Section 3.4.19.3, "Open the Runtime RPD and Obtain a List of Projects".

   -P Password of the Golden RPD.

   Note: The -I 'Project Name' command must be repeated for each project.

   For example, C:\RUP3\oraclebi\orahome\bifoundation\server\bin>extractprojects.exe -B C:\Work\GoldenRPD\GoldenRPD.rpd -O C:\Work\ExtractRPD\ExtractRPD.rpd -P Admin123 -I "Fusion Incentive Compensation Transactional Analysis" -I "Fusion Marketing Transactional Analysis" -I "Fusion Partner Transactional Analysis" -I "Fusion Sales Transactional Analysis" -I "Marketing Analytics Fusion Edition" -I "Partner Analytics Fusion Edition" -I "Sales Analytics Fusion Edition"

   The output from the extractprojects command is an RPD named ExtractRPD.rpd that will be referred to as the Extract RPD. The Extract RPD contains the same set of projects as the Runtime RPD.

3.4.19.5 Equalize the Runtime RPD and the Golden RPD

1. In a command project, execute the equalizerpd command using the following parameters and values:

   -C *<Path>*\ExtractRPD.rpd

   -B Password of the Extract RPD

   -F *<Path>*\RuntimeRPD.rpd

-E Password of the Runtime RPD

-O *<Path>*\EqualizeRPD.rpd

For example:
C:\RUP3\oraclebi\orahome\bifoundation\server\bin>equalizerpds.exe -C
C:\Work\ExtractRPD\ExtractRPD.rpd -B Admin123 -F
C:\Work\RuntimeRPD\RuntimeRPD.rpd -E welcome1 -O
C:\Work\EqualizeRPD\EqualizedRPD.rpd

The output from the equalizerpd command is an RPD named EqualizeRPD.rpd
that will be referred to as the Equalized RPD. This Equalized RPD contains the
merged metadata from the Runtime RPD and the Golden RPD.

3.4.19.6 Create a Patch for the New RPD

The patch being created is a diff between the Equalized RPD and the Extracted RPD,
this patch will contain the all the configuration and customization changes that have
been made.

1. In Oracle BI Administration Tool, open the Equalized RPD.

2. Choose File, then Compare, to display the Compare repositories dialog.

3. From the Compare repositories dialog, open the Extract RPD as the Original
   Repository.

4. When prompted, specify the Golden RPD (that is, Admin123) password to open
   the Extract RPD.

   The Oracle BI Administration Tool analyzes the RPDs and displays a list of
   differences.

5. Click Create Patch to display Create Patch dialog, and specify a patch file name
   and location (for example: C:\Work\Patch\patch.xml).

6. Click Save.

3.4.19.7 Apply the Patch for the New RPD

1. Apply the patch obtained from the previous task using the biserverxmlexec
   command with the following commands and parameters:

   -P Password of GoldenRPD

   -I Path_Name\File_Name of the Patch.xml file

   -S Password of the RuntimeRPD

   -B *<Path>*\GoldenRPD.rpd

   -O *<Path>*\RestoreRPD.rpd

   Note: RestoreRPD.rpd referred here is the name of the Restored RPD

   For example:
   C:\RUP3\oraclebi\orahome\bifoundation\server\bin>biserverxmlexec.exe -P
   Admin123 -I C:\Work\Patch\patch.xml -S welcome1 -B
   C:\Work\GoldenRPD\GoldenRPD.rpd -O
   C:\Work\RestoreRPD\RestoreRPD.rpd

3.4.19.8 Upload the New RPD

1. Use Oracle Fusion Middleware Control to upload the new RPD.

For example, if you specified the output file in the previous task as '-O C:\Work\RestoreRPD\RestoreRPD.rpd', then you upload RestoreRPD.rpd.

Tip: In Oracle Fusion Middleware Control, use the Upload BI Server Repository area on the Business Intelligence\coreapplication\Deployment\Repository tab.

For detailed instructions on how to upload an RPD file, see Section 10.2 'Using Fusion Middleware Control to Upload a Repository and Set the Oracle BI Presentation Catalog Location' in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

3.4.20 Setup Step: Run Domains ETL

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

This task loads source-specific data into to the Oracle BI Applications Configuration Manager tables. This enables Oracle BI Applications Configuration Manager to display the appropriate source-specific values as choices to Functional Developers in FSM. The data from the Fusion Applications source is loaded into the Oracle BI Applications Configuration Manager tables using ETL routines.

Work with your ETL Administrator to complete this procedure.

Note: Before you run ETL Execution Plans, make sure that you have set the environment variables as specified in Section 3.4.9.1.2, "Before Installing Informatica PowerCenter for Oracle BI Applications Environments".

To run domains ETL:

1. Start DAC Client.

   To start DAC Client, navigate to the \dac\ folder and execute the file startclient.bat.

2. At the Login dialog, select an appropriate Connection.

3. When prompted for a User Name and Password, enter a User Name with the appropriate permissions to access the Execute view in DAC Client and to run ETL.

   For more information about granting access to Oracle BI Applications Components, see Section 3.4.6, "Setup Step: Grant User Access to Oracle BI Applications Components."

4. In DAC Client, navigate to the **Execute** view, and display the **Execution Plans** tab.

5. Run the Execution Plan named **'Domains - Fusion 1.0'**.
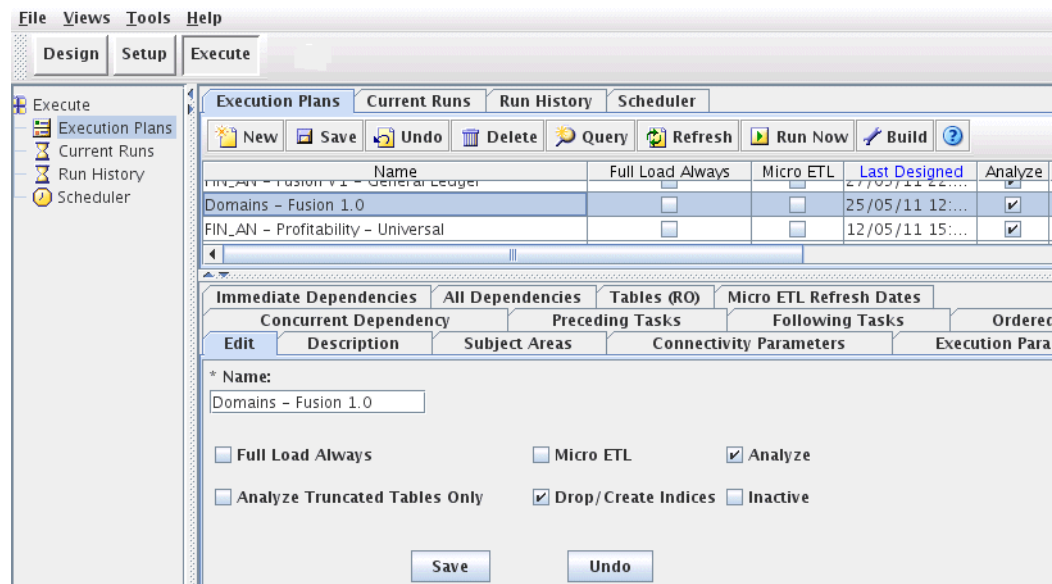
3.4.21 Setup Step: Verifying Duty Roles required for Oracle BI Applications

This task is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI Applications".

Oracle BI Applications is installed with a Role Catalog containing pre-configured Duty Roles that provide access to Oracle BI Applications dashboards and data. Each Duty Role is mapped to one or more Fusion Applications Enterprise Roles (also referred to as Job Roles). When you deploy Oracle BI Applications, you need to verify that the Duty Roles are mapped to the default Fusion Applications Enterprise Roles. When Fusion Applications is deployed, Duty Roles might have been removed from the Enterprise Role hierarchies, and therefore must be restored (or re-mapped) before you can deploy Oracle BI Applications. For details of Duty Roles that might be been removed from Role hierarchies, refer to section 'Oracle Business Intelligence Dashboards and Reports' in *Oracle Fusion Applications Release Notes.*

A list of required Enterprise Roles and Duty Roles for Oracle BI Applications is published on My Oracle Support (ID 1333454.1.) as a spreadsheet named 'Oracle Business Intelligence Applications Duty Role Assignments for Fusion Applications'.

For instructions on how to verify Duty Roles for Oracle BI Applications, see Appendix A.5, "How to Verify Duty Roles required for Oracle BI Applications."

3.4.22 About Customizing Oracle-Delivered Oracle BI Presentation Catalog Content

The Oracle BI Applications artifacts delivered by Oracle in the Oracle BI Presentation Catalog such as dashboards, analyses, prompts, and filters, are read-only and can be changed only by a user with the BI Platform Administrator role. If you want to customize analyses or dashboards, Oracle recommends that you do not directly modify the Oracle-delivered content. Instead, copy the content you want to change and modify only the copied content.

Note: The BI Platform Administrator role is available only to On-Premise and On-Demand customers. The role is not available to Oracle Cloud customers.

For information about changing permissions for dashboards and analyses, see the chapter about managing objects in Oracle BI Presentation Catalog in *Oracle Fusion*

Middleware User's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition) for the Oracle BI Enterprise Edition release that you are using.

3.4.23 Next Steps

After you complete the setup tasks described above, Oracle BI Applications is ready for functional configuration, which is accomplished using Oracle BI Applications Configuration Manager and Functional Setup Manager. To enable functional configuration, you must provide the Oracle BI Applications Configuration Manager URL to your Implementation Managers and Functional Developers. To learn about the functional configuration process for Oracle BI Applications, see Part II, "Functional Configuration for Oracle Business Intelligence Applications".

After functional configuration is complete, your environment is ready for ETL and data-load into Oracle Business Analytics Warehouse. Data Load is performed using DAC and Informatica PowerCenter. Furthermore, after Oracle BI Applications analyses and dashboards have been tested and the data validated, the Oracle BI Applications System Administrator will need to inform the Fusion Applications Administrator to enable the appropriate embedded analyses and dashboard (ADR) regions in Fusion Applications. The ADR regions are enabled using Functional Setup Manager for Fusion Applications.

Part II

Functional Configuration for Oracle Business Intelligence Applications

Part II covers functional configuration of Offerings for Oracle BI Applications. It contains the following sections:

- Chapter 4, "Overview of Functional Configuration in Oracle BI Applications"
- Chapter 5, "Performing Functional Configuration"
- Chapter 6, "Administering and Maintaining Functional Configuration Data"

4

Overview of Functional Configuration in Oracle BI Applications

This section includes an overview of how to configure Oracle Business Intelligence Applications.

This chapter contains the following topics:

- Section 4.1, "Terminology"
- Section 4.2, "Overview of Oracle BI Applications Configuration Manager and Functional Setup Manager"
- Section 4.3, "Getting Started With Oracle BI Applications Configuration Manager"
- Section 4.4, "About Users and Roles in Oracle BI Applications Configuration Manager"
- Section 4.5, "About Setup Objects in Oracle BI Applications Configuration Manager"
- Section 4.6, "About Functional Setup Tasks in FSM"
- Section 4.7, "About the Main Task Areas in Oracle BI Applications Configuration Manager"
- Section 4.8, "About the Oracle BI Applications Configuration Manager Work Area"
- Section 4.9, "About Accessibility Features In Oracle BI Applications Configuration Manager"

4.1 Terminology

This section lists terminology that relates to configuration in Oracle Business Intelligence Applications.

- Source Instance - The name given by the Oracle BI Applications Administrator to the Fusion Applications transactional system that serves as the source of data for Oracle Business Analytics Warehouse. A Source Instance is registered in the System Setups\Define BI Applications Instance\Source Systems dialog in Oracle BI Applications Configuration Manager.

- BI Application Offering - A BI Application product you have purchased. For example, Oracle Financial Analytics, or Oracle Sales Analytics. An Oracle BI Applications Offering can have one or more Modules and Functional Areas.

- Functional Area - A component part of an Oracle BI Applications Offering. For example, Workforce Effectiveness, Leave & Accrual, Workforce Development are

Functional Areas in Oracle Human Resources Analytics. A Functional Area is the smallest unit of an Oracle BI Applications Offering that can be implemented.

- BI Application Module - A component part of a Functional Area. One or more BI Application Modules may comprise an Oracle BI Applications Offering. For example, the Oracle Financial Analytics Fusion Edition Offering consists of the Financial Analytics module and the Employee Expense functional area from Procurement and Spend Analytics module.

4.2 Overview of Oracle BI Applications Configuration Manager and Functional Setup Manager

Functional configuration for Oracle Business Intelligence Applications is done using the following tools:

- Oracle BI Applications Configuration Manager:
  - Contains the setup objects for Oracle BI Applications.
  - Provides administrative GUIs for setup and configuration.
  - Recommended for on going administration and maintenance of functional setups.
  - Quick review of the setup values and for troubleshooting.

  For more detailed information about Oracle BI Applications Configuration Manager, see Section 4.2.1, "What is Oracle BI Applications Configuration Manager?" .

- Functional Setup Manager (FSM) for Oracle BI Applications
  - Oracle BI Applications Configuration Manager leverages FSM to provide Task based implementation, with guided set of sequenced tasks and dependency management.
  - Provides phased implementations capability.
  - FSM for Oracle BI Applications is different from the Oracle Fusion Functional Setup Manager:

    - FSM for Oracle BI Applications includes only the features applicable for Oracle BI Applications functional setup.

    - FSM for Oracle BI Applications has lighter Tech stack dependencies.
  - FSM is launched from Oracle BI Applications Configuration Manager to perform functional configurations.
  - FSM is recommended for initial implementation.
  - FSM invokes Oracle BI Applications Configuration Manager UI for setup tasks.

  For more detailed information about FSM for Oracle BI Applications, see Section 4.2.2, "What is Functional Setup Manager?" .

4.2.1 What is Oracle BI Applications Configuration Manager?

Oracle BI Applications Configuration Manager is a Web application for setting up and maintaining an Oracle Business Intelligence Applications environment.

Figure 4–1   Oracle BI Applications Configuration Manager Main Page
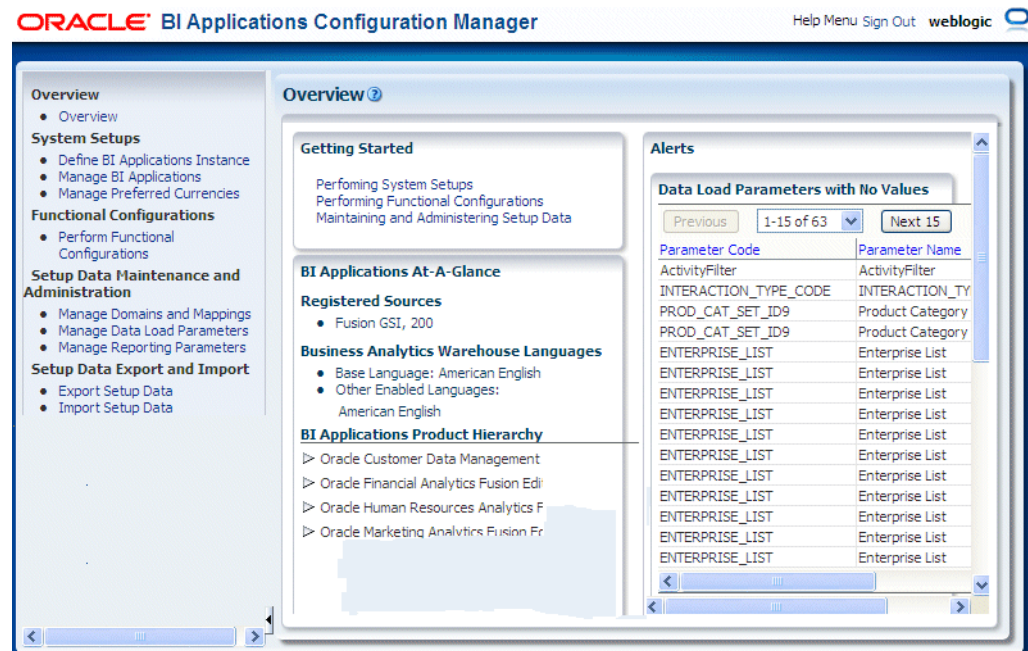
For functional configuration, Oracle BI Applications Configuration Manager works in conjunction with Functional Setup Manager to provide guided Tasks to configure Offerings and Functional Areas. For more information about using FSM in conjunction with Oracle BI Applications Configuration Manager, see Section 5.1, "What is Functional Configuration?".

Oracle BI Applications Configuration Manager enables you to do the following:

- Launch Functional Setup Manager to configure Offerings and Functional Areas (for example, by specifying Domain Mappings, Data Load Parameters, and Reporting Parameters).

  Functional Setup Manager provides a list of guided Tasks that enable you to configure Oracle BI Applications Modules and Functional Areas. For example, a Functional Developer might use a Task named 'Configure Initial Extract Date' to set the value of INITIAL_EXTRACT_DATE to '2005-01-01'.

  For more information, see Chapter 5, "Performing Functional Configuration".

- Monitor and manage setup data, and extend Oracle Business Analytics Warehouse where necessary.

  For example, a Functional Developer might have set the value of INITIAL_EXTRACT_DATE to '2005-01-01'. If you want to change this value to '2006-01-01', then you might use the Manage Data Load Parameters tab in Oracle BI Applications Configuration Manager to edit the value of this parameter.

  For more information, see Section 6.2, "Roadmap for Setup Data Maintenance and Administration".

- Migrate configuration data across environments, using the Import and Export options.

  For more information, see Section 6.13, "About Exporting and Importing Setup Data for Oracle BI Applications Configuration Manager".

4.2.2 What is Functional Setup Manager?

Functional Setup Manager (FSM) enables you to manage and perform functional configuration tasks for configuring Offerings.
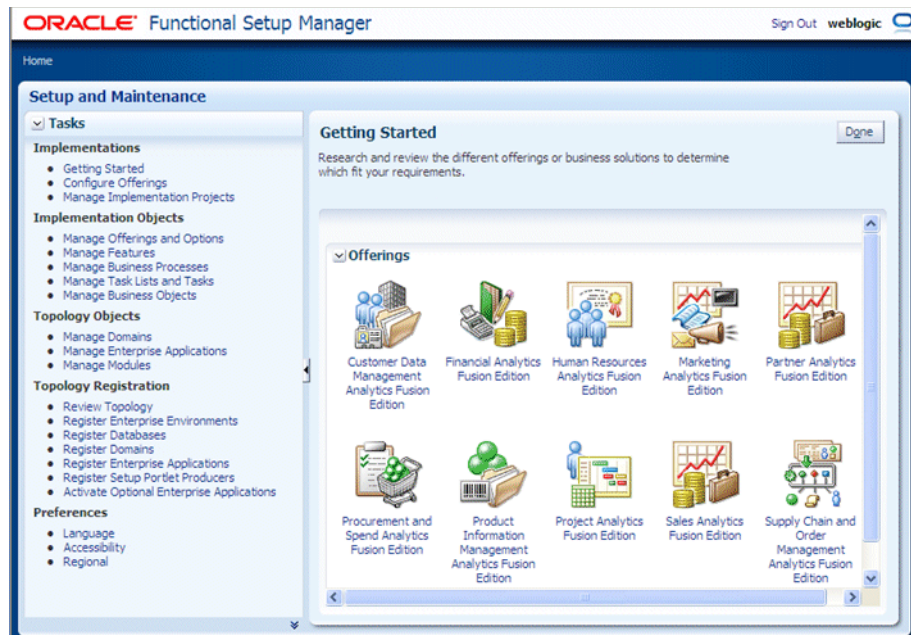
FSM is installed and deployed as part of Oracle BI Applications. In FSM, you select the Oracle BI Applications Offering and Functional Areas that you wish to deploy. FSM generates a list of configuration tasks specific to the Offering and Functional Area(s) that were selected. These tasks can be assigned to different functional developers and the status of the Oracle BI Applications implementation project can be monitored in FSM. Setup UIs guide functional developers through the performance of each task.

In Oracle BI Applications Configuration Manager, clicking on the **Perform Functional Configurations** link launches FSM.

Figure 4–2   Performing Functional Configuration
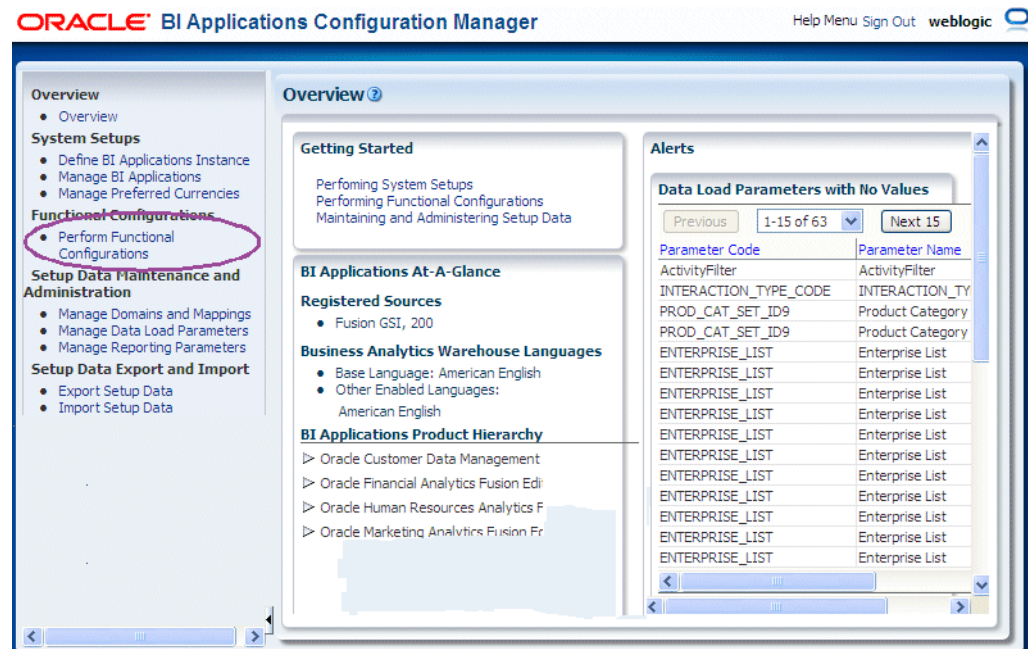
When you launch FSM as a user with the BI Applications Administrator role or the BI Applications Implementation Manager role and open an Implementation Project, you see the Functional Tasks that relate to the Offering being deployed by that Implementation Project.

The example screen shot below shows the Functional Tasks for Oracle Financial Analytics.
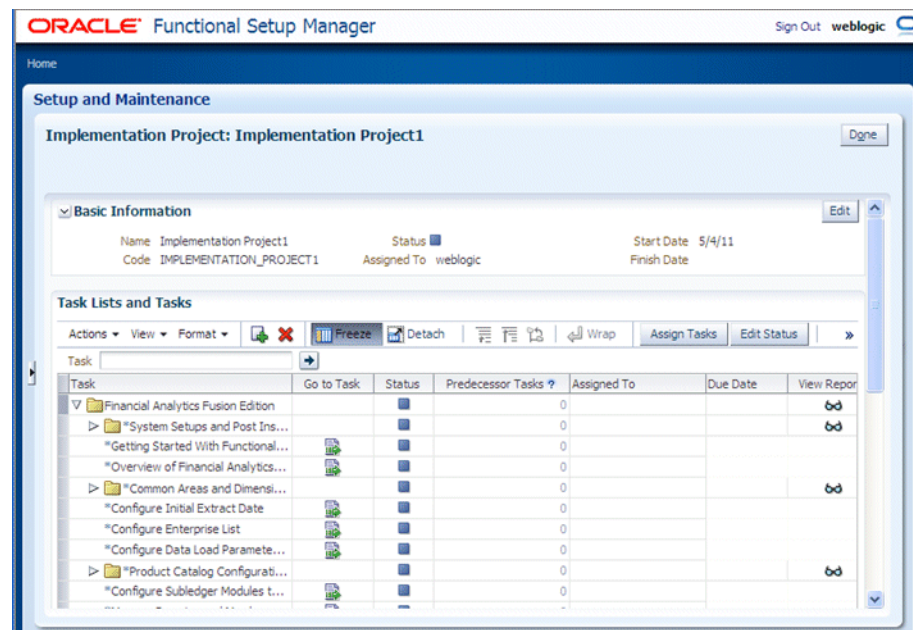
4.3 Getting Started With Oracle BI Applications Configuration Manager

This section explains how to log into Oracle BI Applications Configuration Manager and how to access the User Assistance, Help, Documentation, and other resources that are available.

4.3.1 About Getting Help

Oracle BI Applications Configuration Manager provides the following Help and User Assistance features:

- Context-sensitive Help - Click the Help icon on any dialog to view context-sensitive Help for that dialog. When the context-sensitive Help is displayed, use the Table of Contents, Index, and Search facilities to locate User Assistance on other areas of the product.

  Tip: To maximize the text pane in the Help window, choose **Maximize Reading Pane** from the View menu. Alternatively, use the Collapse Pane arrow to hide the left hand tabs (see the following screenshot), or use the vertical sizing bar (above the Collapse Pane arrow) to reduce the width of the tabs.
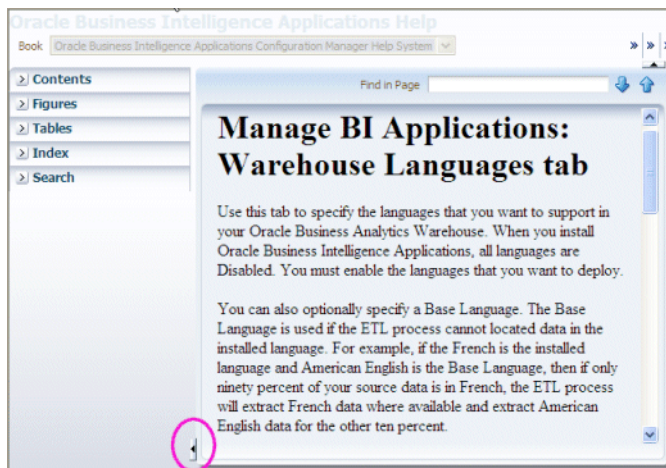
- Inline Help, such as Mouse-over Tool tips - Hold the cursor over an object to view a tool-tip for that object.

4.3.2 About Setup Data Migration in Oracle BI Applications Configuration Manager

Use the Oracle BI Applications Configuration Manager Import and Export facilities to migrate your setup data across environments (for more information, Section 6.13, "About Exporting and Importing Setup Data for Oracle BI Applications Configuration Manager").

4.4 About Users and Roles in Oracle BI Applications Configuration Manager

To access Oracle BI Applications Configuration Manager or Functional Setup Manager, you must log in as a user with one of the following Duty Roles:

- BI Applications Administrator

  Users with the BI Applications Administrator Duty Role have access to all Oracle BI Applications Configuration Manager UIs and all FSM UIs. For Oracle BI

Applications Configuration Manager, only users with this duty role can perform System Setups.

- BI Applications Developer

  Users with the BI Applications Functional Developer Duty Role have access to Oracle BI Applications Configuration Manager UIs except for the System Setup screens. In FSM, these users have access to the list of functional setup tasks assigned to them, and can execute setup tasks (that is, using the **Go to Task** option in FSM).

- BI Applications Implementation Manager

  Users with the BI Applications Implementation Manager Duty Role have access to the Oracle BI Applications Configuration Manager Overview page and dialogs for the Export and Import of Setup Data. In FSM, these users have access to Configure Offerings and Manage Implementation Projects UIs, but they cannot execute setup tasks (that is, they do not have access to the **Go to Task** option in FSM).

Setting up users with these Duty Role privileges is performed when Oracle BI Applications is set up.

To find out which user IDs have which Duty Role privileges, use your LDAP tools and Oracle APM to determine which users have the Duty Role privileges BIA_ADMINISTRATOR_DUTY, BIA_IMPLEMENTATION_MANAGER_DUTY, and BIA_FUNCTIONAL_DEVELOPER_DUTY. For more detailed information about security privileges required for Oracle BI Applications components, see Section 2.1.10 'Security Overview of Oracle BI Applications Configuration Manager and Functional Setup Manager' in *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*.

4.5 About Setup Objects in Oracle BI Applications Configuration Manager

Oracle BI Applications Configuration Manager data includes the following Setup objects:

- Data Load Parameters - used to control the nature of data loaded from the transactional system into the Business Analytics Warehouse. Examples: INITIAL_EXTRACT_DATE, TIME_GRAIN.

- Domain Mappings and Member Mappings - used to map a Source column List of Values to a Warehouse column List of Values. Example: Domain Maps for the Employee Dimension: COUNTRY -> W_COUNTRY.

- Reporting Parameters - correspond to Dynamic Repository Variables in the RPD and allow these variables to be set using Oracle BI Applications Configuration Manager. Example: FSCM_MASTER_ORG.

4.6 About Functional Setup Tasks in FSM

There are four types of Functional Setup Tasks in FSM:

- Tasks to configure Data Load Parameters. For example, Example: task Configure Initial Extract Date displays Data Load Parameter INITIAL_EXTRACT_DATE.

- Tasks to manage Domains and Mappings. Example: task Manage Domains and Member Mappings for Employee Dimension displays Domain Maps for the Employee Dimension.

- Tasks to configure Reporting Parameters - Example, task 'Configure Reporting Parameters for Master Organization' displays Reporting Parameter FSCM_MASTER_ORG.

- Tasks that provide information. These tasks might be:

  - A set of instructions for configurations to be performed in Informatica, DAC, RPD, or elsewhere 'outside' of Oracle BI Applications Configuration Manager. These Tasks are named 'How To….'. Example, task 'How to Configure Group Account Numbers' provides instructions to complete this setup.

  - Conceptual, overview or supporting information. Example, task 'Overview of Financial Analytics Fusion Edition' provides information on the Financial Analytics offering.

4.7 About the Main Task Areas in Oracle BI Applications Configuration Manager

Oracle BI Applications Configuration Manager has the following main task areas:

- System Setups - use this area to set up Source Systems, Target Databases, Warehouse Languages and so on.

- Functional Configuration - use the **Perform Functional Configurations** link to perform Functional Configuration using Oracle Functional Setup Manager.

- Setup Data Maintenance and Administration - use this area to monitor the configuration settings that your Functional Developers have made using Oracle Functional Setup Manager, and make changes where required. For example, a Functional Developer might have used a Task in FSM to set the value of INITIAL_EXTRACT_DATE to '2005-01-01'. If you want to change this value to '2007-01-01', then you might use the Manage Data Load Parameters tab in Oracle BI Applications Configuration Manager to edit the value directly, instead of reassigning the Task in FSM.

- Export and Import Configuration - use this area to backup your setup data and migrate your setup data to a separate environment.

4.8 About the Oracle BI Applications Configuration Manager Work Area

This section explains the work area in Oracle BI Applications Configuration Manager.

4.8.1 About the Work Area

The figure below (Figure 4–3) shows the main Tasks bar and the Functional Configuration work area for Domains and Mappings.

> **Tip:** To set accessibility preferences, click the Accessibility link in the top right hand corner to display the Enable Accessibility Preferences dialog. For example, you might want to display high contrast colors, or use large fonts.

Figure 4–3   Oracle BI Applications Configuration Manager Working Area
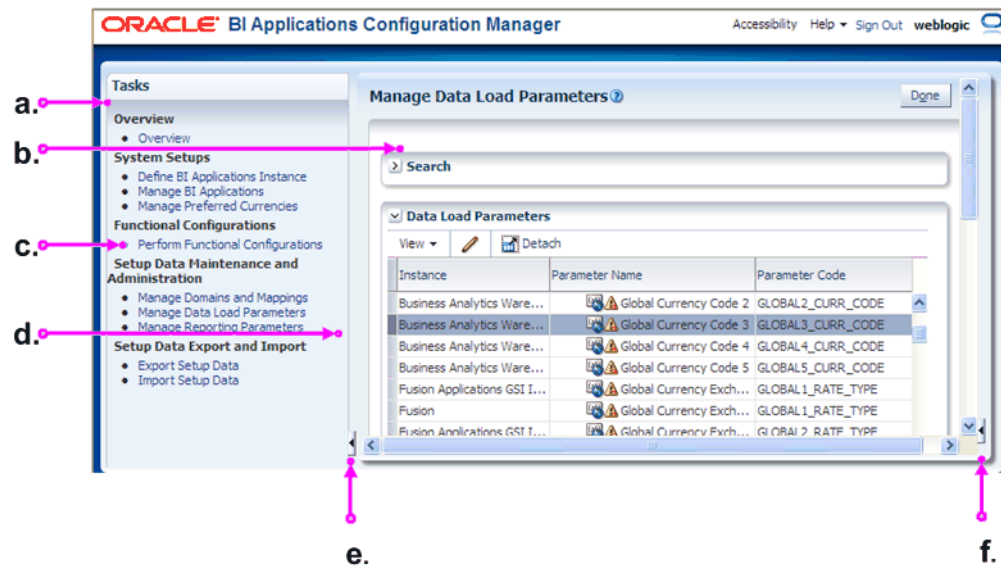
Key to figure:

a. Tasks bar, which provides links to Oracle BI Applications Configuration Manager options.

b. Work Panel, which displays the currently selected option.

c. The Perform Functional Configurations option starts FSM.

d. Resize bar for the Tasks Pane.

e. Collapse Tasks Pane. **Tip**: Use the Collapse Task Pane arrow to hide the Tasks bar and maximize the screen area for displaying the Setup Data pages.

f. Expand/Collapse Contextual Pane arrow (for Data Load Parameters and Reporting Parameters only).

Note: Some pages (for example, the Manage Data Load Parameters page) have an additional Contextual Pane at the right hand side that can be expanded (and resized), or collapsed. The following screenshot shows the Expand/Collapse Pane arrow in the bottom right hand corder of the work area.

Figure 4–4   Data Load Parameters tab with Contextual Pane
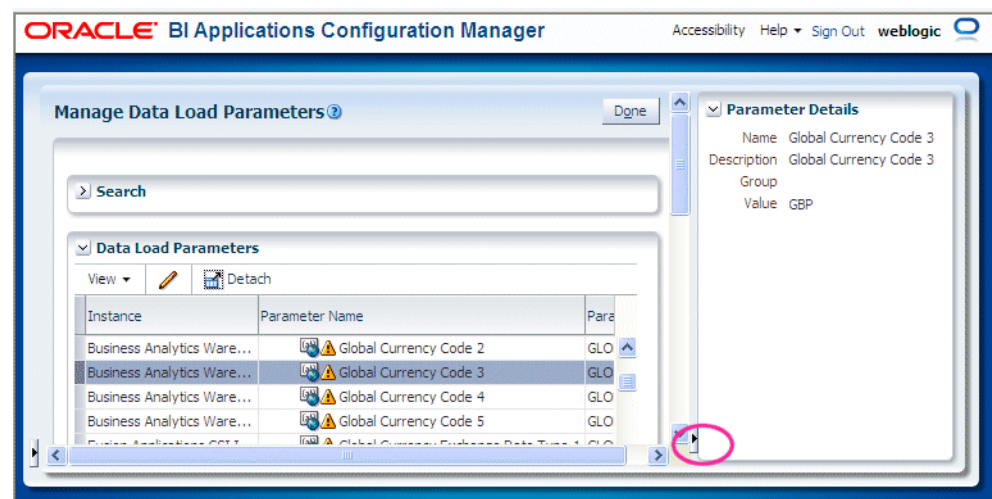
4.8.2 About the Menu Options

Oracle BI Applications Configuration Manager uses the following menus and options:

Table 4–1   Menu Options in Oracle BI Applications Configuration Manager

Icon or Menu Option	Description
Actions ▾	**Actions** Use the Actions menu to display a list of available options for the currently selected object. For example, common actions are Enable, Disable, Add, and Edit.
Detach	**Detach** Use the Detach icon to view a table tab in a separate larger pane. For example, use this option to view data in wide columns more easily. To re-attach a separated pane, click Detach again. **Note**: If you are editing a master-detail table, the Detach option does not detach the master-detail relationship.
View ▾	**View** Use the View menu to select which columns to view, and re-order columns.

4.8.3 About the Icons

Oracle BI Applications Configuration Manager uses the following icons:

*Table 4–2   Icons and Menu Options in Oracle BI Applications Configuration Manager*

Icon or Menu Option	Description
➕	**Add** Use the Add icon (or + symbol) to add an object. For example, click the Add icon on the Define BI Applications Instance - Source Systems tab to add a new Source System.
⚠	**Set parameter value before full data load** This icon indicates that the value of a Data Load Parameter must be set before you perform a full load of data from the source instance to Oracle Business Analytics Warehouse.
✎	**Edit** Use the Edit icon (or pencil symbol) to edit the object that is currently selected in the table. For example, click the Edit icon on the Define BI Applications Instance - Source Systems tab to edit the currently selected Source System details. Alternatively, click on the value in the Parameter Value column to edit the value.
🌐	**Global Parameter** This icon indicates that the parameter is global (that is, it applies to all (or a majority of) ETL tasks).

Table 4–2   (Cont.) Icons and Menu Options in Oracle BI Applications Configuration

Icon or Menu Option	Description
	Go To Top Use this icon to display the all nodes in the domain hierarchy.
	Go Up Use this icon to display the next highest level of nodes in the domain hierarchy.
	Overridable Parameter This icon indicates an overridable Application Specific parameter. An Overridable parameter icon indicates that each associated Fact Group or Dimension Group can have a different value. For more information, see Section 6.5.1, "Key points about working with Data Load Parameters".
	Parameter This icon denotes a parameter. For Data Load Parameters, if displayed in the **Data Load Parameters** pane (or top list), then this icon indicates an Application Specific parameter. If displayed in the **Group Specific Parameter Values For** pane (or lower list), then this icon indicates a Non-Overridable Application Specific parameter, which means that each associated Fact Group or Dimension Group must have the same value. In other words, if you change the value for one Fact Group or Dimension Group, then all Fact Groups and Dimension Groups are updated with that value (if you first confirm at a Warning dialog).
	Show as Top Use this icon to display only the currently selected parent node and child nodes. **Note**: This icon is only active for child nodes.
	Read Only Parameter This icon indicates that a Data Load Parameter parameter value is read-only, which means that you cannot change that parameter value on the Manage Data Load Parameter dialog. For example, Base Language, which you set using the "Manage BI Applications: Warehouse Languages tab" in Oracle BI Applications Configuration Manager (this option is not available in FSM).

Table 4–2   (Cont.) Icons and Menu Options in Oracle BI Applications Configuration

Icon or Menu Option	Description
	Parameter Category This icon denotes a grouping of related parameters, for example, the Configure Time Dimension category is a group of parameters that are related to calendars. To expand a Parameter Category, click Expand: To collapse a Parameter Category, click Collapse:
	Query By Example Use the Query By Example icon to toggle the display of the Query By Example fields above each display column. When Query By Example fields are displayed, enter a value into a Query By Example field and press [Enter] to search for records that match that value. To clear a query, clear the text in the Query By Example field and press [Enter] again. For example, to search for parameters with names beginning with Global, type Global in the query box above a **Name** column, then press [Enter]. To clear the query, clear the text 'Global' from the query box above the **Name** column, then press [Enter]. For more information about how to use Query By Example, see Section 4.8.5, "How to use Query By Example".
	Select Date Use the Select Date icon to toggle the Calendar pane, which enables you to select a date. For example, on the Export Setup Data tab, select January 1 2010 to display only files that were exported on that day.
	Start Search Use the Start Search icon to display parameters matching the value or wildcard specified in the adjacent text box.

Tip: To see a tool tip for an icon, mouse-over the icon. In the following screenshot, mousing-over the Edit icon displays the tool tip 'Edit Data Source Information'.
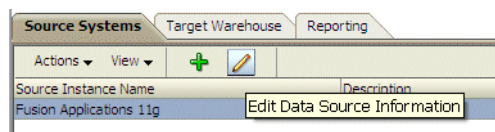
4.8.4 About sorting items

To change the sort order of displayed items, click on or mouse-over the field name on which you want to sort, then click the up arrow or down arrow that is displayed in the field label. Sort options are only displayed on mouse-over or when a field label or column is selected. For example, to display domain names in descending alphabetical order, you might click on the **Name** field label and click the down (or Sort Descending) arrow.

The following screenshot shows the sort options the **Name** field on the Domain Mappings dialog.

Figure 4–5   Name field with sorting options displayed
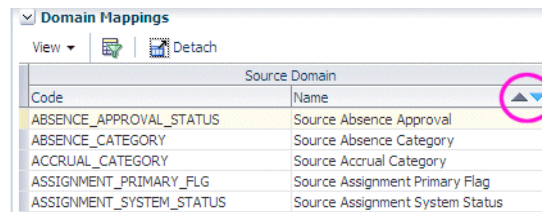
4.8.5 How to use Query By Example

Query By Example enables you to locate parameter values using a free-text search.

To use Query By Example:

1. If the Query By Example fields are not displayed, click the Query By Example icon (see the following screenshot).

   For information about menus, see Section 4.8.3, "About the Icons"

2. Enter a value into each Query By Example field on which you want to search.

   For example, to search for domains with a code beginning with 'AP', type AP into the box above the **Code** column (see the following screenshot).
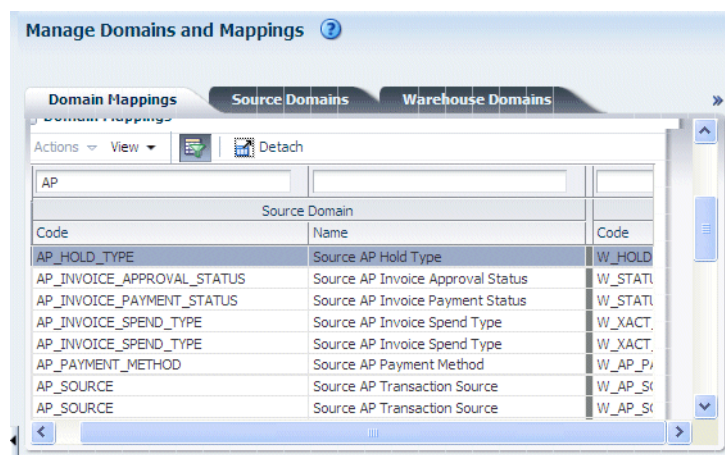
3. Press [Enter] to start the query.

4. To clear a query, clear the text in the Query By Example field and press [Enter] again.

4.9 About Accessibility Features In Oracle BI Applications Configuration Manager

Oracle BI Applications Configuration Manager provides the following accessibility features:

- Screen Reader - this feature provides accessibility-specific constructs in the User Interface to improve the user-experience with a screen reader. For example, menu items are labelled with "Menu Item", and tables and list items are provided with a radio button to enable navigation and selection using the Tab and Spacebar keys.

- High contrast colors - this feature provides high-contrast-friendly visual content, in conjunction with the high-contrast mode in the browser and in the operating system.

- Large fonts - this feature provides browser-zoom-friendly content in large text.

If accessibility features are not required, you should disable the above features.

How to enable or disable Accessibility features:

1. Select the **Accessibility** link in the top right hand corner of the screen to display the Enable Accessibility Preferences page
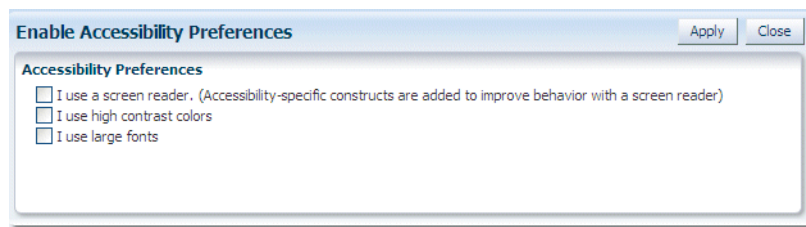
2. Use the check boxes to select or de-select the features.

3. Click Apply.

5

Performing Functional Configuration

This section explains how to perform Functional Configuration for Oracle Business Intelligence Applications

This chapter contains the following topics:

- Section 5.1, "What is Functional Configuration?"
- Section 5.2, "Roadmap for Functional Configuration"
- Section 5.3, "Performing Functional Configuration"

5.1 What is Functional Configuration?

Accurate and successful movement of data from source database to target Business Analytics Warehouse requires several functional setups of BI Applications Offerings to be performed. These functional setups, based on either business requirements or on transactional source system settings or configurations, direct the manner in which relevant data is moved and transformed from source database to target database. Additionally, some functional setups of BI Applications control the manner in which data is displayed. Functional setups are also called functional configurations.

About Starting ETL After Functional Configuration

After all Tasks have been completed in FSM, use the **ETL and Additional Information for <Offering>** Informational Task to determine the Execution Plan details that you must specify to perform ETL for that Offering. The ETL and Additional Information for *<Offering>* Informational Task specifies Subject Areas that you must include in the Execution Plan, and other ETL requirements.

5.1.1 About Task Lists and Tasks for Oracle Business Intelligence Applications Offerings

When you navigate to an Implementation Project in FSM, you see the Tasks related to the Offering specified for that Implementation Project. The example screen shot below shows the Functional Tasks for Oracle Financial Analytics.
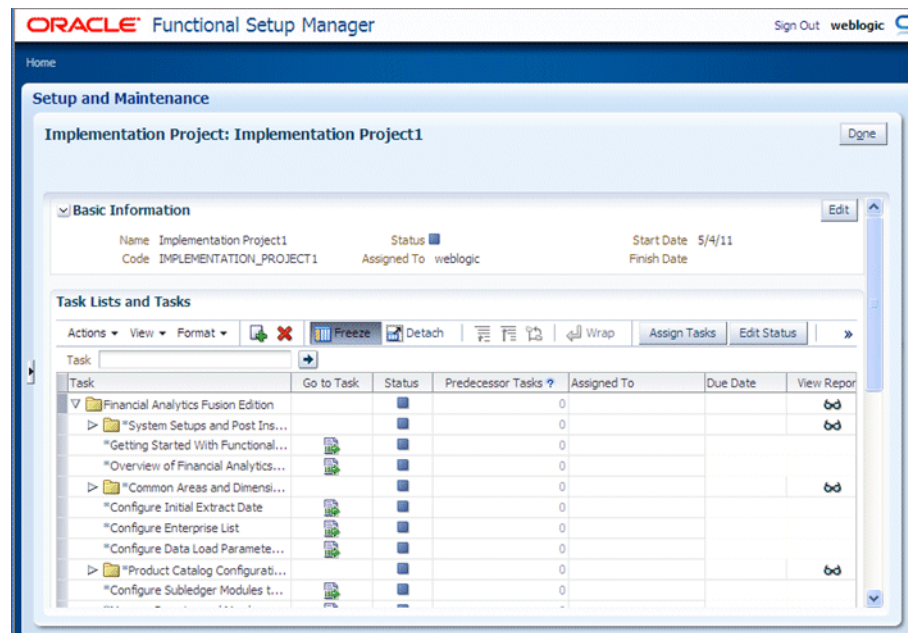
For more information about types of Functional Task, see Section 4.6, "About Functional Setup Tasks in FSM".

The following Task Lists and Tasks are available for Oracle Business Intelligence Applications:

- Getting Started with *<Offering>* - Each Offering has this Informational Task, which provides an overview of that Offering, and any other information that you need to begin configuration.

- ETL and Additional Information for *<Offering>* - Each Offering has this Informational Task, which lists Execution Plans, Subject Areas, and other useful information about configuring the Offering.

- System Setup and Post-Install Setup - This task must be completed once per Oracle BI Applications deployment by the System Administrator before Functional Configuration can begin on any Offering, and includes all mandatory post-installation steps for Oracle BI Applications.

- Common Areas and Dimensions - this Task List includes Tasks that are common to multiple Offerings. For more information about completing common Tasks, see Section 5.3.3.7, "About Completing Tasks in the Common Areas and Dimensions Configurations Task List".

- *<Offering specific Tasks>* - for a full list of functional Tasks for each Offering, see Section B.1, "Lists of Functional Configuration Tasks For Each Offering".

5.1.2 About Administrator Passwords

When you launch FSM from Oracle BI Applications Configuration Manager, you log into FSM using the same username and password that you used to log into Oracle BI Applications Configuration Manager.

5.2 Roadmap for Functional Configuration

The list below provides a high-level roadmap for functionally configuring Oracle Business Intelligence Applications.

High-Level Functional Configuration Roadmap

> **Note:** To launch FSM, click the **Perform Functional Configurations** link in Oracle BI Applications Configuration Manager. If Single Sign-On has not been configured between Oracle BI Applications Configuration Manager and FSM, you will be required to log into FSM. You use the same login credentials for FSM as you did for Oracle BI Applications Configuration Manager.

1. Configure the Offerings and Modules to deploy, and select Feature Choices. For example, you might deploy Oracle Financial Analytics, with Functional Areas: Accounts Payable, Accounts Receivable, and General Ledger.

    For more information, see Section 5.3.1, "How to Enable Offerings and Select Feature Choices".

2. Create an Implementation Project and select an Offering and one or more Modules. For example, you might create an Implementation Project to configure Oracle Financial Analytics, with Functional Areas: Accounts Payable, Accounts Receivable, and General Ledger.

    For more information, see Section 5.3.2, "How to Create an Implementation Project and Select Offerings".

    Important Note: When you create an Implementation Project, FSM generates the Tasks required to configure the specified Offerings. By default, the Tasks are assigned to the BI Administrator user. If required, you can optionally assign Tasks to Functional Developers, who will then perform the Tasks.

3. (Optional) Assign the Tasks for the specified Offerings and BI modules to one or more Functional Developers. For example, you might assign Accounts Payable tasks to user Fred, Accounts Receivable tasks to user Jill, and General Ledger tasks to user Mike.

    Alternatively, the Tasks can be performed by the default BI Administrator user.

    For more information, see Section 5.3.3.2, "How to Assign Tasks to Functional Developers".

4. Complete the functional configuration Tasks by clicking the **Go to Task** link.

    To access the 'Go To Task' option in FSM, you must be logged into FSM as a user with either BI Applications Functional Developer Duty privileges or BI Applications Administrator Duty privileges.

    For example, user Fred performs the Tasks related to General Ledger.

    For more information, see Section 5.3.3.4, "How to Perform Functional Tasks using the Functional Developer role" and Section 5.3.3.3, "How to Perform Functional Tasks using the Administrator role".

5. Monitor the progress of the Implementation Project to check that the tasks have been completed. For example, you might use the Alerts pane on the Overview page in Oracle BI Applications Configuration Manager to check for unmapped domains values, or Data Load Parameters with no values specified. In addition, Functional Setup Manager provides charts for monitoring the progress of Implementation Projects.

    For more information, see Section 5.3.3.5, "How to Monitor Implementation Projects".

6. If required, use Oracle BI Applications Configuration Manager to make changes to the setup data. For example, the Implementation Manager might use the Domains and Mappings page in Oracle BI Applications Configuration Manager to add a Domain, or to resolve unmapped domain values.

For more information about making configuration changes in Oracle BI Applications Configuration Manager, see Section 6.9, "How to Locate Unmapped Domain Values in Oracle BI Applications Configuration Manager".

5.3 Performing Functional Configuration

This section explains how to functionally configure Oracle BI Applications Offerings.

5.3.1 How to Enable Offerings and Select Feature Choices

At the start of a deployment project, you need to enable your Offerings and Functional Areas for implementation, and select Feature Choices, which typically enable you to integrate Offerings.

To enable Offerings and Functional Areas for implementation:

1. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link to launch Functional Setup Manager.

2. Select the **Configure Offerings** link in the Tasks bar, to display the Configure Offerings page.

3. For each Offering and Functional Area that you want to deploy, do the following:

a. Select the **Enable for Implementation** check box next to the Offering and each Functional Area within that Offering that you want to deploy.

Note: If the value in the Provisioned column for the selected Offering is No, then you will see the following warning:

```
Warning: This offering is not provisioned. Offering implementations cannot
be completed until the offering is provisioned. Do you want to continue?
```

If you want to continue, you must click Yes.

b. Click the Edit icon in the **Select Feature Choices** column to display the Select Feature Choices page.

Note: Only the following Offerings have feature choices: Financial Analytics, Procurement and Spend Analytics, Project Analytics, and Supply Chain and Order Management Analytics.

   c. Select the check box for each Module that you want to deploy within that Offering.

   The following screenshot shows the Select Feature Choices dialog for Financial Analytics.
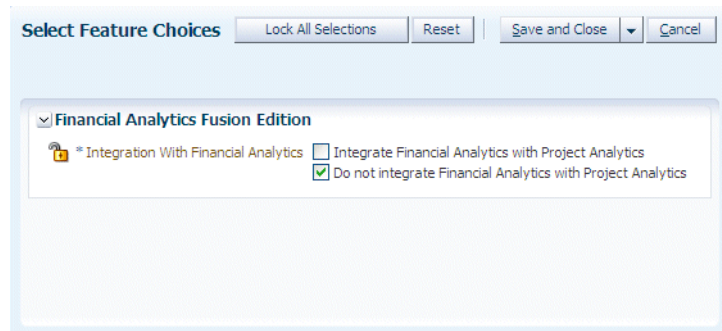
   Notes: If you want to integrate Offerings, then select the appropriate 'Integrate *<Offering>* with *<Offering>*' check box, and clear the 'Do not integrate' check box. If you do not want to integrate Offerings, then select the 'Do not integrate' check box, and make sure that the other 'Integrate *<Offering>* with *<Offering>*' check boxes are cleared.

   If you erroneously select options to both integrate and not integrate two Offerings, the integration option overrides the non-integration option.

   d. Select any other options that are specific to the Offerings that you have selected.

   For example, if you have selected the Financials Offering, you must use the **Subledger Accounting Rules** field to specify the appropriate accounting method.

4. Save your changes.

Notes on Selecting Offerings and Feature Choices

■ If you do not enable an Offering for implementation, you will not be able to configure that Offering using FSM.

5.3.2 How to Create an Implementation Project and Select Offerings

You use Functional Setup Manager to create an Implementation Project to configure an Offering and the Modules that you want to deploy. For example, if you have installed Oracle Fusion Applications HCM, you might create an Implementation Project to configure the ETL for Oracle Fusion Applications HCM.

To configure ETL for Oracle Fusion Applications, you must create at least one Implementation Project. When you create an Implantation Project, you select the Offering to deploy as part of that project.

When you create an Implementation Project, FSM generates the Tasks required to configure the specified Offerings. By default, the Tasks are assigned to the BI Administrator user. If required, you can optionally assign Tasks to Functional Developers, who will then perform the Tasks.

To create an Implementation Project:

1. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link to launch Functional Setup Manager.

2. Display the Manage Implementation Projects page, choose Actions, then Create, to display the Enter Basic Information page.

3. Use the Enter Basic Information page to specify the project details.

   Note: A default Name, Code, and Description are created for you. You can change these values if required. If you change the **Code** value, you must specify a unique code.

   > **Tip:** Use the **Name** field to specify a meaningful project name that includes the Offerings being deployed. After selecting an Offering and Functional Area(s) for implementation as part of the IP in the next step, there is no way to go back and see which Offerings and Functional Area(s) had been selected.

4. Click Next to display the Select Offerings to Implement page.

   > **Note:** Do not click **Save and Open Project**. Choosing **Save and Open Project** will create an incomplete Implementation Project, for which you cannot later specify an Offering and Functional Areas.
   >
   > You must click **Next** to specify an Offering on the next page.

5. Use the Select Offerings to Implement page to specify the Offering and the Functional Areas to include in the project.

   For example, if you are configuring Oracle Procurement and Spend Analytics with Payables and Sourcing, you select the Include check boxes next to Procurement, Payables, and Sourcing.

   Note: To make Offerings easier to manage, Oracle recommends that you deploy one Offering in an Implementation Project. In other words, if you are deploying three Offerings, then create three Implementation Projects.
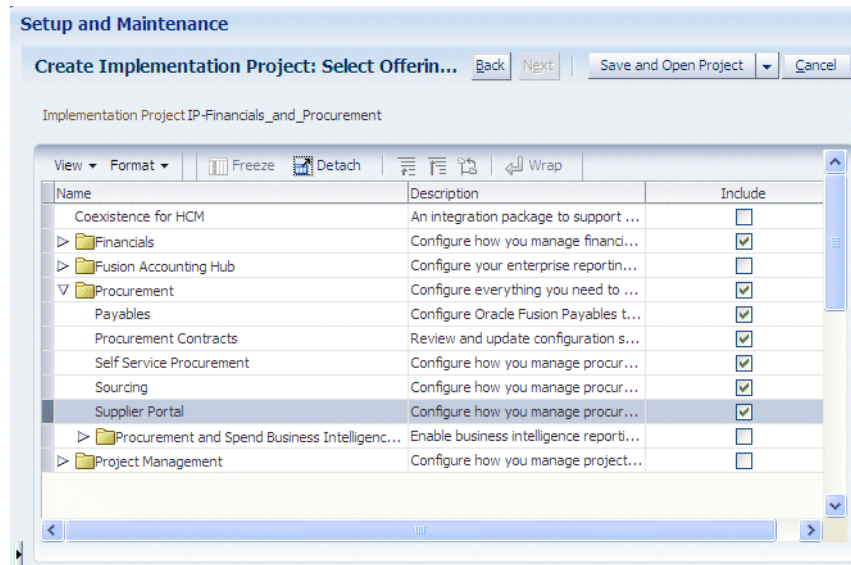
6. Save the details.

When you save the project, FSM generates the list of configuration tasks for the Offering and Functional Areas that you included in the Implementation Project.
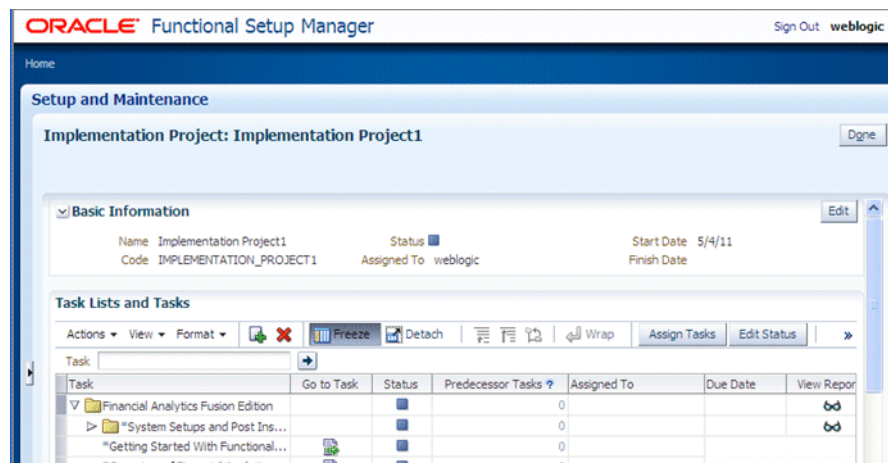
Now that you have created an Implementation Project, FSM generates a list of Tasks required to configure the selected Offering and Functional Areas. You can perform the functional configuration of an Offering in two ways:

- In a small deployment, a single person logged in with BI Applications Administrator Duty privileges can configure Offerings using the 'Go to Task' links. For more information, see Section 5.3.3.3, "How to Perform Functional Tasks using the Administrator role".

- In a large deployment, Functional Developers typically configure Offerings, as follows:

  - The Implementation Manager assigns Tasks to Functional Developers, see Section 5.3.3.2, "How to Assign Tasks to Functional Developers"

  - Functional Developers logged in the BI Applications Functional Developer Duty privileges configure the Offerings (for more information, see Section 5.3.3.4, "How to Perform Functional Tasks using the Functional Developer role").

5.3.3 Additional Steps for Managing Projects in FSM

This section describes additional management tasks that you perform using FSM, and contains the following topics:

5.3.3.1 About Performing Functional Tasks

When you complete a Functional Task in FSM, you update the status of the Task (for example, to 'Completed', or 'Completed with Errors').

When you click 'Go To Task' for an Informational Task, you display a list of steps that you must perform externally to FSM. For example, you might need to configure a value in DAC. When you have completed the steps listed in the Informational Task, you must manually set the status of the Task to 'Completed'.

If a parameter value must be re-set after the initial configuration is completed, the BI Administrator can alternatively use the Setup Data Maintenance and Administration options in Oracle BI Applications Configuration Manager.

5.3.3.2 How to Assign Tasks to Functional Developers

You assign Tasks to Functional Developers so that Functional Developers can configure BI Application Offerings. By default, Tasks are assigned to the BI Applications Administrator user. When you assign a Task to a Functional Developer, that Task is displayed in the Assigned Implementation Tasks tab in FSM when that person is logged into FSM.

To assign tasks to Functional Developers:

1. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link to start FSM.

   You must be logged into Oracle BI Applications Configuration Manager with the Implementation Manager (or Administrator role).

2. Display the Implementation Projects tab, and select the Implementation Project.

3. On the Task Lists and Tasks pane, select one or more Tasks, click Assign Task to display the Assign Tasks page.

4. Use the Assign Tasks page to search for and select appropriate Functional Developers.

When Functional Developers log in and display the Assigned Implementation Tasks tab, they only see the Tasks that have been assigned to them (for more information, see Section 5.3.3.4, "How to Perform Functional Tasks using the Functional Developer role"). When BI Administrators log in and display the Implementation Projects tab, they see all Tasks (for more information, see Section 5.3.3.3, "How to Perform Functional Tasks using the Administrator role").

5.3.3.3 How to Perform Functional Tasks using the Administrator role

In a small deployment project, a single person with BI Applications Administrator Duty privileges might perform the setup and functional configuration tasks for Oracle BI Applications.

When you log into FSM with BI Applications Administrator Duty privileges, you see all Tasks that are included in an Implementation Project.

To perform functional tasks as Administrator:

1. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link to start FSM.

   You must be logged into Oracle BI Applications Configuration Manager with BI Applications Administrator Duty privileges.

2. Display the Implementation Projects tab, and select the Implementation Project that you created for your Offerings.

3. On the Task Lists and Tasks pane, select a Task, and click Go to Task.

   When you click Go to Task, you display a configuration screen that enables you complete the task.

5.3.3.4 How to Perform Functional Tasks using the Functional Developer role

In a medium to large deployment project, a number of Functional Developers might perform functional tasks for an Offering. When you log into Functional Setup Manager with the Functional Developer role, you only see Tasks that have been assigned to you. You do not see Tasks that have been assigned to other Functional Developers.

To perform functional tasks using the Functional Developer role:

1. Log into Oracle BI Applications Configuration Manager with the Functional Developer role.

2. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link.

3. Display the Assigned Implementation Tasks tab.

4. On the Task Lists and Tasks pane, select a Task, and click Go to Task.

   When you click Go to Task, you display a configuration screen that enables you complete the task.

5.3.3.5 How to Monitor Implementation Projects

You use Functional Setup Manager to monitor Implementation Projects to track progress of the implementation.

To monitor Implementation Projects:

1. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link to start FSM.

   You must be logged into Oracle BI Applications Configuration Manager with an Implementation Manager role.

2. Display the Implementation Projects tab, and select the Implementation Project that you want to deploy.

   For example, the Overview page enables you to monitor progress using pie chart.

3. Use the Implementation Project Details pane to monitor the status of the project.

You can also use Oracle BI Applications Configuration Manager to monitor the progress of a project, maintain setup data, and extend Oracle Business Analytics Warehouse if required. For more information, see Section 6, "Administering and Maintaining Functional Configuration Data".

5.3.3.6 How to Monitor Functional Setup Tasks

You monitor Functional Tasks that are assigned to you to track your progress in configuring the ETL for the Oracle Fusion Applications that are being deployed.

To monitor Functional Tasks assigned to me:

1. From the **Tasks** bar in Oracle BI Applications Configuration Manager, select the **Perform Functional Configurations** link to start FSM.

   You must be logged into Oracle BI Applications Configuration Manager with the Administrator role.

2. Display the Manage Implementation Projects page, and select the Implementation Project that you want to deploy.

3. Use the graphs and charts to monitor the progress on the selected Implementation Project.

5.3.3.7 About Completing Tasks in the Common Areas and Dimensions Configurations Task List

Whichever Offerings you include in an Implementation Project, the Tasks will include a **Common Areas and Dimensions Configurations** Task List, which includes Tasks that are common to multiple Offerings. For example, Configure Global Currencies is a Task that is common to multiple Offerings.
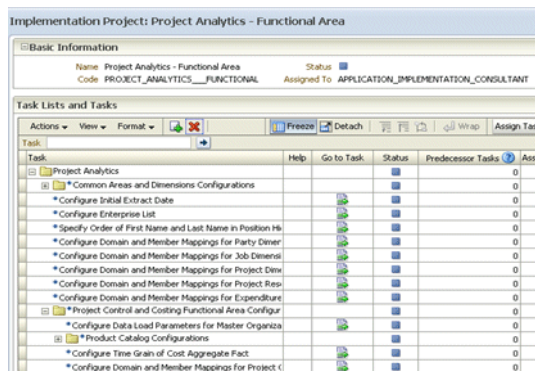
When a Task is completed, the Task status is set to 'Completed'. Completed Tasks in the **Common Areas and Dimensions Configurations** Task List are set to status 'Completed' across all Offerings. In other words, you only need to perform the Tasks in **Common Areas and Dimension Configurations** Task List once for each Oracle BI Applications deployment.

6

Administering and Maintaining Functional Configuration Data

This section explains how to use Oracle BI Applications Configuration Manager and Functional Setup Manager to administer and maintain functional configuration data.

This section contains the following topics:

6.1 About Maintaining and Administering Functional Configuration Data

During an Oracle BI Applications deployment project, you use Oracle BI Applications Configuration Manager and Functional Setup Manager to manage and make changes to configuration values for Data Load Parameters, Domains and Mappings, and Reporting Parameters.

For Functional Configuration road map information, see Section 6.2, "Roadmap for Setup Data Maintenance and Administration".

6.2 Roadmap for Setup Data Maintenance and Administration

The list below provides a high-level roadmap for Setup Data Maintenance and Administration.

High-Level Setup Data Maintenance and Administration Roadmap

1. During the Functional Configuration stage, you typically use Oracle BI Applications Configuration Manager to monitor the setup data values.

   - To monitor Domain Mappings, in Oracle BI Applications Configuration Manager, select the **Manage Domains and Domain Mappings** link from the Tasks bar to display the select Manage Domains and Domain Mappings dialog.
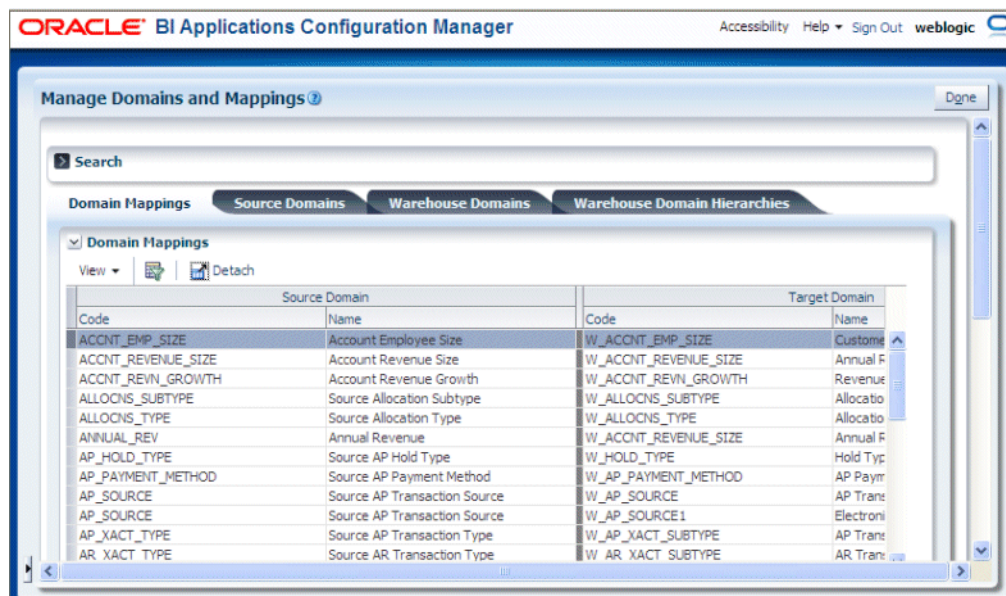
   Note: Alternatively, in Functional Setup Manager, select the **Go to Task** link for a Task that updates a Domain or Domain Member mappings. For example, the Task name 'Manage Domains and Member Mappings for Employee Dimension' updates Domain and Domain Member Mappings.

   - To monitor Data Load Parameters, select the **Data Load Parameters** link from the Tasks bar to display the select Manage Data Load Parameters dialog.
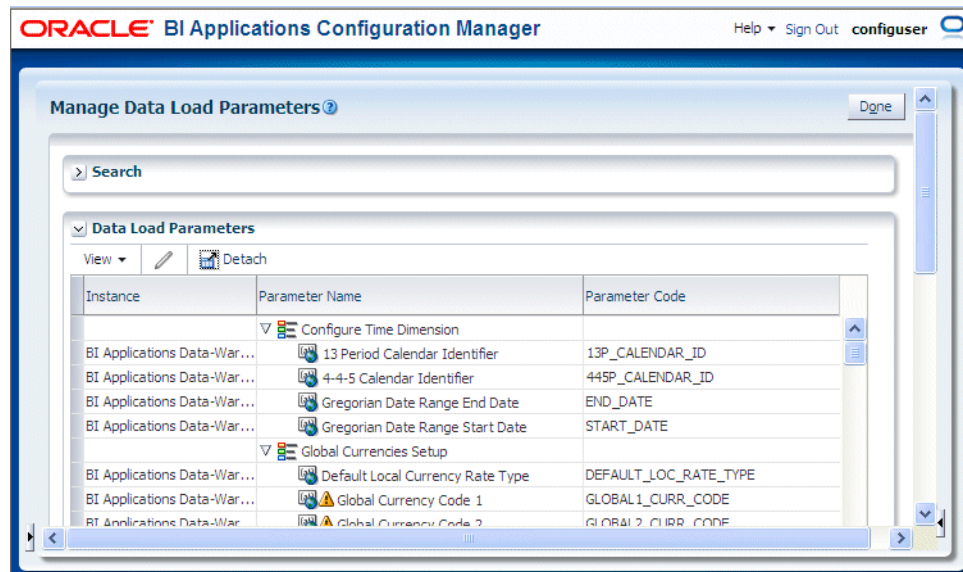
Note: Alternatively, in Functional Setup Manager, select the **Go to Task** link for a Task that updates a Data Load Parameter value. For example, the Task named 'Configure Data Load Parameter Workforce Adjusted Service Date' updates a Data Load Parameter value.

■ To monitor Reporting Parameters, select the **Manage Reporting Parameters** link from the Tasks bar to display the select Manage Reporting Parameters dialog.
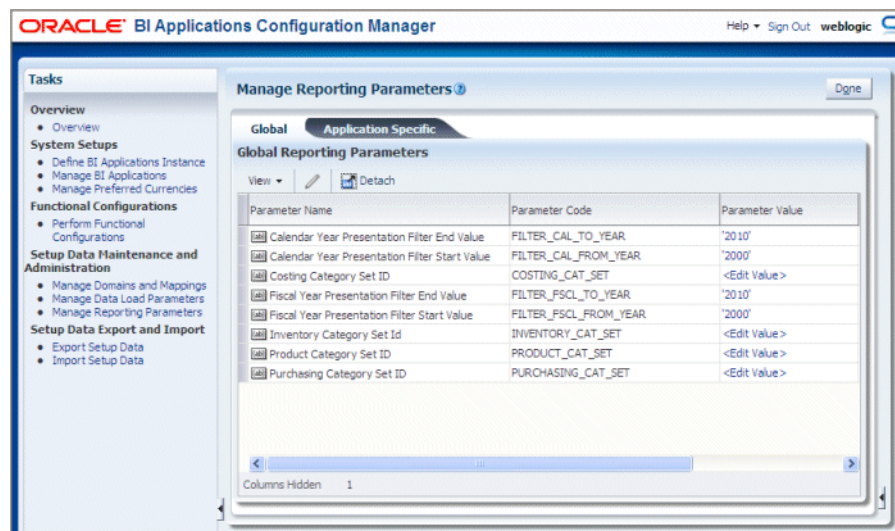
Note: Alternatively, in Functional Setup Manager, select the **Go to Task** link for a Task that updates a Reporting Parameter. For example, the Task named 'Configure Reporting Parameters for Year Prompting' updates a Reporting Parameter.

Tip: Domain values can be used to define delivered business metrics; therefore, you must review the delivered domain member values and map them to the correct source values.

2. During Functional Configuration of purchased Offerings, you use Functional Setup Manager to monitor the status of Tasks and the setup data values that have been set.

   For example, you might use the Implementation Project page in Functional Setup Manager to assess the number of completed tasks.
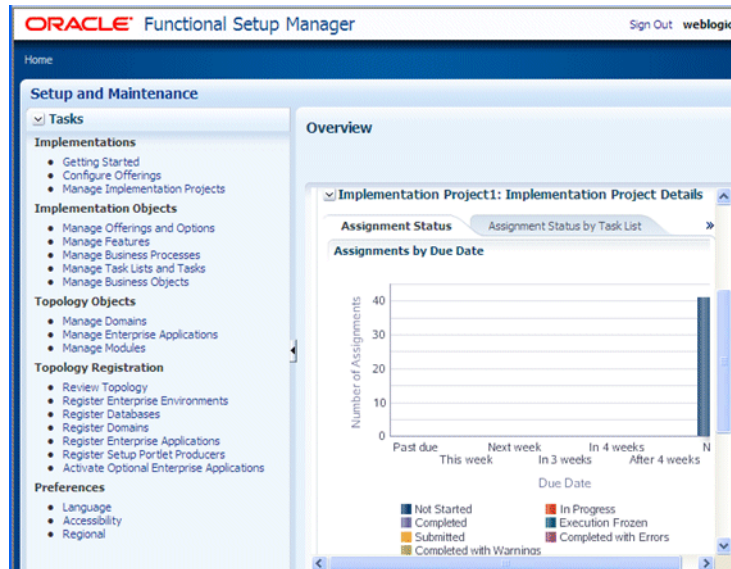
3. If required, you use Oracle BI Applications Configuration Manager to make changes to the configuration values. For more information, see Section 6.8, "How to Change Configuration Values Using Oracle BI Applications Configuration Manager".

4. If required, you use Oracle BI Applications Configuration Manager to add Domains or Domain Member values. For more information, see Section 6.11, "About Adding Target Domain Members".

6.3 About Functional Configuration Data

Functional Configuration Data for Oracle BI Applications is information about the following:

■ Domains and Mappings (for more information, see Section 6.4, "About Working With Domains and Domain Mappings").

■ Data Load Parameters (for more information, see Section 6.5, "About Working With Data Load Parameters").

■ Reporting Parameters (for more information, see Section 6.6, "About Working With Reporting Parameters").

Functional Configuration Data for Oracle BI Applications is:

■ configured using Functional Tasks in Functional Setup Manager.

■ monitored and updated using Oracle BI Applications Configuration Manager.

6.4 About Working With Domains and Domain Mappings

Domains are pre-seeded dimensional values that help define business metrics. For example, in Financial Analytics, domains store information about the General Ledger accounts.

To manage Domains, you use the Manage Domains and Mappings dialog (see the following screenshot).

To display the Manage Domains and Mappings dialog, do one of the following:

- In Oracle BI Applications Configuration Manager, select the Manage Domains and Mappings link in the Tasks pane.

- In Functional Setup Manager, select the Go to Task link for a Task that updates a Domain and Domain Member mappings.
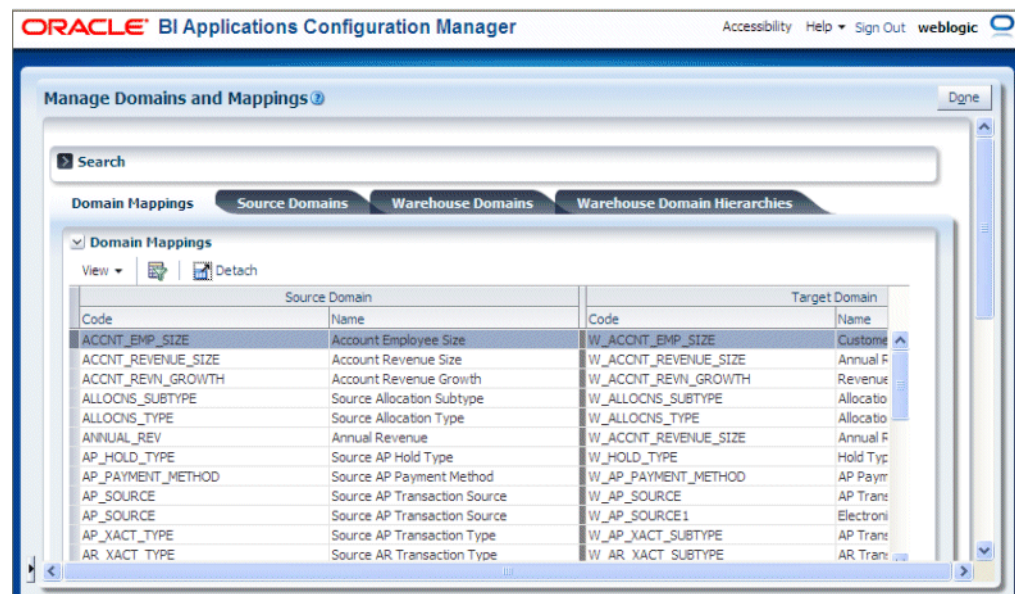
The Manage Domains and Mappings dialog contains the following tabs:

- Domain Mappings - this tab shows how data fields in the Source System map to data fields in Oracle Business Analytics Warehouse (for more information, see Section 6.4.1, "About Domain Mappings and Domain Member Mappings").

- Source domains - this tab shows data fields and Domain Members in the Source System (for more information, see Section 6.4.2, "About Source Domains").

- Warehouse Domains - this tab shows data fields and Warehouse Members in Oracle Business Analytics Warehouse (for more information, see Section 6.4.3, "About Warehouse Domains").

- Warehouse Domain Hierarchies - this tab shows Domains that have been organized into hierarchies to enable the data to be more effectively analyzed (for more information about this tab, see Section 6.4.4, "About Warehouse Domain Hierarchies").

6.4.1 About Domain Mappings and Domain Member Mappings

Domain Mappings specify how data in a source system is extracted and loaded into Oracle Business Analytics Warehouse. For example, the data in domain Source Group

Account (BI_GROUP_ACCOUNT) extracts and loads into the domain Group Account (W_GL_GROUP_ACCOUNT).

The following screenshot shows example domain mappings for Oracle Financial Analytics.
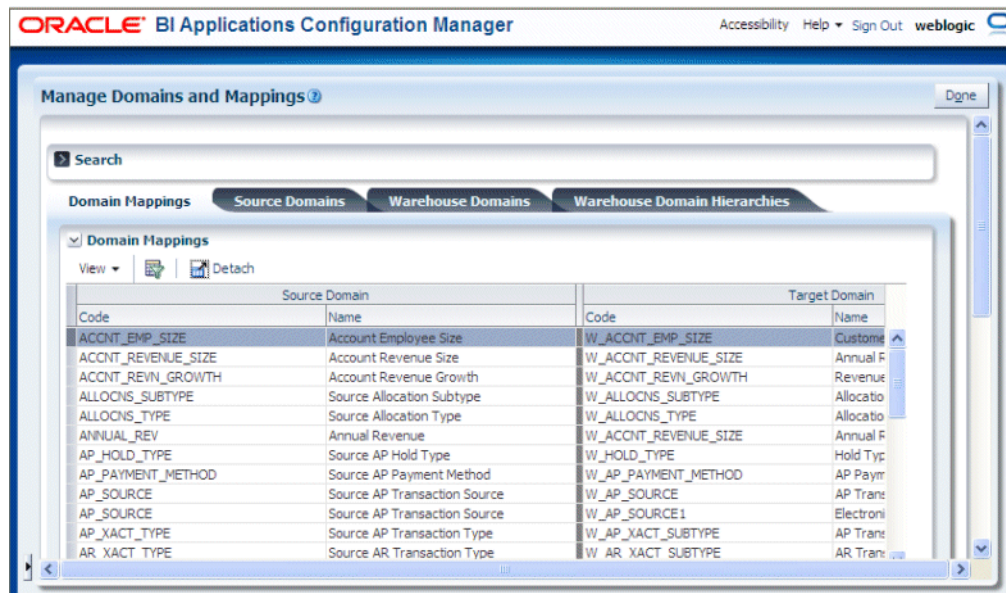
Domain Member Mappings specify how domain member data in a source system is extracted and loaded into domain member data in Oracle Business Analytics Warehouse. For example, in Oracle HR Analytics, domain Gender (W_SEX_MF_CODE) has a source value 'Male' that is mapped to a domain member value 'M' in Oracle Business Analytics Warehouse.

6.4.1.1 About Regular Domains and Band Domains

There are two types of Domain, as follows:

- Regular Domains

  Regular Domains have members consisting of a single value. For example, members for a Purchase Order Status domain might have the following members:

  – Cancelled

  – Closed

  – Incomplete

  These single values map to single member values in the target system. For example, Cancelled maps to Cancelled, Closed maps to Closed, and so on.

- Band Domains

  Band Domains have members consisting of two values (Range Start, and Range End) that specify a range. For example, an Account Employee Size domain might have the following members:

  – 1, 5000

  – 5001, 10,000

  – 10,001, 1,000,000.

Each range maps to a single target Domain Member. For example, 1, 5000 maps to Small, 5001, 10,000 maps to Medium, and so on.

6.4.2 About Source Domains

Data fields in a Source System application are referred to as Source Domains. Data fields in Oracle Business Analytics Warehouse are referred to as Warehouse Domains.

The following screenshot shows example source domains for Oracle Financial Analytics.

Note: Source Domains displayed on the Source Domains tab are read-only.
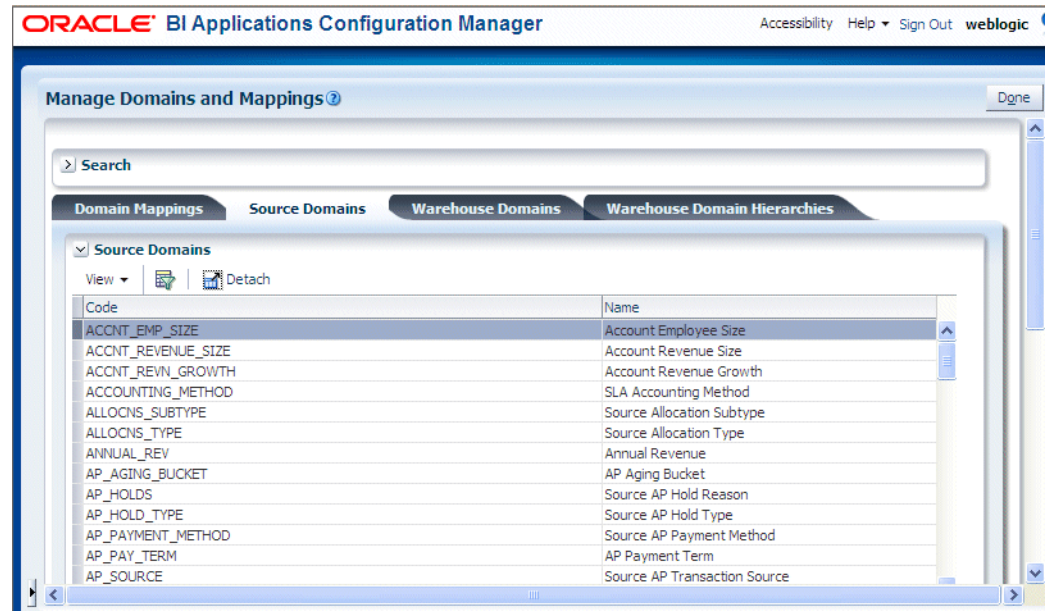
Domain Members are the permitted values for a Source or Warehouse Domain. For example, the Domain Members for MARITAL_STATUS include D for Divorced, M for Married, S for Single, and so on.

> **Tip:** Domain values can be used to define delivered business metrics; therefore, you must review the delivered domain member values and map them to the correct source values.

Domain Mappings specify how entities in a Source System application are loaded into Oracle Business Analytics Warehouse.

6.4.3 About Warehouse Domains

Data fields in Oracle Business Analytics Warehouse are referred to as Warehouse Domains.

The following screenshot shows example warehouse domains for Oracle Financial Analytics.
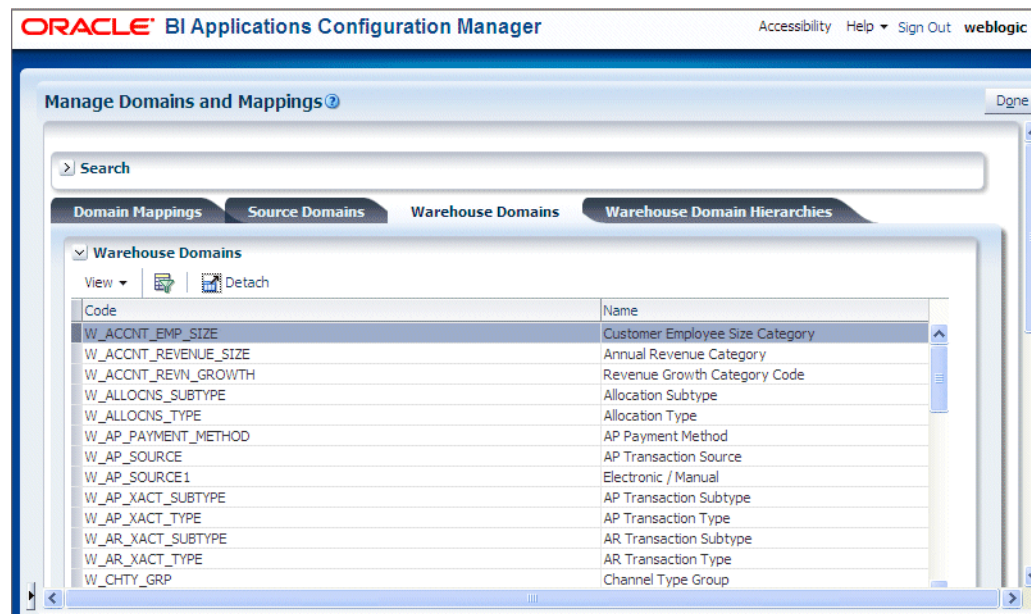
6.4.4 About Warehouse Domain Hierarchies

Warehouse Domain Hierarchies are Domains that have been organized into hierarchies to enable the data to be more effectively analyzed. For example, in Oracle HR Analytics, you might need to have a workforce event hierarchy: Event Group -> Event Sub-group -> Event Detail.

Note: Domain Hierarchies are displayed in inverted format, that is in the following format:

<Child 1>\

    <Child *n*>\

       <Parent>.

For more information, see Section 6.12, "Viewing Domain Hierarchies".

The following screenshot shows example warehouse domain hierarchies for Oracle Financial Analytics.
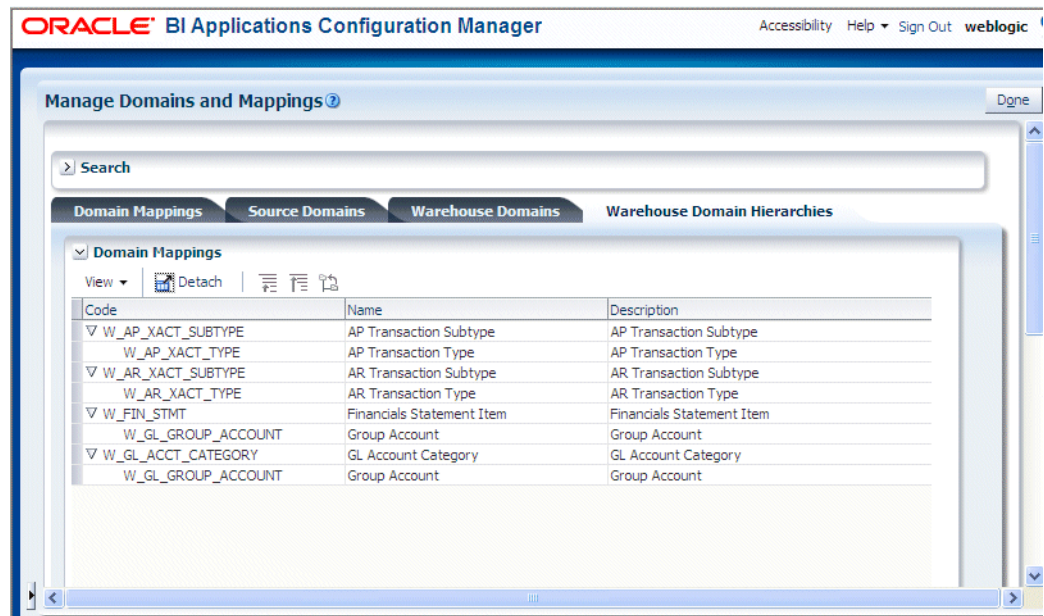
6.4.5 About Setting Up Domain Member Mappings

Oracle Business Intelligence Applications ships default domain value mappings that map the seeded BI Application domain values to the seeded configuration data in Oracle Enterprise Resource Planning applications.

When you configure your Offerings, you need to review the default mappings for domain values, and if necessary update them to suit the categories that you want to use to report on your data.

For example, in Oracle HR Analytics, the default domain values for Performance Range might be similar to the following:

0 - 50: PERF_RANGE_1

50 - 60: PERF_RANGE_2

60 - 70: PERF_RANGE_3.

If you want to use these default categories, you do not need to make any changes to these mappings before you start your ETL processes.

If you want to make changes to default mappings, then you use the Domain Mappings tab to implement these changes.

For example, you might want to change the range for PERF_RANGE_1 from 0 - 50 to 0 - 100. Or you might want to add a new category named PERF_RANGE_4 and assign the range 100 - 500 to the new PERF_RANGE_4 category.

6.4.5.1 How to edit a Domain Member Mapping

To edit a Domain Member Mapping:

1. Navigate to the Domain that you want to edit, using the "Manage Domains and Mappings: Domain Mappings tab".

   To display the Domain Mapping tab, do one of the following:

   - In Oracle BI Applications Configuration Manager, select the Manage Domains and Mappings link in the Tasks pane, display the Domain Mappings tab, then select a Domain.

    ■   In Functional Setup Manager, select the Go to Task link for a Task that updates a Domain or Domain Member Mappings.

2. Scroll down to the Domain Member Mappings pane.

3. Click the Edit Domain Member Mappings icon to display the "Edit Domain Member Mappings dialog",
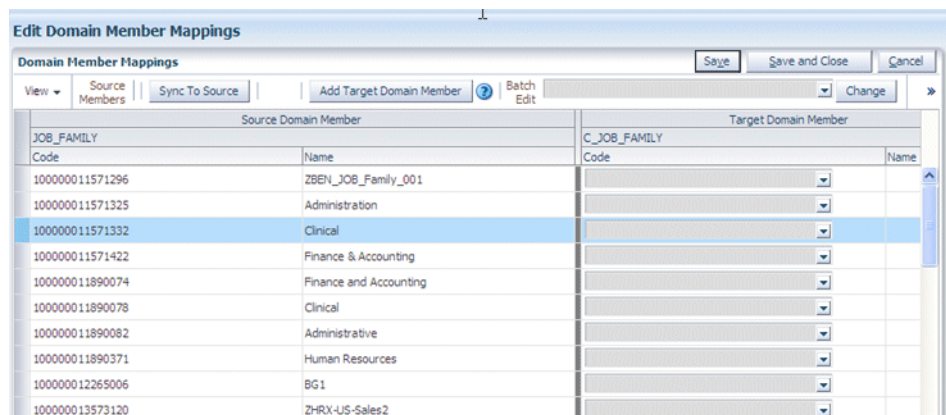
4. Use the "Edit Domain Member Mappings dialog" to edit the domain mapping values.

6.4.5.2 How to add a Range Member Mapping

To add a Range Member Mapping:

1. Navigate to the Domain that you want to edit, using the "Manage Domains and Mappings: Domain Mappings tab".

   To display the Domain Mapping tab, do one of the following:

   ■   In Oracle BI Applications Configuration Manager, select the Manage Domains and Mappings link in the Tasks pane, display the Domain Mappings tab, then select a banded (or ranged) Domain.

- In Functional Setup Manager, select the Go to Task link for a Task that updates a banded or ranged Domain or Domain Member Mappings.
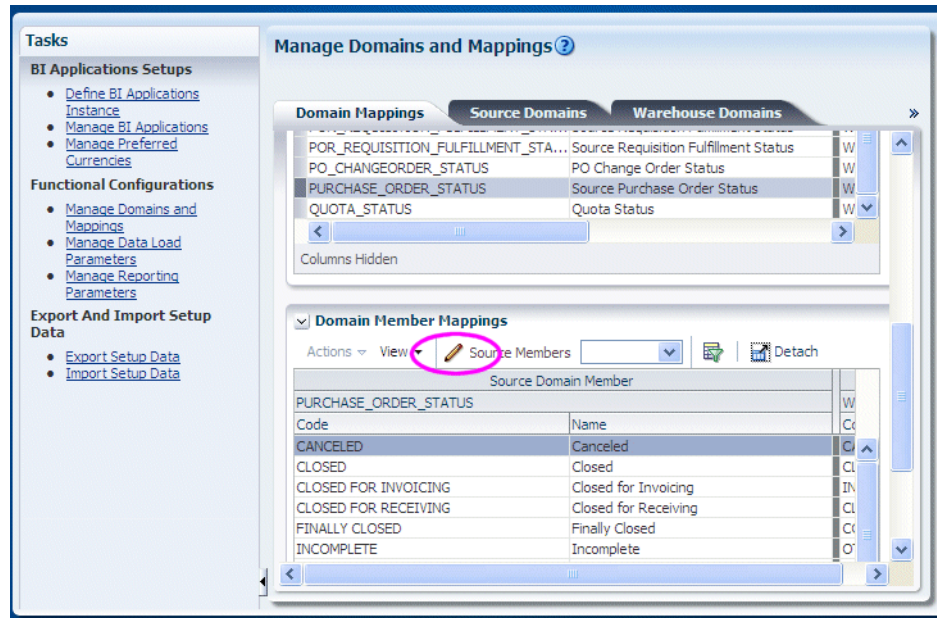
2. Scroll down to the Domain Member Mappings pane.

3. Click the Edit Domain Member Mappings icon to display the "Edit Domain Member Mappings dialog",
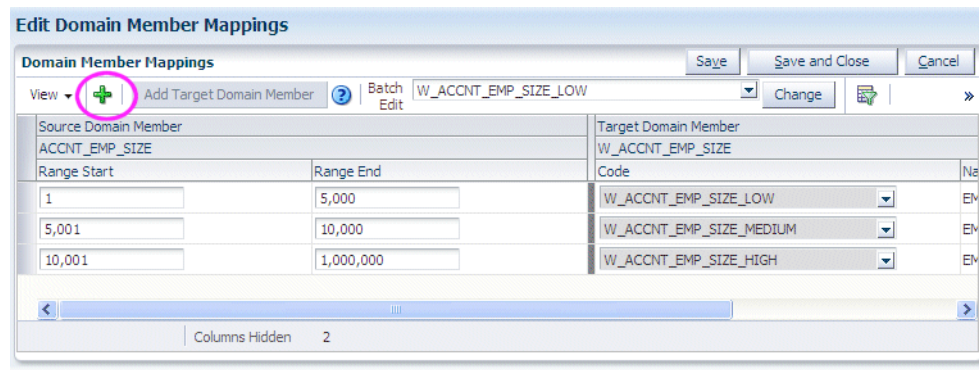
4. Click the Add Range Member Mapping (+) icon and specify values in the **Range Start**, **Range End**, and **Target Domain Member** - **Code** fields.

   Tip: Before you create a new range, you might first want to use the **Add Warehouse Member** button to first create a target Warehouse Member, which is then available as an option in the **Target Domain Member** - **Code** list. For example, you might add a Warehouse Member called 'Greater than 250,000' to map to the range 250,000 to 1,000,000.

6.4.5.3 How to add a Target Domain Member

To add a Target Domain Member:

1. Navigate to the Domain that you want to edit, using the "Manage Domains and Mappings: Domain Mappings tab".

   To display the Domain Mapping tab, do one of the following:

■ In Oracle BI Applications Configuration Manager, select the Manage Domains and Mappings link in the Tasks pane, display the Domain Mappings tab, then select a Domain.

■ In Functional Setup Manager, select the Go to Task link for a Task that updates a Domain or Domain Member Mappings.

2. Scroll down to the Domain Member Mappings pane.

3. Click the Edit Domain Member Mappings icon to display the "Edit Domain Member Mappings dialog",
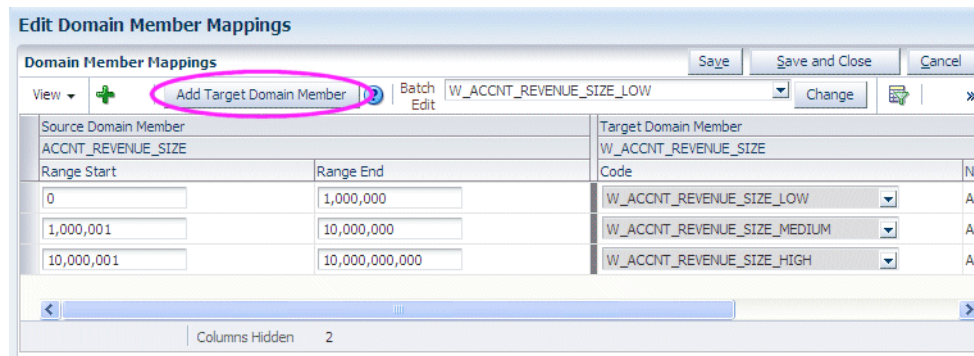
4. Click **Add Target Domain Member** to display the Add Target Domain Member dialog, which enables you to specify a Name, Code, and optional Description.
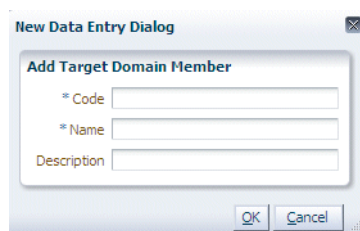
For example, you might add a Warehouse Member called 'Greater than 250,000' to map to the range 250,000 to 1,000,000.

Note: If the **Add Target Domain Member** option is grayed out or not displayed, then the domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").

When you click OK to return to the "Edit Domain Member Mappings dialog", you can map a Source Domain to the Target Domain that you just created.

6.4.5.4 How to Localize a New Domain Member

If you added a new domain member and it requires localization, follow the procedure in this section.

To add string localizations for Oracle BI Applications Configuration Manager metadata:

1. Open a database administration tool, and connect to the Oracle Business Analytics Warehouse schema.

2. Query for the table C_DOMAIN_MEMBER_TL and the new domain member record you added in Section 6.4.5.3, "How to add a Target Domain Member."

   You can query for the new domain member record by using the query filter on the columns CREATED_BY and CREATION_DATE.

   Each new domain member record will have 28 rows in the C_DOMAIN_ MEMBER_TL table.

3. Update the LANGUAGE_CODE column to match the localized deployment language:

   a. Identify the language code for the localized language using the following SQL:

   ```
   SELECT LANGUAGE_CODE, NLS_LANGUAGE, NLS_TERRITORY
   FROM FND_LANGUAGES_B
   WHERE INSTALLED_FLAG IN ('B', 'I');
   ```

   b. Update the domain member name, description, and source language code strings for the localized language using the following SQL.

   In this example, the localized language is Arabic, and the LANGUAGE_CODE is AR.

   ```
   UPDATE C_DOMAIN_MEMBER_TL
   SET DOMAIN_MEMBER_NAME = '<Arabic translated string for domain member
   name>', DOMAIN_MEMBER_DESCR = '<Arabic translated string for domain member
   description>', SRC_LANGUAGE_CODE = 'AR'
   WHERE DOMAIN_KEY = '<Domain key value for the record you want to update>'
   AND DOMAIN_MEMBER_CODE = '<Domain member code value for the record you want
   to update>'
   AND LANGUAGE_CODE = 'AR';
   ```

4. Exit the database administration tool.

5. Restart the Oracle WebLogic Server.

6.4.5.5 How to Add String Localizations for Oracle BI Repository Metadata

If you added a new domain member, follow this procedure to add string localizations in the Oracle BI Repository metadata.

To add string localizations for Oracle BI repository metadata:

1. Stop the OPMN services.

   Use the command: `opmnctl stopall`.

2. Open a database administration tool, and connect to the Oracle Business Analytics Warehouse schema.

3. Identify the strings for the following presentation objects:

    - Subject area

   - Presentation table

   - Presentation hierarchy

   - Presentation level

   - Presentation column

   For example, for the subject area Payables Invoices - Prepayment Invoice Distributions Real Time, you would enter the following strings:

   Table 6–1 Example of Localization Strings

String	Presentation Object
Payables Invoices - Prepayment Invoice Distributions Real Time	Subject area
Time	Presentation table
Date - Year	Presentation hierarchy
Total	Presentation level
Year	Presentation level
Calendar Year	Presentation column

4. For each subject area, externalize the strings for localization and generate custom names for the presentation objects:

   a. In the Oracle BI Administration Tool, right-click the subject area and select **Externalize Display Names**, and then select **Generate Custom Names**.

   b. Save your work.

      For more information about localizing strings, see "Localizing Metadata Names in the Repository," in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

5. Check the consistency of the repository, and remove any inconsistencies.

   For instructions, see "Checking the Consistency of a Repository or Business Model," in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition)*.

6. Enter the custom name of one of the presentation objects into the table C_RPD_MSGS:

   ```
   INSERT INTO C_RPD_MSGS(MSG_ID, CREATED_BY, CREATION_DATE)
   VALUES('<CUSTOM NAME OF PRESENTATION OBJECT>', 'CUSTOM', SYSTIMESTAMP);
   COMMIT;
   ```

   Note: To view the values for custom names and logical columns in the Administration Tool, right-click the presentation object and select **Properties**. The data in the "Custom display name" field appears in the format VALUEOF(NQ_

SESSION.VALUE, where VALUE is the custom name for a presentation object, or the logical value for a presentation column. This value is the value that you need to enter in the VALUES section of the SQL statement above.

7. Enter the localized string for the presentation object in the previous step into the table C_RPD_MSGS_TL:

```
INSERT INTO C_RPD_MSGS_TL(MSG_ID, MSG_TEXT, LANGUAGE_CODE, CREATED_BY,
CREATION_DATE)
VALUES('<CUSTOM NAME OF PRESENTATION OBJECT>', '<LOCALIZATION OF THE STRING'>,
'<LANGUAGE CODE FOR TRANSLATED LANGUAGE>', 'CUSTOM', SYSTIMESTAMP);
COMMIT;
```

To identify the language code for a particular language, use the following SQL:

```
SELECT LANGUAGE_CODE, NLS_LANGUAGE, NLS_TERRITORY
FROM FND_LANGUAGES_B
WHERE INSTALLED_FLAG IN ('B', 'I');
```

8. Enter additional details about the presentation object into the table C_RPD_MSGS_REL as indicated by the following SQL:

```
INSERT INTO C_RPD_MSGS_REL(MSG_ID, MSG_NUM, MESSAGE_TYPE, CREATED_BY, CREATION_
DATE)
VALUES('<CUSTOM NAME OF PRESENTATION OBJECT>', '<TRANSLATION OF THE STRING'>,
'<LANGUAGE CODE FOR TRANSLATED LANGUAGE>', 'METADATA','CUSTOM', SYSTIMESTAMP);
COMMIT;
```

9. Repeat steps 6 through 8 for each presentation object requiring localization.

10. Validate that the physical connection of the session initialization block INIT_USER_LANGUAGE_CODE is operable:

   a. In the Oracle BI Administration Tool, select **Manage**, **Variables**, **Session Initialization Block**.

   b. Right-click I**NIT_USER_LANGUAGE_CODE**.

   c. In the Properties dialog, click **Edit Data Source**.

   d. Click **Test**, and input the value for the language code. Then, click **OK**.

   For example, for Arabic enter 'AR'.

   The value USER_LANGUAGE_CODE = '<language code>' should be returned.

   If this value is not returned, the TNS entry for the data source is not properly configured.

11. Restart the OPMN services.

12. Verify the localized strings in Oracle BI Answers. On the login page, specify the appropriate language.

6.4.5.6 How to use Sync to Source to synchronize a Target Domain with a Source Domain

In some scenarios, you might only know what target Domain member values should be when you deploy Oracle BI Applications. For example, in Order Management or Supply Chain Analytics, UOM (Unit of Measurement) is typically not known until deployment time. You can set up a non-ranged target domain using the **Sync to Source** option to automatically synchronize a target domain with values from the source domain. This process inserts new target members from the source domain, and

automatically generates 1:1 mappings. This is useful for large domains with many member mappings that might otherwise take a long time to set up.
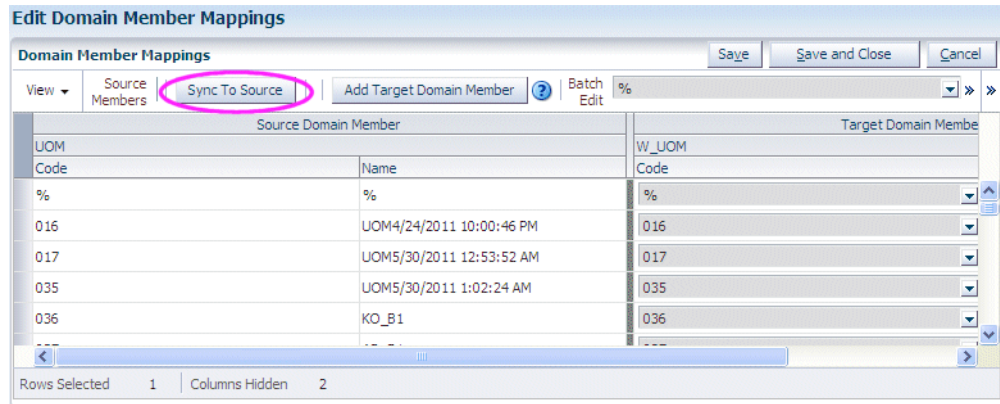
Note: Sync to Source is only available for extensible non-ranged Domains.

How to synchronize a Target Domain with a Source Domain:

1. Navigate to the Domain that you want to synchronize.

   If you are in Oracle BI Applications Configuration Manager, from the Tasks bar click Manage Domains and Mappings, display the Domain mappings tab, select the Domain that you want to edit, then click the Edit Domain Member Mappings icon in the Domain Member Mappings pane to display the "Edit Domain Member Mappings dialog".

   If you are in Functional Setup Manager, when you click **Go to Task** for the Task that is updating a Domain, you display the "Edit Domain Member Mappings dialog" dialog.
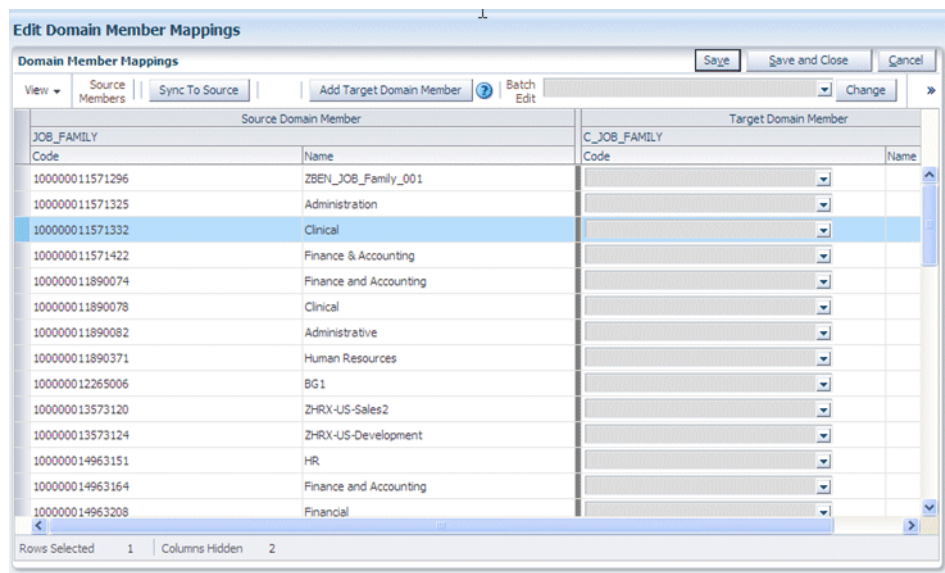
2. Click Sync to Source.

3. At the Warning dialog, click OK.
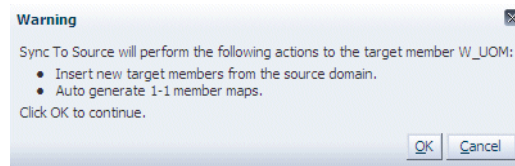
Note: If you click OK to continue, then you commit changes to the target Domain members, even if you do not click 'Save' or click 'Save and Close' on the Edit Domain Member Mappings dialog.

Target Domain member values are generated. In the example below, the target codes for C_JOB_FAMILY are automatically synchronized with the Source member codes.
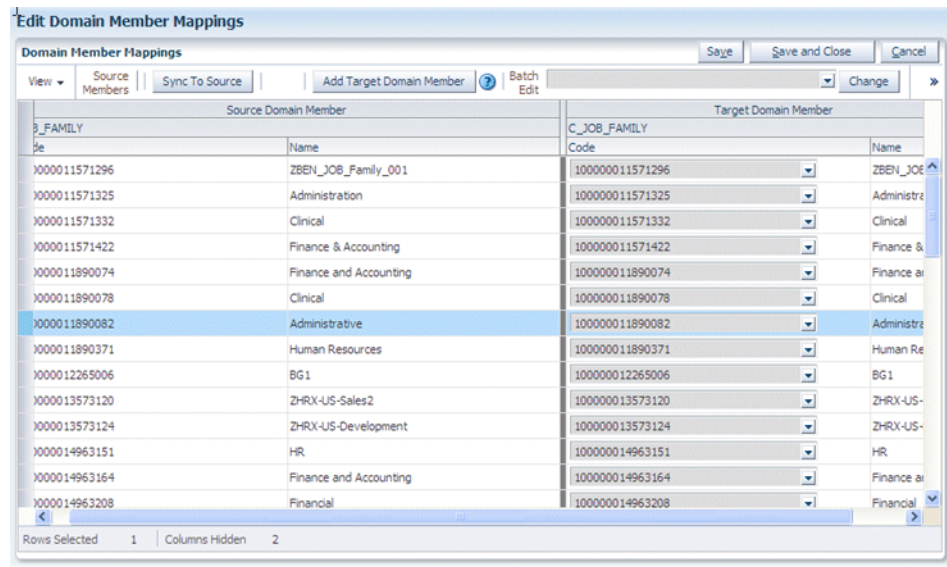
6.4.5.7 How to use Batch Edit to update multiple Target Domain Member Values

You can set up a target domain using the **Batch Edit** option to update multiple target domain members with the same value. This is useful for large domains with many member mappings that require the same value.

To use Batch Edit to update multiple Target Domain Member Values:

1. Navigate to the Domain that you want to edit.

   If you are in Oracle BI Applications Configuration Manager, from the Tasks bar click Manage Domains and Mappings, display the Domain mappings tab, select the Domain that you want to edit, then click the Edit Domain Member Mappings icon in the Domain Member Mappings pane to display the "Edit Domain Member Mappings dialog".

   If you are in Functional Setup Manager, when you click Go to Task for the Task that is updating a Domain, you display the "Edit Domain Member Mappings dialog" dialog.

2. Multi-select one or more rows in the table.

   Tip: To multi-select, use Ctrl + click to select multiple rows.

3. Select a value from the **Batch Edit** drop down list.

4. Click Change to apply the value selected in the **Batch Edit** drop down list to all specified members.

6.4.6 How to modify a Warehouse Domain Hierarchy

Oracle BI Applications Warehouse Domains are organized into hierarchies. You might want to modify a hierarchy to enable data to be more effectively analyzed. For example, you might change the order of items in a hierarchy.

To create or edit a Warehouse Domain Hierarchy:

1. In the Domain mappings list, select the Domain Mapping that you want to edit.

2. Use the options at the top of the Domain Mapping list to change the hierarchy.

6.4.7 Why are some domains non-extensible?

To maintain data integrity in Oracle Business Intelligence Applications, some domains have been designed as non-extensible, and are therefore read-only.

If a domain is non-extensible, then when the domain is selected, the following options are greyed out:

- the Edit icon in the Domain Member Mappings pane (on the "Manage Domains and Mappings: Domain Mappings tab" or "Manage Domains and Mappings: Warehouse Domain Hierarchies tab").

- the Add Target Domain Member button (on the "Edit Domain Member Mappings dialog").

- the Add Warehouse Domain Member button (on the "Manage Domains and Mappings: Warehouse Domains tab"\Warehouse Members pane).

6.5 About Working With Data Load Parameters

Data Load Parameters are configuration values that specify how Source System data is loaded into Oracle Business Analytics Warehouse. For example, the cost time grain parameter COST_TIME_GRAIN specifies whether costs are aggregated weekly, monthly, or quarterly, during the data loading process.

6.5.1 Key points about working with Data Load Parameters

When you work with Data Load Parameters in Oracle BI Applications Configuration Manager, note the following key points:

- Data Load Parameters can be either Global or Application Specific, as follows:

  - Global parameters apply to all applications, and are indicated with the (ab) and globe icon:

    Global Data Load Parameters can also be associated with specific Fact Groups or Dimension Groups.

  - Application Specific apply to specific applications, and indicated with the (ab) icon:

    Application Specific Data Load Parameters are always associated with one or more Fact Groups or Dimension Groups.

- If a Global Data Load Parameter is associated with one or more Fact Groups or Dimension Groups, then each Fact Group or Dimension Group can have a different value. For example, the value of Slowly Changing Dimension Flag for Absence Even Dimension might be Yes, and the value of Slowly Changing Dimension Flag for Asset Dimension might be No.

  Values for Global Group-specific parameters are always overridable, and are indicated by the Overridable Parameter icon (that is, the (ab) icon with a pencil):

  The example below shows the Global parameter 'Slowly Changing Dimension Flag' with associated Group-specific values displayed in the lower pane.
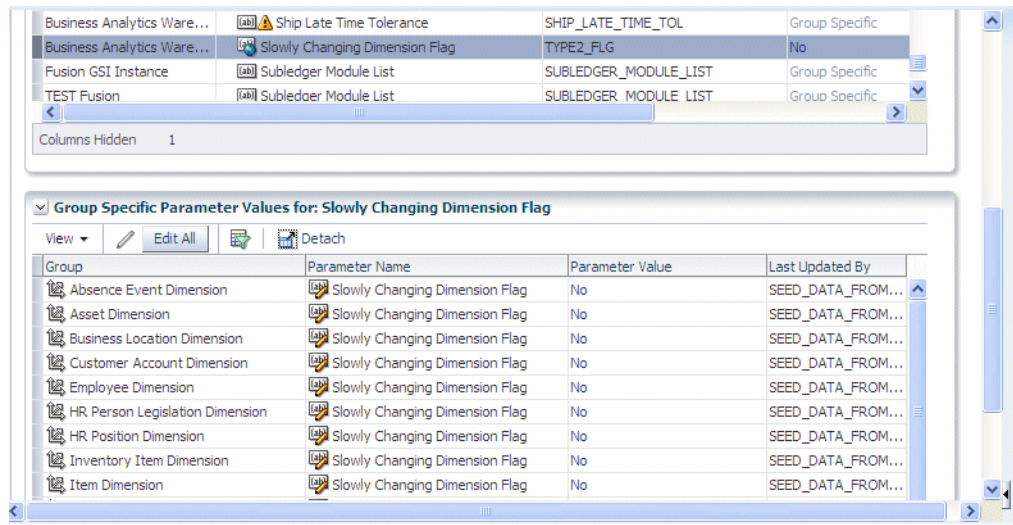
- For Application Specific Data Load Parameters, values for associated Fact Groups or Dimension Groups can either be Overridable or Non-Overridable, as follows:

  Overridable - each associated Fact Group or Dimension Group can have a different value. Overridable parameters are indicated by the Overridable Parameter icon (that is, the (ab) icon with a pencil):

  Non-Overridable - each associated Fact Group or Dimension Group must have the same value. Non-Overridable parameters are indicated by the Parameter icon (that is, the (ab) icon without a pencil):

  The example below shows the Application Specific parameter 'Product Category Set ID 3' with associated Group-specific values displayed in the lower pane. Note that the Group-specific values in this example cannot be overridden.
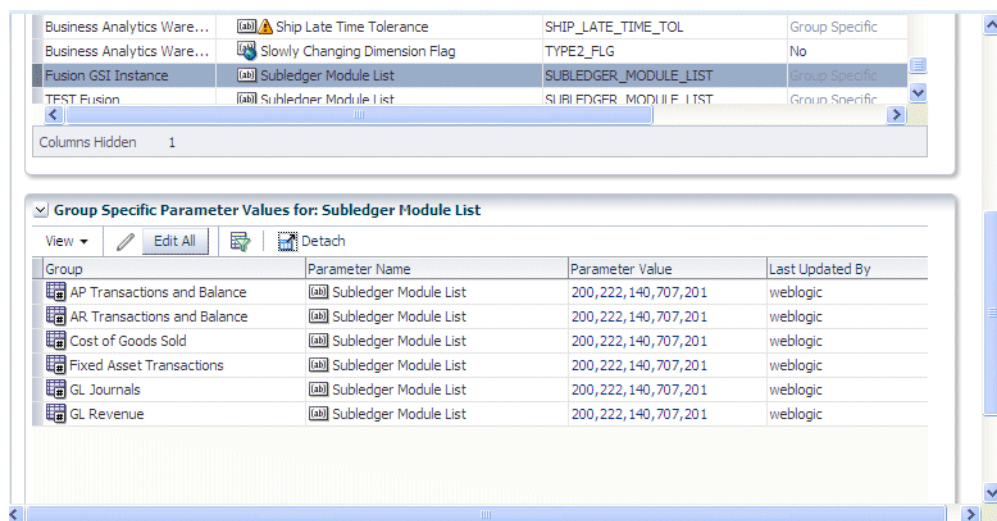
6.5.2 About editing Data Load Parameters

You edit Data Load Parameters (whether they are Global or Application Specific) using the Data Load Parameters dialog, which is displayed in Functional Setup Manager when you edit a Data Load Parameter (see the following screenshot). The Data Load Parameters list above (or master table) displays the parameters and values, and the Group Specific Parameter Values list below displays associated Fact Groups or Dimension Groups (if there are any).
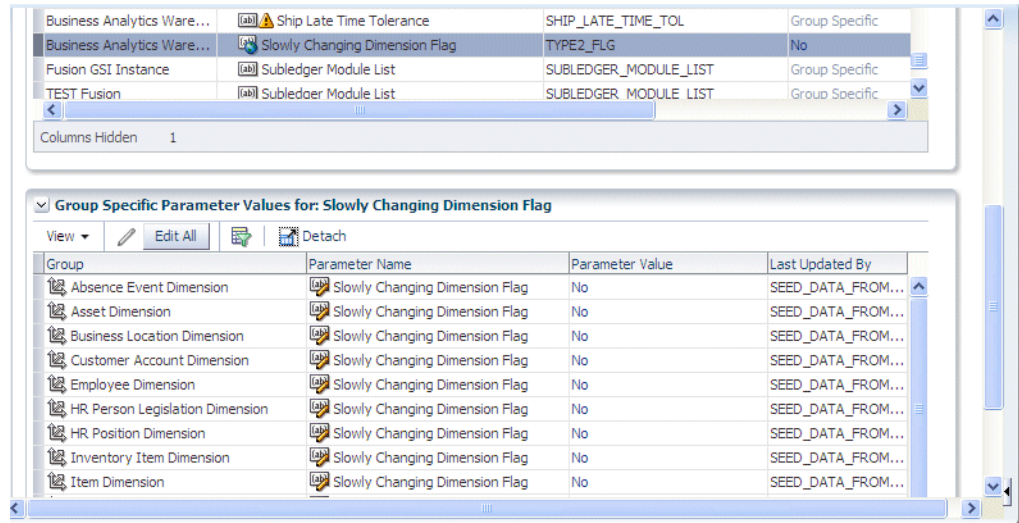
To display Data Load Parameters in Oracle BI Applications Configuration Manager, select the **Manage Data Load Parameters** link on the Tasks bar to display the "Manage Data Load Parameters dialog".
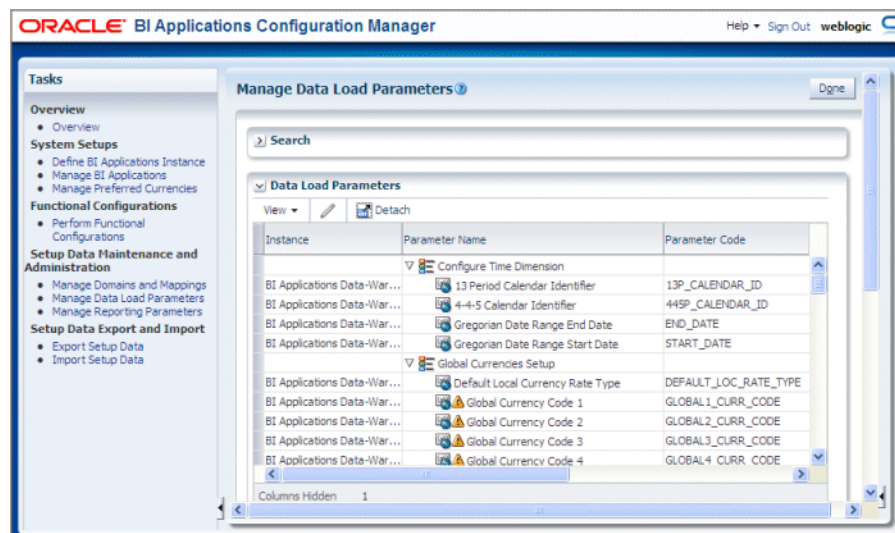
6.5.3 About Editing Global Data Load Parameters

To edit the value of a Global Parameter, select the record in the **Data Load Parameters** master table, and then either click the Edit icon in the table toolbar or click on the link in the **Global Parameter Values** column.

If the Global Parameter that you edit is associated with Fact Groups or Dimension Groups, then a warning message is displayed to verify that you want to update the

value for all associated Fact Groups and Dimension Groups. If you click Yes at the warning message, then the values of all occurrences of the parameter at the Group level will be updated to the new value.

For example, if you click the Edit icon or the link in the Parameter Value column for the parameter SUBLEDGER_MODULE_LIST in the Data Load Parameters master table, then the following Warning dialog is received:
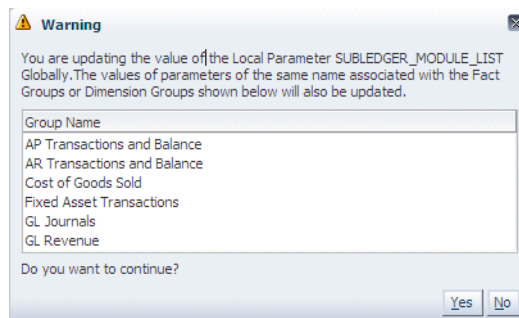
Clicking Yes allows you to continue with the edit of the parameter value. A change to the parameter value is applied to all parameter occurrences at the Group level.

To change the value of parameter for a specific Fact Group or Dimension Group, select the Global Parameter in the Data Load Parameters master table, and then select the parameter in the Group Specific Parameter Values for: <Parameter Name> detail table. Click on the Edit icon in the table tool bar or the link in the Parameter Value column to open the Parameter Value Edit dialog.
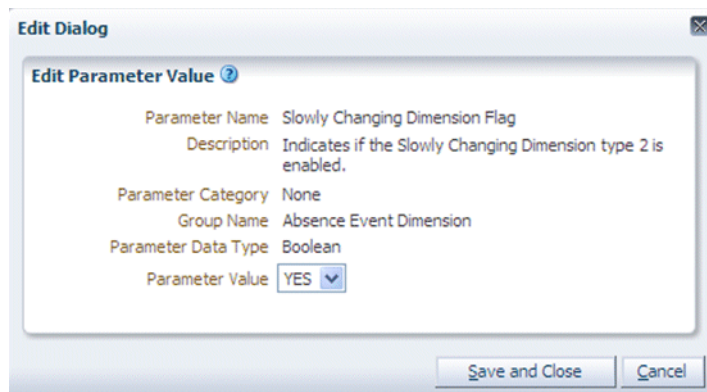
Changing the parameter value in the above Edit Dialog updates the parameter value for the Slowly Changing Dimension Flag parameter associated with the Absence Event Dimension Group.

6.5.4 About Editing Application Specific Parameter Values

You use the **Edit** option to edit the value of a Fact Group or Dimension Group that is associated with an Application Specific parameter (if the parameter is overridable).

You use the **Edit All** option to edit the value of all Fact Groups and Dimension Groups that are associated with an Application Specific parameter.

If you attempt to edit the value of a specific Fact Group or Dimension Group, then:

- if the parameter is overridable at the Group level, then you update only that specific Fact Group or Dimension Group.

■ if the parameter is Non-Overridable at the Group level, then a Warning dialog displays a list of Fact Groups and Dimension Groups that will be affected if you click Yes to edit the value.
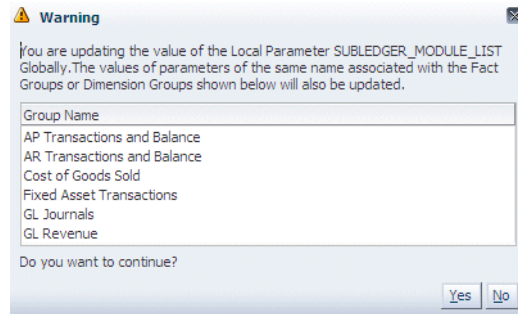
6.5.5 How to Edit a Data Load Parameter value

You edit Data Load Parameters using the Manage Data Load Parameters dialog.

To display the Manage Data Load Parameters dialog, do one of the following:

■ In Oracle BI Applications Configuration Manager, select the **Manage Data Load Parameters** link in the Tasks pane.

■ In Functional Setup Manager, select the **Go to Task** link for a Task that updates a Data Load Parameter (for example, Configure Initial Extract Date).

To edit a Data Load Parameter:

1. Select the parameter in the **Data Load Parameters** list.

   If the parameter is a Global parameter, then the **Global Parameter Value** field displays the actual value (for example, 'MONTHLY').

   If the parameter is a non-Global parameter, then the **Global Parameter Value** field displays the text 'Group Specific').

2. Do one of the following:

   ■ To update a Global parameter, in the **Data Load Parameters** list click the value displayed in the **Global Parameter Value** field to display the "Edit Parameter Value dialog (for Data Load Parameters)", and edit the value. Alternatively, select the row and click the Edit icon.
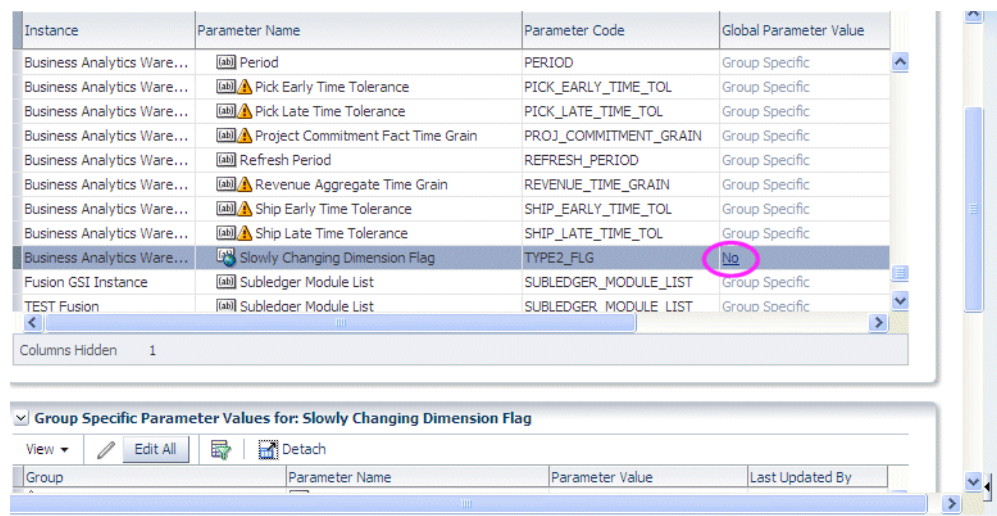
If a Global parameter is associated with Fact Groups or Dimension Groups, you can override the Global value for specific Fact Groups or Dimension Groups by editing the parameter value in the **Group specific Parameter Values for** pane.

■ To update an Application Specific parameter, select the parameter in the Data Load Parameters list, scroll down to the **Group Specific Parameter Values for** list, and click either the Edit icon, or the value in the Parameter Value column (or select the parameter row and click the Edit icon).
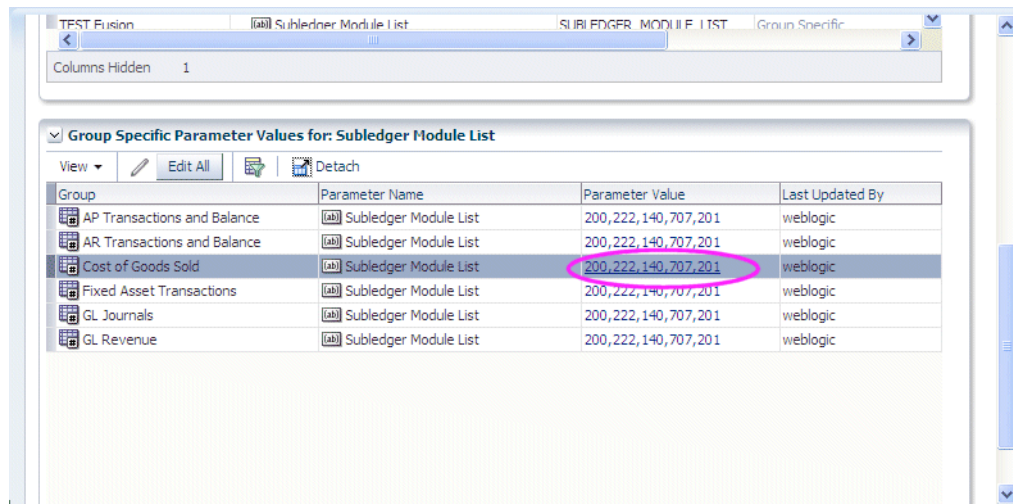
If the parameter is overridable at the Group level, then you update only that specific Fact Group or Dimension Group using the Edit Parameter Value dialog.
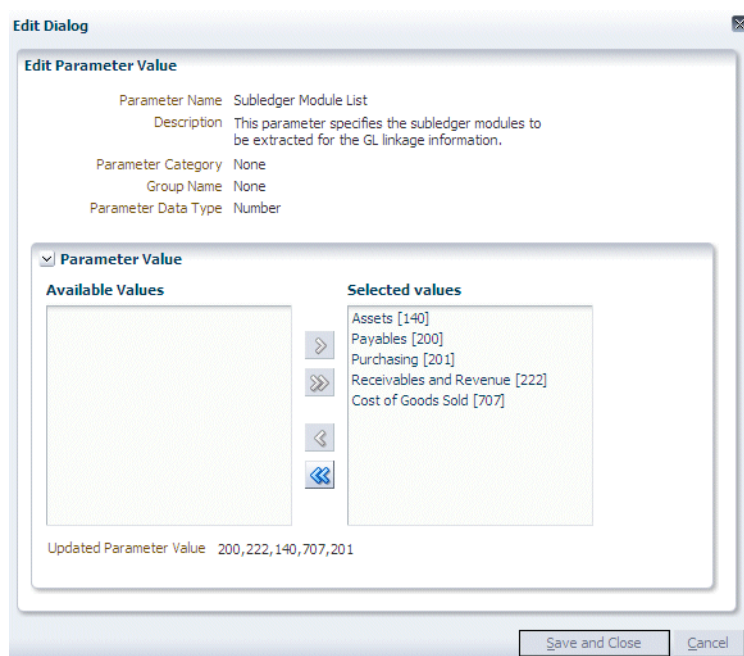
If the parameter is Non-Overridable at the Group level, then a Warning dialog displays a list of Fact Groups and Dimension Groups that will be affected if you click Yes to edit the value.
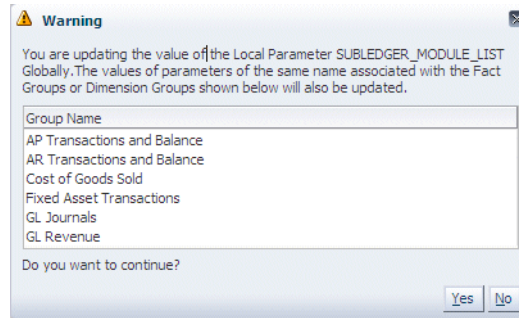
■ To update the parameter value for all Fact Groups and Dimension Groups in the **Group Specific Parameter values for** list, click Edit All to display the Edit Parameter Value dialog.
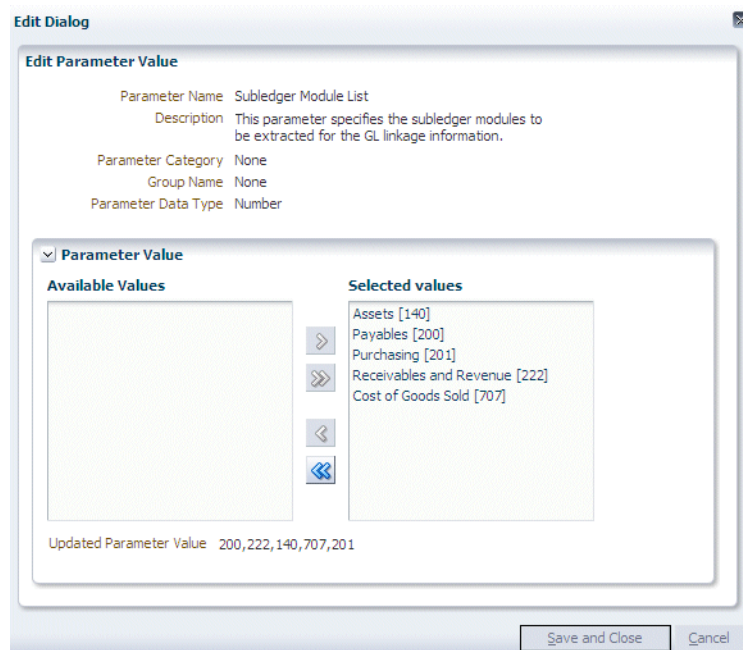
6.6 About Working With Reporting Parameters

Reporting Parameters are configuration values that specify how data is presented in Business Intelligence dashboards.

There are two categories of Reporting Parameter:

■ Global, which apply to all applications. These are displayed on the Global tab.

■ Application Specific, which apply to specific applications. These are displayed on the Application Specific tab.

You manage and edit Reporting Parameters using the "Manage Reporting Parameters: Global/Application Specific tab" (see the following screenshot).
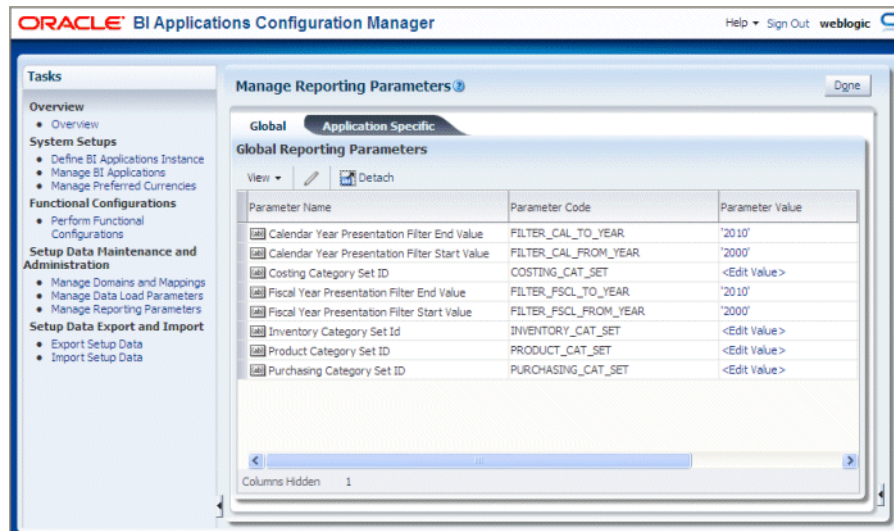
You display the "Manage Reporting Parameters: Global/Application Specific tab" by doing one of the following:

- In Oracle BI Applications Configuration Manager, select the **Manage Reporting Parameters** link in the Tasks pane.

- In Functional Setup Manager, select the **Go to Task** link for a Task that updates a Reporting Parameter.

To edit a Reporting Parameter, select the parameter in the parameter list, then either click the Edit icon, or click the value in the **Parameter Value** column.

6.7 How to Monitor Setup Data

During a functional configuration project, you monitor Setup Data to ensure that your Offerings are being correctly configured. For example, you might use the Alerts pane on the Overview page to check for unmapped domain values.

You can monitor set up data in the following ways:

- In Oracle BI Applications Configuration Manager, you can monitor the status of the underlying set up data values, as follows:

  - Use the pages and tabs in the **Setup Data Maintenance and Administration** area on the Tasks bar.

    From the Tasks bar, select one of the links in the Setup Data Maintenance and Administration area. For example, select the Data Load Parameters link to display the Data Load Parameters page.
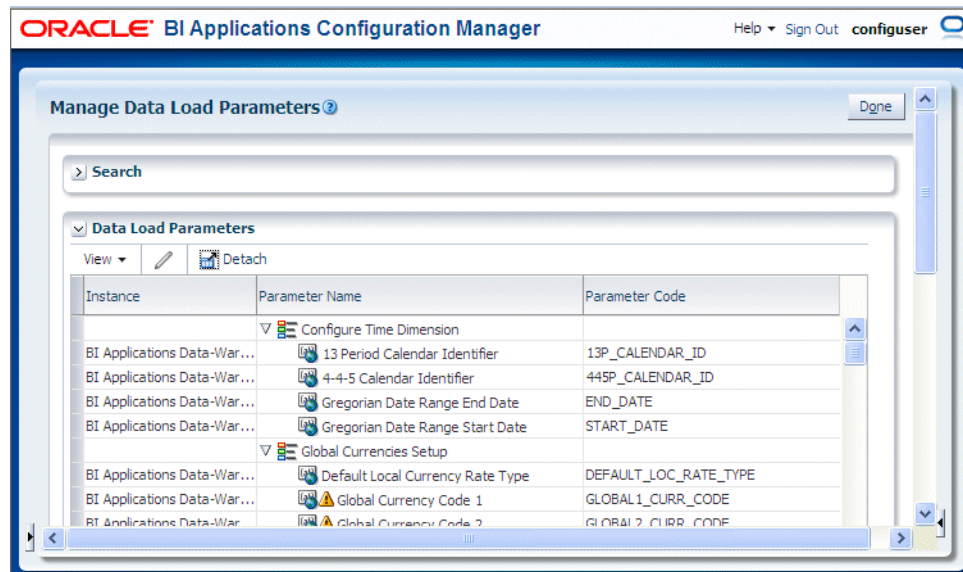
- Use the Alerts pane on the Overview page.

  Display the Overview page, and view the **Alerts** panes **Data Load Parameters with No Values** (see the following example screenshot), and **Domain Maps with Unmapped Source Members**.
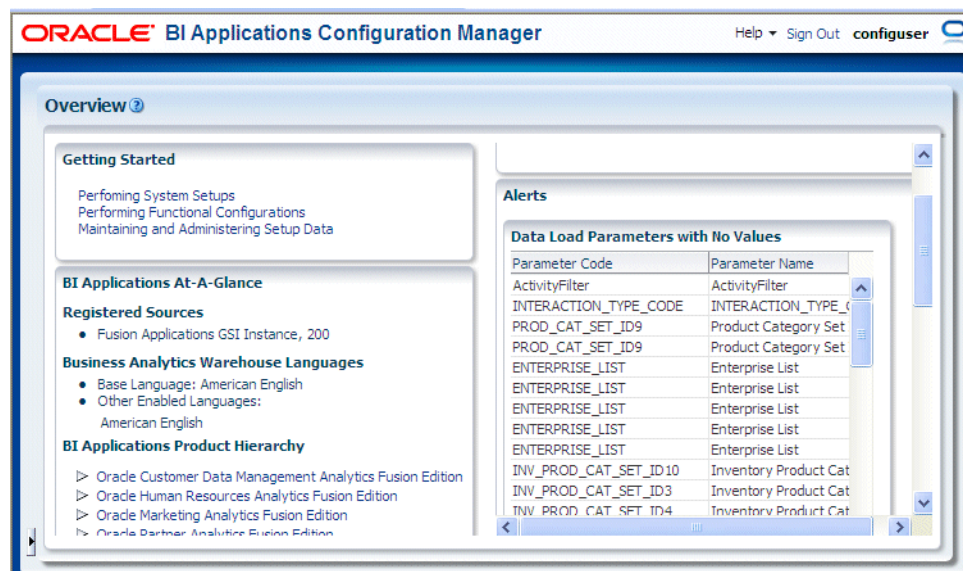
  The Alerts pane only displays alerts for 'enabled' Offerings.

6.8 How to Change Configuration Values Using Oracle BI Applications Configuration Manager

If you need to make configuration changes after an Implementation Project has been completed in Functional Setup Manager, you can use Oracle BI Applications Configuration Manager to update setup data. To change setup data values, you use the options in the **Setup Data Maintenance and Administration** area on the Tasks bar.

To change configuration values using Oracle BI Applications Configuration Manager:

1. In Oracle BI Applications Configuration Manager, use options in the **Setup Data Maintenance and Administration** area on the Tasks bar as follows:

   ■ To make changes to domain mappings, select **Manage Domains and Mappings** and display the tab for the domain type that you want to edit. To edit a Domain Mapping, select a domain in the Domain Mapping list and click the Edit icon in the Domain Member Mappings pane.
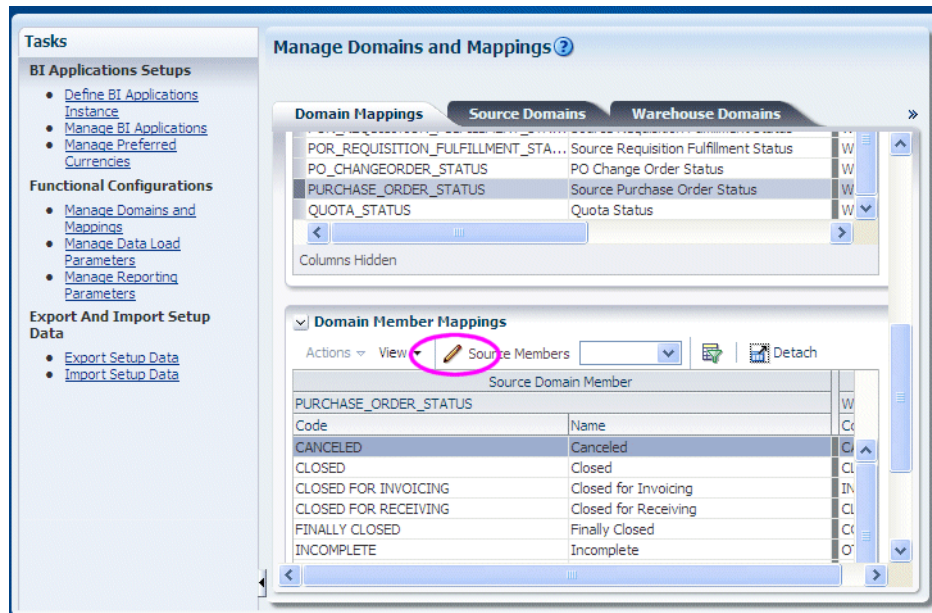
   ■ To make changes to data load parameters, select **Manage Data Load Parameters** and use the Search pane to locate the parameters that you want to edit. To edit the value of a Global Data Load Parameter, click the value in the **Global Parameter Value** column (or select the row then click the Edit icon) to display the "Edit Parameter Value dialog (for Data Load Parameters)".
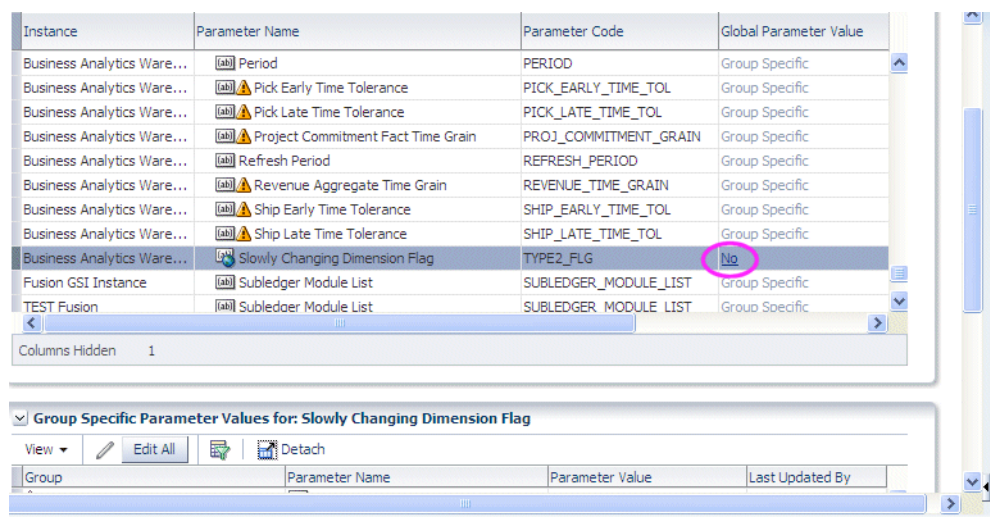
   For Application Specific parameters, select the parameter in the Data Load Parameters list, scroll down to the **Group Specific Parameter Values for** list, and click either the Edit icon or the value in the Parameter Value column (or select the parameter row and click the Edit icon).

- To make changes to reporting parameters, select **Manage Reporting Parameters**, and display either the Global tab or the Application Specific tab. To edit the value of a Reporting Parameter, click the value in the Parameter Value column (or select the row then click the Edit icon) to display the "Edit Parameter Value dialog (for Reporting Parameters)".

Tip: For information about locating unmapped domain values, see Section 6.9, "How to Locate Unmapped Domain Values in Oracle BI Applications Configuration Manager".

6.9 How to Locate Unmapped Domain Values in Oracle BI Applications Configuration Manager

You locate unmapped Domain Values to enable you to make sure that you have mapped all of your source system domain values.

Tip: You can also use the Alerts pane on the Overview page in Oracle BI Applications Configuration Manager to identify target domains with one or more unmapped source members. Display the Overview page, and navigate to the **Domain Maps with Unmapped Source Members** pane.

How to locate unmapped domain values:

1. Start Oracle BI Applications Configuration Manager.

2. Do one of the following:

- From the **Tasks** bar, select **Manage Domains and Mappings** to display the Manage Domains and Mappings dialog.

  Display the "Manage Domains and Mappings: Domain Mappings tab".

  In the Domain Member Mappings pane, select Unmapped from the **Source Members** list.
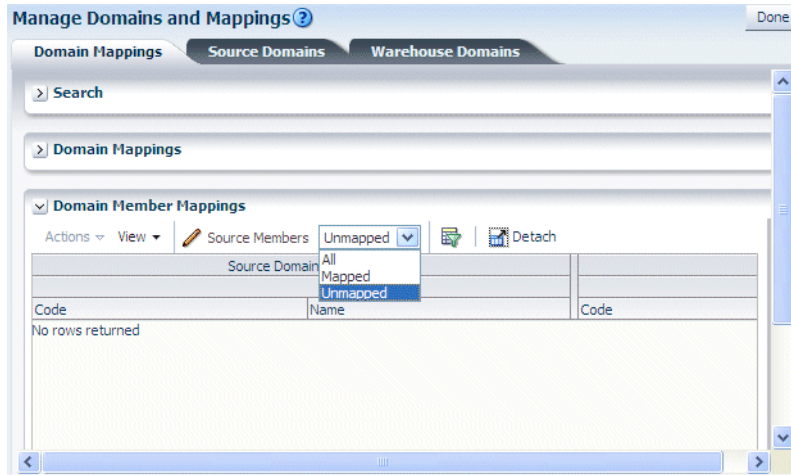
- From the **Tasks** bar, select **Manage Domains and Mappings** to display the Manage Domains and Mappings dialog.

  Display the "Manage Domains and Mappings: Warehouse Domain Hierarchies tab".

  In the Domain Member Mappings pane, select Unmapped from the **Source Members** list.
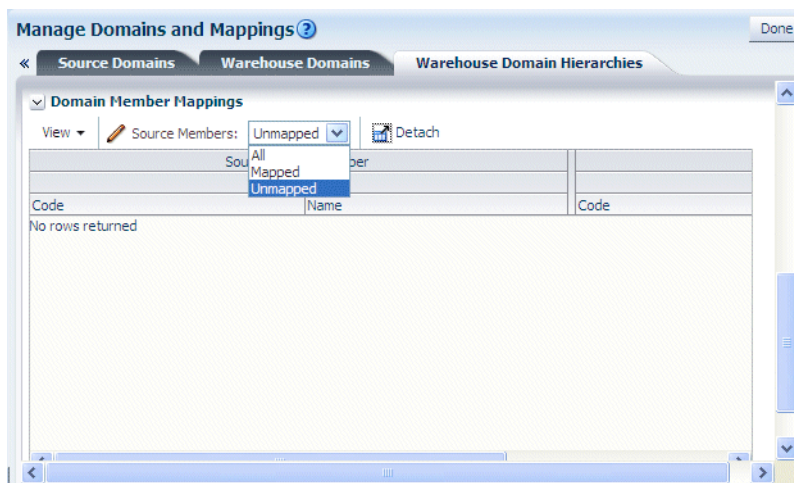
6.10 How to Locate Data Load Parameters with no Values in Oracle BI Applications Configuration Manager

You locate Data Load Parameters with no values to check all Tasks in Functional Setup Manager relating to Data Load Parameters have been completed correctly.

How to locate Data Load Parameters with no values:

1. Start Oracle BI Applications Configuration Manager.

2. Do one of the following:

- From the **Overview** page, use the **Data Load Parameters with no Values** list in the Alerts pane.

- From the **Tasks** bar, select **Manage Data Load Parameters** to display the "Manage Data Load Parameters dialog", which enables you to view and edit parameters.

6.11 About Adding Target Domain Members

You add Target Domain Members to extend Oracle Business Analytics Warehouse. For example, you might want to create a new salary category called 'Range 5' so that you can map salary values to this new category.

- Section 6.11.1, "How to add a Target Domain Member"

6.11.1 How to add a Target Domain Member

You add a target domain member to extend Oracle Business Analytics Warehouse.

To add a Target Domain Member:

1. Display the Warehouse Domains tab.

   In Functional Setup Manager, the Warehouse Domain tab is displayed when you perform a Task that edits a Warehouse Domain value.

   In Oracle BI Applications Configuration Manager, from the Tasks bar, select Manage Domains and Mappings to display the Manage Domains and Mappings dialog.

2. In the Warehouse Members pane, click Add Warehouse Domain Member, to display the "Add Warehouse Domain Member/Add Target Domain Member dialog".

3. Use the "Add Warehouse Domain Member/Add Target Domain Member dialog" to specify the details.

4. Click OK to save the details and close the dialog.

6.12 Viewing Domain Hierarchies

To view Domain Hierarchies, select the Manage Domains and Mappings link on the Tasks bar, then display the Warehouse Domain Hierarchies tab.
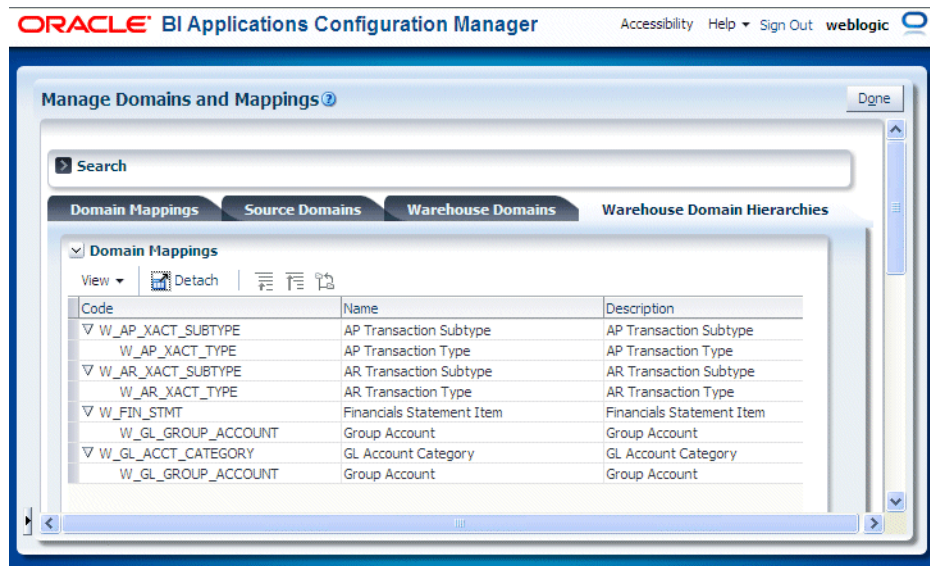
Notes

- Domain Hierarchies are displayed in inverted format, that is in the following format:

  <Child 1>\

      <Child *n*>\

          <Parent>.

  For example, in the screenshot above, the child node AP Transaction Subtype is shown above and to the left of the parent node AP Transaction Type.

- Domain Hierarchies are read-only. However, you can change the domain mappings.

- For field level help for the Warehouse Domains Hierarchies dialog, see Section C.21, "Manage Domains and Mappings: Warehouse Domain Hierarchies tab".

6.13 About Exporting and Importing Setup Data for Oracle BI Applications Configuration Manager

You export and import Setup Data for Oracle BI Applications Configuration Manager to:

- make a backup of your configuration settings for security purposes. For example, you might keep a record of the configuration changes that you have made.

- migrate the Setup Data for Oracle BI Applications Configuration Manager from one environment to another environment. For example, you might move the configuration changes that you have made from a Test environment to a Production environment.

6.13.1 What Data is Exported?

When you export Setup Data, you export the changes that you have made to the values of the following objects:

- Data Load Parameters

- Domains and Mappings

- Reporting Parameters

Unchanged configuration values are not exported. In other words, if you only change the value of DEFAULT_CURRENCY from USD to Euro and then you export your data, then the export ZIP file that is produced will only contain columns for 'DEFAULT_CURRENCY=Euro'.

Note: Other Oracle BI Applications Configuration Manager data (for example, Source System configuration settings, Oracle Business Analytics Warehouse configuration settings) is not included in export files. In other words, before you import on the target system, you must first re-specify the Source System and Oracle Business Analytics Warehouse.

6.13.2 What Data is Imported?

When you import Setup Data from a ZIP file, you import whatever configuration changes were exported to that ZIP file. For example, if you first export only Reporting Parameters to a ZIP file, then you import only the changes that were made to Reporting Parameters.

6.13.3 About Migrating Setup Data

To migrate Setup Data, you do the following:

- In the source environment, export the Oracle BI Applications Configuration Manager Setup Data as a ZIP file. For more information about exporting Setup Data, see Section 6.13.5, "How to Export Setup Data".

- Copy the ZIP file from the source environment to the target environment.

- In the target environment, import the Oracle BI Applications Configuration Manager Setup Data from the ZIP file. For more information about importing Setup Data, see Section 6.13.6, "How to Import Setup Data".

  Note: The data source number in the target system must be the same as the data source number in the source system.

6.13.4 About Backing-up Setup Data

To back-up Setup Data, you do the following:

- In the source environment, export the Oracle BI Applications Configuration Manager Setup Data as a ZIP file. For more information about exporting Setup Data, see Section 6.13.5, "How to Export Setup Data".

- Store the ZIP file in a secure location.

- In the target environment, make sure that you have installed Oracle BI Applications Configuration Manager, which installs the default setup data. **Note**: The data source number in the target system must be the same as the data source number in the source system.

- To restore the backed-up data, you import the Oracle BI Applications Configuration Manager Setup Data from the ZIP file. For more information about importing Setup Data, see Section 6.13.6, "How to Import Setup Data".

6.13.5 How to Export Setup Data

You export Setup Data to back-up or migrate a Oracle BI Applications Configuration Manager environment.

To export Setup Data:

1. In the source environment, start Oracle BI Applications Configuration Manager.

2. From the **Tasks** bar, select Export Setup Data to display the "Export Setup Data dialog".
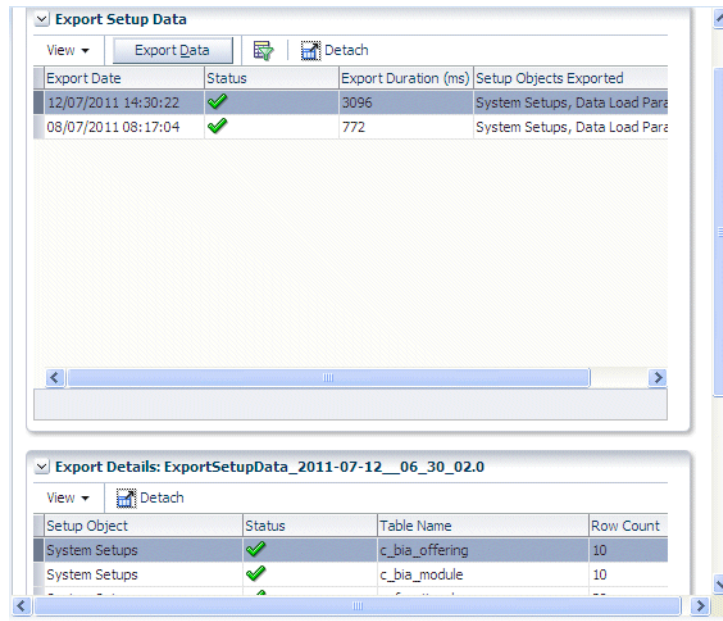
3. Click Export Data to display the "Export Data dialog".

4. Use the "Export Data dialog" to specify the setup objects that you want to export.
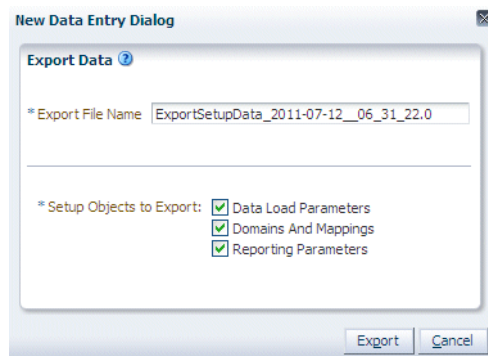
5. Click Export.

6. When the File Download dialog is displayed, click **Save** to save the ZIP file to a location that you specify.

    If you first click **Open** at the File Download dialog to open the ZIP file, then make sure that you save the ZIP file in the ZIP program that you are using.

The Export Setup Data table is updated with the export details.

6.13.6 How to Import Setup Data

You import Setup Data to restore or migrate Setup Data. Before you import Setup Data, you must copy the previously exported ZIP file to a file location that is accessible from the machine that is running the Oracle BI Applications Configuration Manager client.

Note: When you import Setup Data, the data source number in the target system must be the same as the data source number in the source system.

To import Setup Data:

1. In the source environment, start Oracle BI Applications Configuration Manager.

2. From the **Tasks** bar, select Import Setup Data to display the "Import Setup Data dialog".
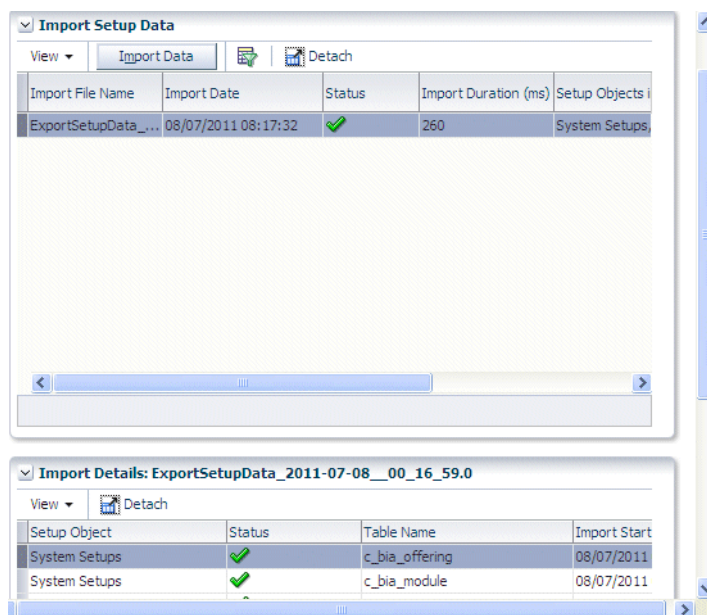
3. Click Import Data to display the "Import Data dialog".
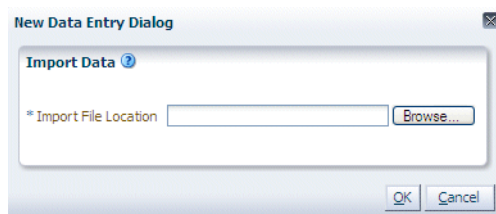
4. Use the "Import Data dialog" to specify the location and name of the ZIP file that contains the configuration changes that you want to import.

5. Click OK.

   The Setup Data is imported from the ZIP file, and the Import table is updated with the details.

Part III

Appendices

Part III contains reference information that is used in the Oracle BI Applications Configuration Manager Help System and Functional Setup Manager, and includes the following sections:

- Appendix A, "Supporting Tasks"
- Appendix B, "Functional Configuration Task Reference"
- Appendix C, "Dialog Reference for Oracle BI Applications Configuration Manager"

A

Supporting Tasks

This section contains supporting information and tasks for setting up Oracle BI Applications. These tasks are used by the setup steps in Chapter 3, "Setting Up Oracle Business Intelligence Applications".

This section contains the following topics:

- Section A.1, "How to Log Into Informatica Administrator"
- Section A.2, "How to Start Oracle BI Applications Configuration Manager"
- Section A.3, "Merging Repositories for Oracle BI Applications Projects"
- Section A.4, "How to Configure DAC Server in a High Availability Environment"
- Section A.5, "How to Verify Duty Roles required for Oracle BI Applications"
- Section A.6, "About Warehouse Component Deployment Assistant (WCA)"
- Section A.7, "How to Restart DAC Server"

A.1 How to Log Into Informatica Administrator

Informatica PowerCenter Administrator is hosted from the machine that has installed the gateway node for the PowerCenter domain. For a single machine install of PowerCenter Services as described in this chapter, PowerCenter Administrator is installed along with PowerCenter Services.

To log into PowerCenter Administrator

1. In a supported Web browser, access the following URL:

   http://<*gateway host*>:<*domain port*>

   Where:

   - <*gateway host*> is the name of the machine on which the gateway node has been configured. For a single-machine installation of PowerCenter Services it is the name of the machine on which PowerCenter Services has been installed.

     For example, http://myhost.

   - <*domain port*> is the port number for the gateway. The default port number is 6008.

2. In the login page, enter the Administrator user name and password that was specified when you created the domain during installation of PowerCenter Services (for example, Administrator\Administrator), then click Log In to display the main page.

In the following example screenshot, the Domain Navigator displays a list of components, including a Repository Service named BIA_REP_SERVICE, and an Integration Service named BIA_INT_SERVICE. The Overview page displays the status of the components.
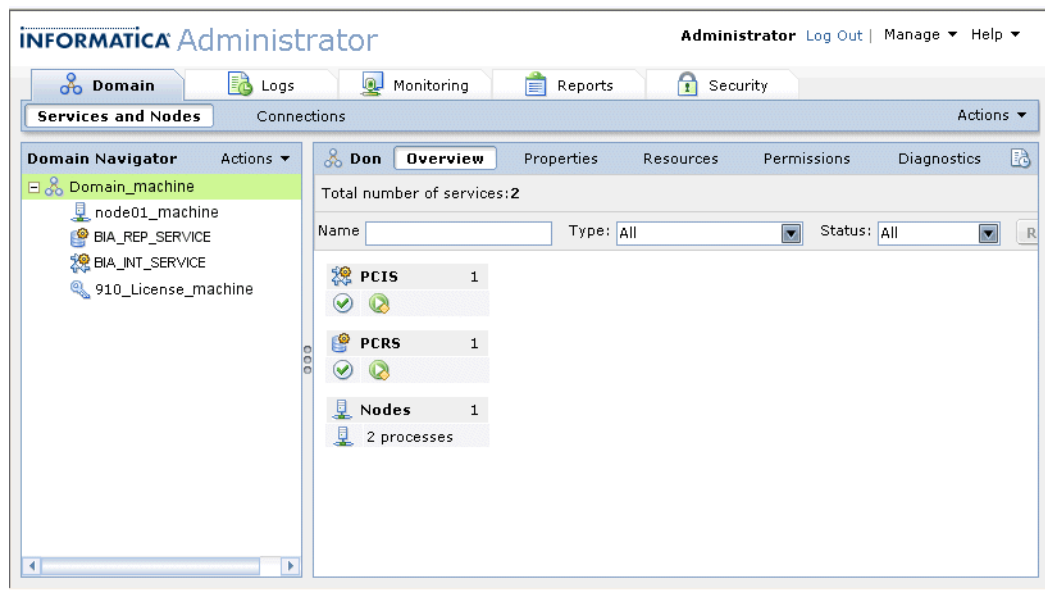
A.2 How to Start Oracle BI Applications Configuration Manager

You use Oracle BI Applications Configuration Manager to set up and manage several aspects of the Oracle BI Applications environment, and to manage and monitor Functional Configurations for Oracle BI Applications Offerings.

To start Oracle BI Applications Configuration Manager:

1. In a Web browser, enter the Oracle BI Applications Configuration Manager URL which your System Administrator provides you.

   For example:

   http://myexample.com:7001/biacm/

   The Oracle BI Applications Configuration Manager Login dialog is displayed.

2. In the Login screen, log in with your user credentials.

   Note: Your User Login must be associated to the appropriate Duty Role. For more information see Section 4.4, "About Users and Roles in Oracle BI Applications Configuration Manager". If you do not have access to Oracle BI Applications Configuration Manager or have incorrect access, then contact your Security Administrator.

   The main Oracle BI Applications Configuration Manager page is displayed.
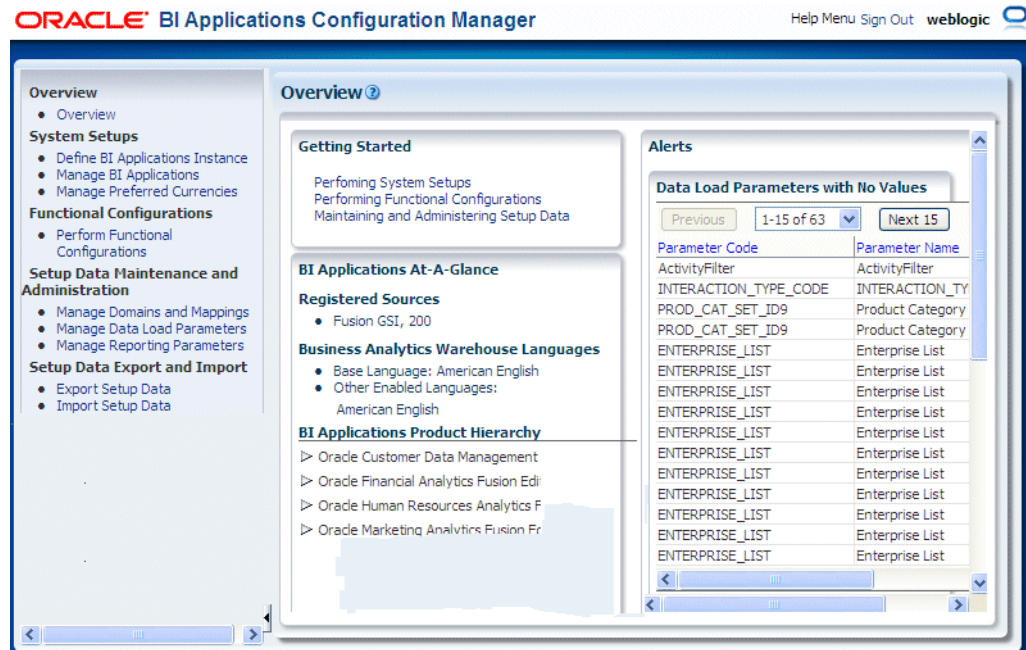
Notes

- Use the Tasks bar on the left-hand side of the Welcome page to navigate the application.

  The options that are displayed in the Tasks bar are dependent on the privileges of the user name that you log in with.

- To display Help for the Overview page, click Help, then Configuration Manager Help. You can also display context-sensitive Help for a dialog by clicking the question mark icon displayed next to the dialog name or a field name.

- To log out of Oracle BI Applications Configuration Manager, select the **Sign out** link in the top right hand corner.

A.3 Merging Repositories for Oracle BI Applications Projects

This section explains how to merge two BI repositories (that is, RPD files) for Oracle BI Applications.

To merge repositories for Oracle BI Applications projects:

1. In the Administration Tool, open in offline mode the Oracle BI repository.

2. On the Administration Tool menu bar, click **File** and then **Merge**.

3. In the Select Original Repository dialog box, select the repository (for example, **OracleBIApps.rpd**).

4. Type the password for the original repository, and then click **OK**.

5. Click **Select** for the Modified Repository field. The Select Modified Repository dialog box opens.

6. In the Select Modified Repository dialog box, select the repository that contains the customizations you made to the previous RPD file (for example, OracleBIAppsOTBIonly.rpd).

7. Click **Open**, type the password for the previously customized RPD file, and then click **OK**.

8. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.

9. To locate subsequent rows with empty Decision fields, click the **Decision** header cell. After all rows have a value in the Decision field, the Merge button becomes enabled.

10. Click **Merge**. A message appears after the merge concludes successfully.

11. On the File menu, click **Save As**, and save the current repository using a new name (for example, OracleBIAppsFinal.rpd).

A.4 How to Configure DAC Server in a High Availability Environment

The section describes how to configure DAC Server in a high-availability environment.

In Oracle BI Applications 11*g* Release 1 (11.1.1), the DAC Server is an enterprise application deployed into the Managed Server of the BI Domain. DAC Client is used to configure the URL of the DAC Server in the DAC Repository.

In a simple deployment (that is, Oracle BI Applications deployed into a single BI domain with the WebLogic Administration Server and Managed Server in the same domain home), the DAC Server URL is configured automatically. In a high-availability deployment, the DAC Server URL must be changed manually to the Oracle HTTP Server URL by following the steps below:

To configure the DAC Server URL:

1. Start DAC Client by navigating to the \dac directory and double-clicking the startclient.bat file. The Login dialog window opens.

2. In the Login dialog window, select the **Configure**.

3. In the Configuring dialog, select **Create Connection**, and then click **Next**.

4. Enter the appropriate connection information:

   - **Name:** Enter a unique name for the connection to the DAC repository.

   - **Connection type:** Choose **Oracle (Thin)**.

   - **Service Name:** Enter the service name for the DAC Repository (*prefix*_DAC).

   - **Database Host:** Enter the name of the computer that hosts the DAC Repository database.

   - **Database Port:** Enter the port number for the database.

5. Click in the **Authentication File** field, and then do the following:

   a. In the Authentication File dialog window, click **Choose existing authentication file**, and then click **OK**.

   b. Navigate to the appropriate directory, select the cwallet.sso file that you copied from the DAC Server computer, and click **OK**.

      If you installed DAC Client on a different machine to the Fusion Applications installed DAC Server, then you first need to copy cwallet.sso file across from server machine to dac client machine.

   c. In the Configuring dialog window, click **Test Connection** to confirm the connection works.

    d. Click **Apply**.

    e. Click **Finish**.

6. In the drop-down list of the Login dialog window, select the Connection that you created.

7. Enter any values in the User Name and Password fields, as the DAC repository does not contain any user names or passwords. A user account will be created with the user name and password you enter. This user account is automatically assigned the Administrator role.

8. Select **DAC** as the Authentication Type.

9. Click **Login**.

10. In DAC Client, navigate to **Tools** menu, click **DAC Server Management**, and then click **Repository Configuration**.

11. In the Repository Configuration dialog window, enter the following information:

    ■ **Mode:** Web

    ■ **URL:** The URL for the DAC Server enterprise application deployed in WebLogic Server. For example, http://*WLS_host*:9704/DACServer.

12. Click **Save**.

> **Note:** For all subsequent logins to DAC Client, select the Connection and choose the FMW Authentication Type. Users will be authenticated against the LDAP system and must be granted the appropriate Oracle BI Applications duty role.

A.5 How to Verify Duty Roles required for Oracle BI Applications

This section explains how to verify that Duty Roles for Oracle BI Applications are correctly mapped to Fusion Applications Enterprise Roles.

A list of required Enterprise Roles and Duty Roles for Oracle BI Applications is published on My Oracle Support (ID 1333454.1.) as a spreadsheet named 'Oracle Business Intelligence Applications Duty Role Assignments for Fusion Applications'.

The spreadsheet contains a sheet for each Oracle BI Applications Offering; for example, Oracle Financial Analytics, Oracle HR Analytics, and so on. Each sheet contains the following information:

■ The Fusion Applications Enterprise Job Roles in column A.

■ The associated Duty Roles in column B.

■ The Presentation Catalog areas mapped to the Duty Role in Column C.

■ The other columns show information, such as the modules to which the roles apply.

The following screenshot shows a section from the Finance sheet, which contains Duty Roles for Oracle Financial Analytics.

To verify that Duty Roles are mapped to appropriate Fusion Applications Enterprise Roles, do the following:

1. Download the spreadsheet named 'Oracle Business Intelligence Applications Duty Role Assignments for Fusion Applications', which is available on My Oracle Review as ID 1333454.1.

2. Use Oracle APM to verify that Duty Roles are mapped to the appropriate Enterprise Roles, as follows:

    a. In Oracle APM, navigate to the 'obi' Application and use the Search options to locate Duty Roles prefixed with 'OBIA_'.
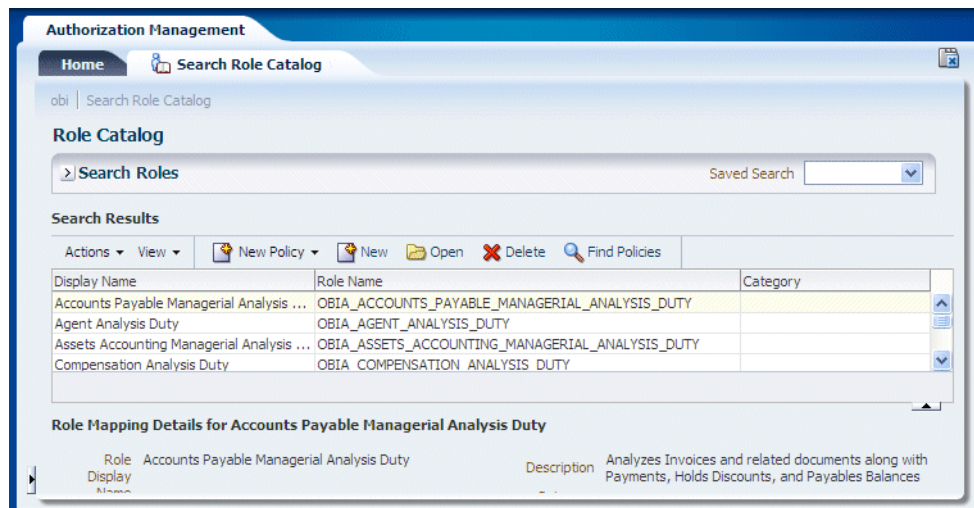
    b. Select each Duty Role, then click Open to display the *<Application>* | Application Role dialog.

    c. Display the External Role Mapping tab, and check that the role list contains the appropriate Enterprise Roles.
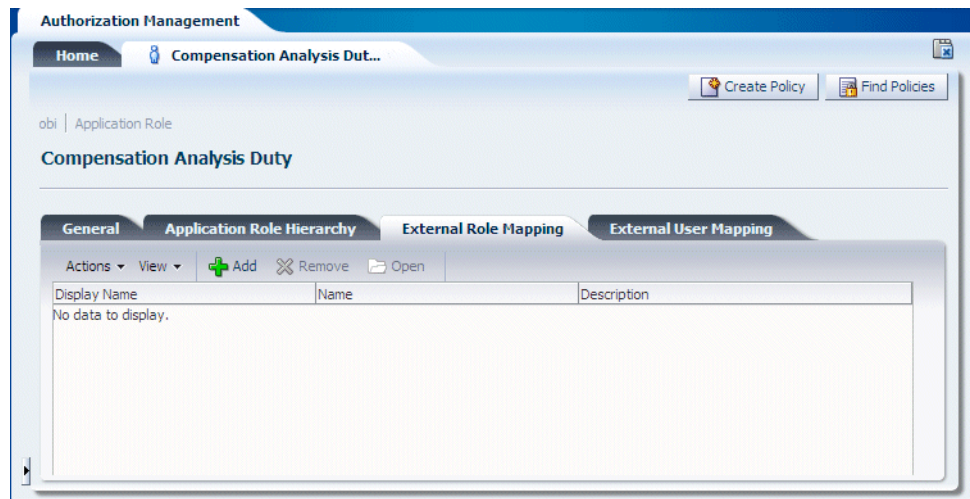
Use the spreadsheet 'Oracle Business Intelligence Applications Duty Role Assignments for Fusion Applications' to determine which Enterprise Roles should be displayed in the role list.

3. If you need to re-map Fusion Applications Enterprise Roles to Duty Roles, do the following:

a. In Oracle APM, navigate to the 'obi' Application and use the Search options to locate and select the Duty Role in the role list.

b. Click Open to display the *<Application>* | Application Role dialog.

c. Display the External Role Mapping tab.
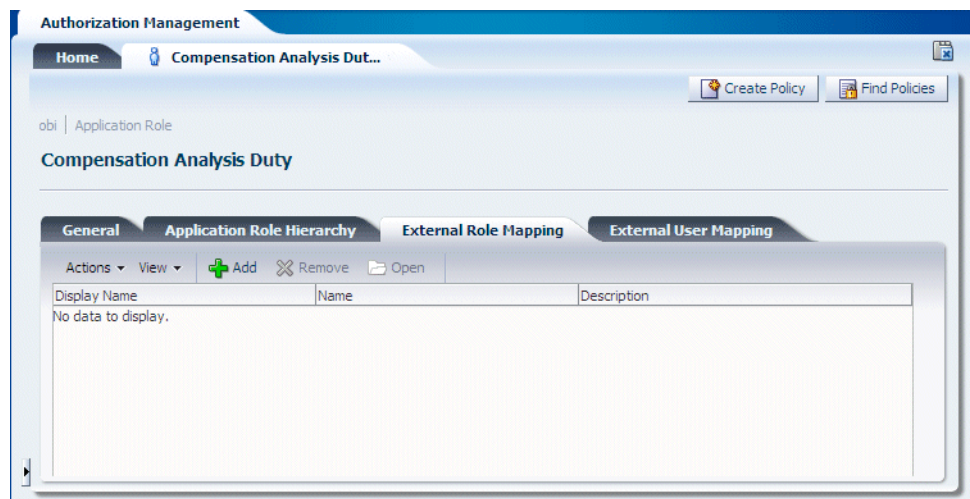
d. Use the External Role Mapping tab to search for and select the Fusion Applications Enterprise Roles that must be mapped to that Duty Role.

Click Add to display the External Role Search dialog, which enables you to locate Enterprise Roles in the LDAP directory.
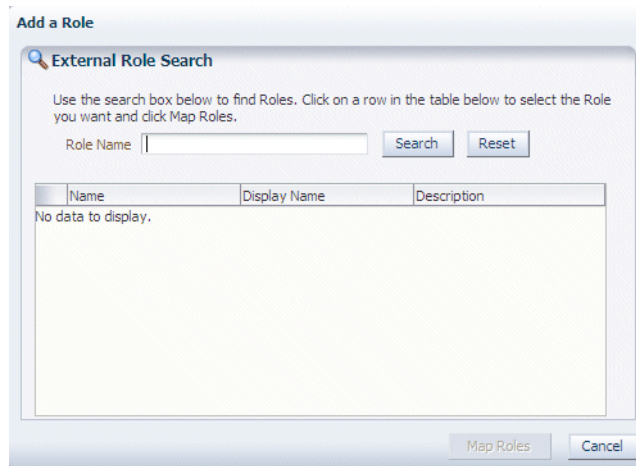
> For example, the OBIA_WORKFORCE_DEPLOYMENT_ANALYSIS_DUTY
> Duty Role must be mapped to the PER_LINE_MANAGER_ABSTRACT Job
> Role and the PER_HUMAN_RESOURCES_VP_JOB Job Role.

e. Click Map Roles to map selected Enterprise Roles to the Duty Role, and return
to the External Role Mapping tab.

For more information about using Oracle APM to manage Duty Roles and
Enterprise Roles, refer to *Oracle Fusion Middleware Oracle Authorization Policy
Manager Administrator's Guide (Oracle Fusion Applications Edition)*.

A.6 About Warehouse Component Deployment Assistant (WCA)

Warehouse Component Deployment Assistant (WCA) is a configuration script that
you use in UNIX environments to set up a new Oracle BI Applications Release 11.1.1.6
environment after Oracle BI Applications has been installed using the Oracle Fusion
Applications installer. WCA is not supported on Windows platforms.

A.6.1 Overview of WCA

WCA is used as part of the setup process for Oracle BI Applications. For information
about how to use WCA to configure an Oracle BI Applications environment, see
Section 3.4.8.1, "Setup Step: Configure Oracle BI Applications components using
Warehouse Component Deployment Assistant (WCA) in a UNIX Environment.",
(which is performed as part of Section 3.1.1, "High-level Tasks for Setting Up Oracle BI
Applications.").

Warehouse Component Deployment Assistant does the following:

- (Optional) Installs Informatica PowerCenter Services V9.1 HotFix 2.

- (Optional) Configures Informatica PowerCenter Services related components.

- Configures DAC Properties, registers Informatica, and sets physical connections.

- Registers Sources in Oracle BI Applications Configuration Manager.

- Extends and configures the BI Domain.

- Configures RPD Connection Pools.

- Enables User Currencies.

- Creates physical connections for Fusion Applications pillars, DW, and Oracle BI
Applications Configuration Manager.

- Creates an Integration Service for Informatica PowerCenter.

  Note: This configuration step requires Informatica PowerCenter Services and DAC Server to be installed on the machine that hosts the BI Domain, and the CONFIGURE_INFA value set to 'y' in the wca_params.properties configuration file. If not, you must manually configure this component as described in Section 3.4.9.2, "Setup Step: Creating the Informatica Repository Service".

- Restores the Informatica repository into the Repository Service.

  Note: This configuration step requires Informatica PowerCenter Services and DAC Server to be installed on the machine that hosts the BI Domain, and the CONFIGURE_INFA value set to 'y' in the wca_params.properties configuration file. If not, you must manually configure this component as described in Section 3.4.9.4, "Setup Step: Load the Prebuilt Informatica Repository".

- Creates a Repository Service for Informatica PowerCenter.

  Note: This configuration step requires Informatica PowerCenter Services and DAC Server to be installed on the machine that hosts the BI Domain, and the CONFIGURE_INFA value set to 'y' in the wca_params.properties configuration file. If not, you must manually configure this component as described in Section 3.4.9.3, "Setup Step: Creating the Informatica Integration Service".

- Copy source files and sets custom properties.

  Note: This configuration step requires Informatica PowerCenter Services and DAC Server to be installed on the machine that hosts the BI Domain, and the CONFIGURE_INFA value set to 'y' in the wca_params.properties configuration file. If not, you must manually configure this component as described in Section 3.4.9.5, "Setup Step: Copying Source Files to the Informatica PowerCenter Services machine".

- Creates an Administrator user for the Informatica repository.

  Note: This configuration step requires Informatica PowerCenter Services and DAC Server to be installed on the machine that hosts the BI Domain, and the CONFIGURE_INFA value set to 'y' in the wca_params.properties configuration file. If not, you must manually configure this component as described in Section 3.4.12.3, "Setup Step: Creating the Repository Administrator User in the Native Security Domain".

- Sets Informatica PowerCenter properties, and creates relational connections in Informatica.

  Note: This configuration step requires Informatica PowerCenter Services and DAC Server to be installed on the machine that hosts the BI Domain, and the CONFIGURE_INFA value set to 'y' in the wca_params.properties configuration file. If not, you must manually configure this component as described in Section 3.4.15.3, and Section 3.4.15.4, "Setup Step: Setting the SiebelUnicodeDB Property in Informatica Integration Services".

A.6.2 Notes on using WCA Configuration Parameters in wca_param.properties

- If you have Informatica PowerCenter Services and DAC Server installed on the machine that hosts the BI Domain, then WCA can configure all components. If you do not have Informatica PowerCenter Services and DAC Server installed on the machine that hosts the BI Domain, then WCA cannot configure Informatica related components, and you must manually configure them, as described in Appendix A.6.1, "Overview of WCA".

- Specify the value of the INFA_INSTALL_UNZIP_LOC parameter as the directory where you extracted the Informatica PowerCenter Services JAR file. For example, INFA_INSTALL_UNZIP_LOC=/scratch/myfiles/infa9. This directory should contain the extracted install.sh file and associated files.

- To install Informatica PowerCenter Services, set INSTALL_INFA to y, and specify values for the following parameters:

```
INFA_DB_HOST
INFA_DB_PORT
INFA_DB_SERVICE
INFA_DOM_DB_PASSWRD
INFA_DOM_DB_USER
INFA_DOM_NAME
INFA_DOM_PASSWRD
INFA_DOM_PORT
INFA_DOM_USER
INFA_HOME
INFA_INSTALL_UNZIP_LOC
INFA_LICENSE_KEY
INFA_NODENAME
INSTALL_INFA
```

If INSTALL_INFA is set to n, then the parameters listed above and any default values are ignored.

- If you are deploying Informatica PowerCenter Services on the same machine as the BI Server, set CONFIGURE_INFA to y, and specify values for the following parameters:

```
CONFIGURE_INFA
INFA_DB_HOST
INFA_DB_PORT
INFA_DB_SERVICE
INFA_DB_TNS_ENTRY
INFA_DB_TYPE
INFA_DOM_DB_PASSWRD
INFA_DOM_DB_USER
INFA_DOM_HTTPS_PORT
INFA_DOM_NAME
INFA_DOM_PASSWRD
INFA_DOM_PORT
INFA_DOM_USER
INFA_DOMAIN_FILE
INFA_HOME
INFA_IS_NAME
INFA_NODENAME
INFA_PCS_HOST
INFA_REP_DB_PASSWRD
INFA_REP_DB_USER
INFA_REP_PASSWRD
INFA_REP_USER
INFA_RS_NAME
```

If CONFIGURE_INFA is set to n, then the parameters listed above and any default values are ignored.

A.7 How to Restart DAC Server

To restart DAC Server:

1. Navigate to the DAC domain home, for example, /instance/domains/*<Domain Host>*/BIDomain/dac.

2. Execute the automationUtils.sh cmdCredentials command with the cwallet.sso file and the administrator username for WebLogic Server Administrator Console. For example:

```
/instance/domains/<Domain
Host>/BIDomain/dac/conf/security/repository/cwallet.sso FAadmin
```

   Note: When prompted for a password, specify the administrator password for WebLogic Server Administrator Console.

3. Execute the following command:

```
sh dacCmdLine.sh StartServer
```

B

Functional Configuration Task Reference

This section contains reference information about the functional configuration Tasks that you use in FSM to configure Oracle BI Applications Offerings. Use this section to find out which functional configuration Tasks are available in FSM for each Oracle BI Applications Offering.

Functional configuration Tasks in Section B.1, "Lists of Functional Configuration Tasks For Each Offering" are listed by name only; for more information about these Tasks, refer to FSM. Information-only tasks (known as Informational Tasks in FSM) in Section B.2, "Informational Task Reference" are included with the text that is displayed in FSM when you click **Go to Task**.

This chapter contains the following sections:

- Section B.1, "Lists of Functional Configuration Tasks For Each Offering"
- Section B.2, "Informational Task Reference"

> **Note:** This chapter lists the names of Tasks that are available in FSM for each Offering. You must use FSM to see the Task details for an Offering. The chapter also includes Information-only Tasks.

B.1 Lists of Functional Configuration Tasks For Each Offering

This section lists the Functional Configuration Tasks for each Offering in Oracle Business Intelligence Applications, and contains the following sections:

- Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions"
- Section B.1.2, "List of Functional Configuration Tasks for Oracle Customer Data Management Analytics"
- Section B.1.3, "List of Functional Configuration Tasks for Oracle Financial Analytics"
- Section B.1.4, "List of Functional Configuration Tasks for Human Resources Analytics"
- Section B.1.5, "List of Functional Configuration Tasks for Oracle Marketing Analytics"
- Section B.1.6, "List of Functional Configuration Tasks for Oracle Partner Analytics"
- Section B.1.7, "List of Functional Configuration Tasks for Oracle Procurement and Spend Analytics"

-

-

-

-

B.1.1 List of Functional Configuration Tasks for Common Areas and Dimensions

This section lists Functional Configuration Tasks that apply to multiple Offerings.

Common Areas and Dimensions

How to Perform System Setups and Post Install Tasks for BI Applications

Configure Data Load Parameters for File Based Calendars

Configure Enterprise List

Configure Global Currencies

Configure Initial Extract Date

Configure Reporting Parameters for Year Prompting

Configure Slowly Changing Dimensions

Define Enterprise Calendar

How to Include and Exclude Multiple Calendar Support for Subject Areas in DAC

How to Reload the Time Dimension Tables After the Data Warehouse Is Loaded

Specify Gregorian Calendar Date Range

B.1.2 List of Functional Configuration Tasks for Oracle Customer Data Management Analytics

This section lists Functional Configuration Tasks for Oracle Customer Data Management Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Customer Data Management Analytics

Overview of Customer Data Management Analytics

Configure Initial Extract Date

Configure Enterprise List

Specify Time in Days before an Account needs an Activity

Specify Time in Days before an Account needs a Contact

Specify Time in Days before a Contact needs an Activity

Specify Order of First Name and Last Name in Position Hierarchy

Specify Interaction Type Code

Specify the Filter to be Applied on Interaction

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Party Dimension

ETL Notes and Additional Information for Customer Data Management Analytics

B.1.3 List of Functional Configuration Tasks for Oracle Financial Analytics

This section lists Functional Configuration Tasks for Oracle Financial Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Financial Analytics
Overview of Financial Analytics

Configure Global Currencies

Specify Gregorian Calendar Date Range

Configure Data Load Parameters for File Based Calendars

Define Enterprise Calendar

How to Reload the Time Dimension Tables After the Data Warehouse Is Loaded

Configure Slowly Changing Dimensions

Configure Reporting Parameters for Year Prompting

Configure Initial Extract Date

Configure Enterprise List

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Configure Data Load Parameters for Product Inventory Catalog

Configure Reporting Parameters for Product Inventory Catalog

Configure Subledger Module to be extracted for GL Linkage information

How to Implement GL Segment and GL Account Dimensions

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for GL Account Dimension

Manage Domains and Member Mappings for Payment Method Dimension

Manage Domains and Member Mappings for Business Location Dimension

Manage Domains and Member Mappings for Customer Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Time Dimension

Manage Domains and Member Mappings for Supplier Account Dimension

Manage Domains and Member Mappings for Project Dimension Group

Manage Domains and Member Mappings for Project Resource Dimension Group

Manage Domains and Member Mappings for Project Contract Dimension Group

Configure the number of historical monthly snapshots for AP Aging

Manage Domains and Member Mappings for AP Hold Release Reason Dimension

Manage Domains and Member Mappings for AP Transaction Source Dimension

Manage Domains and Member Mappings for AP Transaction Type Dimension

Manage Domains and Member Mappings for AP Transactions and Balance

Configure the number of historical monthly snapshots for AR Aging

Manage Domains and Member Mappings for AR Transaction Type Dimension

Manage Domains and Member Mappings for AR Transactions and Balance Fact

Manage Domains and Member Mappings for Fixed Asset Status Dimension

Manage Domains and Member Mappings for Fixed Asset Transaction Type Dimension

How to Implement Asset Category and Asset Location Dimensions

Specify the Ledger or Set of Books for which General Ledger Data is Extracted

How to Set up GL Segments Which Need to be Aggregated for GL Balances

How to Set Up Drill Down in Oracle BI Answers from General Ledger to Subledger

Manage Domains and Member Mappings for Channel Type Dimension

Manage Domains and Member Mappings for Cost Allocation Type Dimension

Manage Domains and Member Mappings for Customer Contact Dimension

Manage Domains and Member Mappings for Cost Element Dimension

Manage Domains and Member Mappings for Customer Expense Transaction Type Dimension

Manage Domains and Member Mappings for Product Expense Transaction Type Dimension

Manage Domains and Member Mappings for Expense Payment Type Dimension

Manage Domains and Member Mappings for Expense Status Dimension

Manage Domains and Member Mappings for Expense Type Dimension

Manage Domains and Member Mappings for Geography Dimension

Specify Order of First Name and Last Name in Position Hierarchy

How to Configure Group Account Numbers

How to Integrate Project Analytics with Financial Analytics

ETL Notes and Additional Information for Financial Analytics

B.1.4 List of Functional Configuration Tasks for Human Resources Analytics

This section lists Functional Configuration Tasks for Oracle Human Resources Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Human Resources Analytics

Overview of Human Resources Analytics

Configure Enterprise List

Configure Workforce Initial Extraction Date

Configure Workforce Snapshot Date

Configure Data Load Parameter for Workforce Adjusted Service Date Flag

Specify a Default Value for Number of Working Hours Per Week

Specify Order of First Name and Last Name in Position Hierarchy

Manage Domains and Member Mappings for Age Band Dimension

Manage Domains and Member Mappings for Business Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Employment Dimension

Manage Domains and Member Mappings for HR Assignment Dimension

Manage Domains and Member Mappings for HR Person Legislation Dimension

Manage Domains and Member Mappings for HR Position Dimension

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Job Dimension

Manage Domains and Member Mappings for Performance Band Dimension

Manage Domains and Member Mappings for Period Of Work Band Dimension

Manage Domains and Member Mappings for Time Dimension

Manage Domains and Member Mappings for Workforce Event Fact

Manage Domains and Member Mappings for Workforce Event Type Dimension

Configure Absence and Accrual Initial Extraction Date

Manage Domains and Member Mappings for Accrual Plan Dimension

Manage Domains and Member Mappings for Absence Event Dimension

Manage Domains and Member Mappings for Absence Type Reason Dimension

Configure Payroll Initial Extraction Date

Manage Domains and Member Mappings for Payroll Balance Dimension

Configure Initial Extract Date

Specify the Ledger or Set of Books for which General Ledger Data is Extracted

Manage Domains and Member Mappings for GL Account Dimension

How to Assign Group Account Numbers to Natural Accounts for HR Analytics

How To Grant GL Data Role to HR VP Users

ETL Notes and Additional Information for Human Resources Analytics

B.1.5 List of Functional Configuration Tasks for Oracle Marketing Analytics

This section lists Functional Configuration Tasks for Oracle Marketing Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Marketing Analytics

Overview of Marketing Analytics

Configure Initial Extract Date

Configure Enterprise List

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Specify Order of First Name and Last Name in Position Hierarchy

Specify the Maximum Number of Days in a Sales Stage

Configure Loading of Resources into Opportunity - Resource Helper Table

Configure Opportunity Revenue Aggregate Table

Manage Domains and Member Mappings for Agreement Status Dimension

Manage Domains and Member Mappings for Agreement Type Dimension

Manage Domains and Member Mappings for Campaign History

Manage Domains and Member Mappings for Channel Type Dimension

Manage Domains and Member Mappings for Customer Contact Dimension

Manage Domains and Member Mappings for Customer Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Geography Dimension

Manage Domains and Member Mappings for Interactions Fact

Manage Domains and Member Mappings for Interactions Opportunity Fact

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Marketing Lead Fact

Manage Domains and Member Mappings for Marketing Source Dimension

Manage Domains and Member Mappings for Opportunity Dimension

Manage Domains and Member Mappings for Opportunity Revenue Fact

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for Response Fact

Manage Domains and Member Mappings for Segment Dimension

Manage Domains and Member Mappings for Time Dimension

ETL Notes and Additional Information for Marketing Analytics

B.1.6 List of Functional Configuration Tasks for Oracle Partner Analytics

This section lists Functional Configuration Tasks for Oracle Partner Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Partner Analytics

Overview of Partner Analytics

Configure Initial Extract Date

Configure Enterprise List

Specify the Maximum Number of Days in a Sales Stage

Configure Loading of Resources into Opportunity - Resource Helper Table

Configure Opportunity Revenue Aggregate Table

Manage Domains and Member Mappings for Channel Type Dimension

Manage Domains and Member Mappings for Customer Contact Dimension

Manage Domains and Member Mappings for Customer Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Geography Dimension

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for Time Dimension

Manage Domains and Member Mappings for Opportunity Dimension

Manage Domains and Member Mappings for Opportunity Revenue Fact

Manage Domains and Member Mappings for Partner Enrollment Fact Group

Manage Domains and Member Mappings for Partner Program Measure Fact Group

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Specify Order of First Name and Last Name in Position Hierarchy

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Marketing Lead Fact

Manage Domains and Member Mappings for Marketing Source Dimension

Manage Domains and Member Mappings for Segment Dimension

ETL Notes and Additional Information for Partner Analytics

B.1.7 List of Functional Configuration Tasks for Oracle Procurement and Spend Analytics

This section lists Functional Configuration Tasks for Oracle Procurement and Spend Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Procurement and Spend Analytics

Overview of Procurement and Spend Analytics

Configure Initial Extract Date

Configure Enterprise List

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Configure Purchase Cycle Lines Aggregate Fact

Configure Purchase Receipts Aggregate Fact

Configure Data Load Parameters for Product Inventory Catalog

Configure Reporting Parameters for Product Inventory Catalog

Configure Subledger Modules to be extracted for GL Linkage information

Configure the number of historical monthly snapshots for AP Aging

Manage Domains and Member Mappings for AP Hold Release Reason Dimension

Manage Domains and Member Mappings for AP Invoice Approval Status Dimension

Manage Domains and Member Mappings for AP Invoice Payment Status Dimension

Manage Domains and Member Mappings for AP Spend Classification Dimension

Manage Domains and Member Mappings for AP Transaction Source Dimension

Manage Domains and Member Mappings for AP Transaction Type Dimension

Manage Domains and Member Mappings for AP Transactions and Balance Fact

Manage Domains and Member Mappings for Movement Type Dimension

Manage Domains and Member Mappings for PO Creation Method Dimension

Manage Domains and Member Mappings for PO Document Style Dimension

Manage Domains and Member Mappings for Purchase Agreement Fact

Manage Domains and Member Mappings for Purchase Change Order Fact

Manage Domains and Member Mappings for Purchase Line Type Dimension

Manage Domains and Member Mappings for Purchase Order Agreement Leverage Type Dimension

Manage Domains and Member Mappings for Purchase Order Shipment Type Dimension

Manage Domains and Member Mappings for Purchase Order Status Dimension

Manage Domains and Member Mappings for Purchase Order Transaction Type Dimension

Manage Domains and Member Mappings for Purchase Receipt Transaction Type Dimension

Manage Domains and Member Mappings for Purchase Receipts Fact

Manage Domains and Member Mappings for Purchase Requisition Fact

Manage Domains and Member Mappings for Spend Invoice Distribution Fact

Manage Domains and Member Mappings for Sourcing Group Type Dimension

Manage Domains and Member Mappings for Sourcing Negotiation and Response Fact

Manage Domains and Member Mappings for Sourcing Negotiation Status Dimension

Manage Domains and Member Mappings for Sourcing Outcome Type Dimension

Manage Domains and Member Mappings for Sourcing Response Line Award Status Dimension

Manage Domains and Member Mappings for Sourcing Response Status Dimension

Manage Domains and Member Mappings for Sourcing Response Visibility Type Dimension

Specify Order of First Name and Last Name in Position Hierarchy

Manage Domains and Member Mappings for Expense Payment Type Dimension

Manage Domains and Member Mappings for Expense Status Dimension

Manage Domains and Member Mappings for Expense Type Dimension

Manage Domains and Member Mappings for Geography Dimension

Manage Domains and Member Mappings for Business Location Dimension

Manage Domains and Member Mappings for Customer Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for GL Account Dimension

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for Payment Method Dimension

Manage Domains and Member Mappings for Supplier Account Dimension

Manage Domains and Member Mappings for Time Dimension

How to Implement GL Segment and GL Account Dimensions

How to Integrate Project Analytics with Procurement and Spend Analytics

Manage Domains and Member Mappings for Project Dimension Group

Manage Domains and Member Mappings for Project Resource Dimension Group

How to Integrate Procurement and Spend Analytics with Spend Classification

How to Remove Spend Classification Integration Metadata

How To Customize Extended Cross Functional Security for Accounts Payables

How To Customize Security for Procurement Executive / Spend Analyst

How To Customize Extended Cross Functional Security for Employee Expenses

ETL Notes and Additional Information for Procurement and Spend Analytics

B.1.8 List of Functional Configuration Tasks for Oracle Product Information Management Analytics

This section lists Functional Configuration Tasks for Oracle Product Information Management Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Product Information Management Analytics
Overview of Product Information Management Analytics

Configure Initial Extract Date

Configure Enterprise List

Configure Data Load Parameters for Master Organization

Configure Reporting Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Item Batch Import Fact

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Item Request Fact

Manage Domains and Member Mappings for Time Dimension

ETL Notes and Additional Information for Product Information Management Analytics

B.1.9 List of Functional Configuration Tasks for Oracle Project Analytics

This section lists Functional Configuration Tasks for Oracle Project Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Project Analytics
Overview of Project Analytics

Configure Initial Extract Date

Configure Enterprise List

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Job Dimension

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for Project Cost Dimension Group

Manage Domains and Member Mappings for Project Dimension Group

Manage Domains and Member Mappings for Project Resource Dimension Group

Manage Domains and Member Mappings for Time Dimension

Configure Time Grain of Cost Aggregate Fact

Configure Time Grain of Project Commitment Snapshot Fact

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Project Control Dimension Group

Manage Domains and Member Mappings for Project Billing Dimension Group

Manage Domains and Member Mappings for Project Contract Dimension Group

Configure Time Grain of Revenue Aggregate Fact

How to Integrate Financial Analytics with Project Analytics

How to Integrate Procurement and Spend Analytics with Project Analytics

How to Perform RPD Modifications for Cost and Revenue Time Grain Changes

ETL Notes and Additional Information for Project Analytics

B.1.10 List of Functional Configuration Tasks for Oracle Sales Analytics

This section lists Functional Configuration Tasks for Oracle Sales Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Sales Analytics
Overview of Sales Analytics

Configure Initial Extract Date

Configure Enterprise List

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Specify Order of First Name and Last Name in Position Hierarchy

Specify Time in Days before an Account needs an Activity

Specify Time in Days before an Account needs a Contact

Specify Time in Days before a Contact needs an Activity

Specify the Maximum Number of Days in a Sales Stage

Configure Loading of Resources into Opportunity - Resource Helper Table

Configure Opportunity Revenue Aggregate Table

Specify Interaction Type Code

Specify the Filter to be Applied on Interaction

Manage Domains and Member Mappings for Agreement Status Dimension

Manage Domains and Member Mappings for Agreement Type Dimension

Manage Domains and Member Mappings for Channel Type Dimension

Manage Domains and Member Mappings for Customer Contact Dimension

Manage Domains and Member Mappings for Customer Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for Geography Dimension

Manage Domains and Member Mappings for Interactions Fact

Manage Domains and Member Mappings for Interactions Opportunity Fact

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Marketing Lead Fact

Manage Domains and Member Mappings for Marketing Source Dimension

Manage Domains and Member Mappings for Opportunity Dimension

Manage Domains and Member Mappings for Opportunity Revenue Fact

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for Sales Forecast Fact

Manage Domains and Member Mappings for Segment Dimension

Manage Domains and Member Mappings for Service Request Fact

Manage Domains and Member Mappings for Territory Quota Fact

Manage Domains and Member Mappings for Territory Quota Dimension

Manage Domains and Member Mappings for Time Dimension

ETL Notes and Additional Information for Sales Analytics

B.1.11 List of Functional Configuration Tasks for Oracle Supply Chain and Order Management Analytics

This section lists Functional Configuration Tasks for Oracle Supply Chain and Order Management Analytics.

Note: For a list of additional common tasks that apply to multiple Offerings, see Section B.1.1, "List of Functional Configuration Tasks for Common Areas and Dimensions".

Supply Chain and Order Management Analytics
Overview of Supply Chain and Order Management Analytics

Configure Initial Extract Date

Configure Enterprise List

Configure Data Load Parameters for Master Organization

Configure Data Load Parameters for Product Catalog

Configure Reporting Parameters for Product Catalog

Configure Data Load Parameters for Product Inventory Catalog

Configure Reporting Parameters for Product Inventory Catalog

Configure Inventory Monthly Balance and the Inventory Transaction Aggregate Table

Manage Domains and Member Mappings for Business Location Dimension

Manage Domains and Member Mappings for Cost Element Dimension

Manage Domains and Member Mappings for Cost Valuation Unit Dimension

Manage Domains and Member Mappings for Customer Contact Dimension

Manage Domains and Member Mappings for Customer Location Dimension

Manage Domains and Member Mappings for Employee Dimension

Manage Domains and Member Mappings for GL Account Dimension

Manage Domains and Member Mappings for Internal Organization Dimension

Manage Domains and Member Mappings for Item Dimension

Manage Domains and Member Mappings for Movement Type Dimension

Manage Domains and Member Mappings for Party Dimension

Manage Domains and Member Mappings for Project Dimension Group

Manage Domains and Member Mappings for Project Resource Dimension Group

Manage Domains and Member Mappings for Supplier Account Dimension

Manage Domains and Member Mappings for Time Dimension

Configure Subledger Modules to be extracted for GL Linkage information

Configure the Customer Status History Fact table

Configure Early and Late Tolerances for Shipments

Configure Sales Invoice Lines and Sales Order Lines Aggregate Tables

Manage Domains and Member Mappings for AR Transaction Type Dimension

Manage Domains and Member Mappings for AR Transactions and Balance Fact

Manage Domains and Member Mappings for Order Fulfillment Fact

Manage Domains and Member Mappings for Channel Type Dimension

Manage Domains and Member Mappings for Order Shipping Fact

Manage Domains and Member Mappings for Order Scheduling Fact

Manage Domains and Member Mappings for Order Cycle Fact

Manage Domains and Member Mappings for Order Invoice Fact

Manage Domains and Member Mappings for Order Orchestration Process Fact

Manage Domains and Member Mappings for Payment Method Dimension

Configure the number of historical monthly snapshots for AR Aging

How to Configure Backlog Period Date

How to Track Multiple Attribute Changes in Bookings

How to Add Dates to the Order Cycle Time Table for Post-Load Processing

How to Add Closed Orders to Backlog Calculations

How to Include Incompleted Invoice Lines

How to Implement GL Segment Dimension and GL Segment Hierarchy

How to Incrementally Refresh the Inventory Monthly Balance Table

How to Configure the Inventory Transaction Aggregate Table for ETL Runs

How to Implement GL Account Dimension

How to Enable Project Dimensions

How to Grant Cross Functional Access to Order Management Users

ETL Notes and Additional Information for Supply Chain and Order Management Analytics

B.2 Informational Task Reference

This section contains Information-only tasks that are exposed in Functional Setup Manager as Informational Tasks. Informational Tasks contain conceptual information, or steps that are performed in tools other than Functional Setup Manager (for example, in DAC Client, or Oracle BI Administration Tool).

This section contains the following topics:

- Section B.3, "ETL Notes and Additional Information for Oracle Product Information Management Analytics"

- Section B.4, "ETL Notes and Additional Information for Oracle Supply Chain and Order Management Analytics"

- Section B.5, "ETL Notes and Additional Information for Oracle Procurement and Spend Analytics"

- Section B.6, "ETL Notes and Additional Information for Oracle Customer Data Management Analytics"

- Section B.7, "ETL Notes and Additional Information for Oracle Project Analytics"

- Section B.8, "ETL Notes and Additional Information for Oracle Partner Analytics"

- Section B.9, "ETL Notes and Additional Information for Oracle Financial Analytics"

- Section B.10, "ETL Notes and Additional Information for Oracle HR Analytics"

- Section B.11, "ETL Notes and Additional Information for Oracle Marketing Analytics"

- Section B.12, "ETL Notes and Additional Information for Oracle Sales Analytics"

- Section B.13, "How to Integrate Project Analytics with Procurement and Spend Analytics"

- Section B.14, "How To Integrate Procurement and Spend Analytics with Spend Classification"

- Section B.15, "How to remove or hide Oracle Spend Classification Integration Metadata"

- Section B.16, "How to Configure the Inventory Transaction Aggregate Table for ETL Runs"

- Section B.17, "How to Add Dates to the Order Cycle Time Table for Post-Load Processing"

- Section B.18, "How to Add Closed Orders to Backlog Calculations"

- Section B.19, "How to Track Multiple Attribute Changes in Bookings"

- Section B.20, "How to Integrate Project Analytics with Financial Analytics"

- Section B.21, "How to Include Uncompleted Invoice Lines"

- Section B.22, "How to Configure Backlog Period Date"

- Section B.23, "How To incrementally Refresh the Inventory Monthly Balance Table"

- Section B.24, "How to Enable Project Analytics Integration with Supply Chain and Order Management Analytics"
- Section B.25, "How to Integrate Project Analytics with Procurement and Spend Analytics"
- Section B.26, "How to Set Up Drill Down in Oracle BI EE from General Ledger to Subledger"
- Section B.27, "How to Implement GL Segment and GL Account Dimensions"
- Section B.28, "How to Set Up GL Balance Segment Aggregates"
- Section B.29, "How to Integrate Financial Analytics with Project Analytics"
- Section B.30, "How to Perform RPD Modifications for Cost and Revenue Time Grain Changes"
- Section B.31, "How to Assign Group Account Numbers to Natural Accounts for Human Resources Analytics"
- Section B.32, "How to Configure Group Account Numbers"
- Section B.33, "How To Customize Extended Cross Functional Security for Accounts Payables"
- Section B.34, "How To Customize Extended Cross Functional Security for Employee Expenses"
- Section B.35, "How To Customize Security for Procurement Executive / Spend Analyst"
- Section B.36, "How to Grant Cross Functional Access to Order Management Users"
- Section B.37, "How To Grant GL Data Role to HR VP Users"
- Section B.38, "How to Implement Asset Category and Asset Location Dimension"
- Section B.39, "How to Perform System Setups and Post Install Tasks for Oracle BI Applications"
- Section B.40, "How to Reload the Time Dimension Tables After the Data Warehouse Is Loaded"
- Section B.41, "How To Configure Scorecard Target Before Running ETL"
- Section B.42, "How to Configure Order Item and Service Request Flat Files For ETL"
- Section B.43, "How to Add Balances to BI Balance Group"
- Section B.44, "Getting Started With Functional Configuration"
- Section B.45, "Overview of Oracle Sales Analytics"
- Section B.46, "Overview of Oracle Marketing Analytics"
- Section B.47, "Overview of Oracle Partner Analytics"
- Section B.48, "Overview of Oracle Human Resources Analytics"
- Section B.49, "Overview of Oracle Customer Data Management Analytics"
- Section B.50, "Overview of Oracle Financial Analytics"
- Section B.51, "Overview of Oracle Product Information Management Analytics"
- Section B.52, "Overview of Oracle Project Analytics"
- Section B.53, "Overview of Oracle Procurement and Spend Analytics"

B.3 ETL Notes and Additional Information for Oracle Product Information Management Analytics

List of DAC Subject Areas for this Offering:

- FSCM - PIM Analytics All

- PIM_AN:Item Batch Import

- PIM_AN:Item Fact Group

- PIM_AN:Item Request Fact

List of Execution Plans:

- PIM_AN - Fusion V1 - Product Management Analytics

B.4 ETL Notes and Additional Information for Oracle Supply Chain and Order Management Analytics

Costing and Logistics

List of DAC Subject Areas for this Offering:

- COSTING_FA

  – SCOM_AN: Item Cost

  – SCOM_AN: Valuation

  – FIN_AN: Cost of Goods Sold

  – FIN_AN: GL Revenue

  – List of Execution Plans:

    SCOM_AN - Fusion V1 Costing

- LOGISTICS_FA

  – SCOM_AN: Inventory Balance

  – SCOM_AN: Inventory Cycle Count

  – SCOM_AN: Inventory Transactions

  – FIN_AN: Cost of Goods Sold

  – List of Execution Plans:

    SCOM_AN - Fusion V1 Logistics

Order Management

List of DAC Subject Areas for this Offering:

- ORDRMGMT_FA
  - SCOM_AN: Order Credit - OMORDERCREDIT_FG
  - SCOM_AN: Order Shipping - OMDELIVERY_FG
  - SCOM_AN: Order Fulfillment - OMORDERFULFILL_FG
  - SCOM_AN: Order Booking - OMBOOKING_FG
  - SCOM_AN: Order Cycle - OMCYCLE_FG
  - SCOM_AN: Order Orchestration Process - OMDOOPRCSS_FG
  - SCOM_AN: Order Customer Status History - OMCUSTSTATHIST_FG
  - SCOM_AN: Order Hold - OMORDERHOLD_FG
  - SCOM_AN: Order Invoice - OMINVOICE_FG
  - SCOM_AN: Order Scheduling - OMSCHEDULE_FG
  - SCOM_AN: Order Backlog - OMBACKLOG_FG
  - SCOM_AN: Order Invoice Credit - OMINVOICECREDIT_FG
  - FIN_AN: GL Revenue
  - FIN_AN: AR Transactions and Balance
  - List of Execution Plans:

    SCOM_AN - Fusion V1 - Order Management

B.5 ETL Notes and Additional Information for Oracle Procurement and Spend Analytics

List of DAC Subject Areas for this Offering:

- Employee Expense Functional Area
  - PROC_SPEND_AN: Expense Credit Card
  - PROC_SPEND_AN: Expense Overview
  - PROC_SPEND_AN: Expense Violations
- Procurement Functional Area
  - PROC_SPEND_AN: Purchase Agreement
  - PROC_SPEND_AN: Purchase Change Order
  - PROC_SPEND_AN: Purchase Cycle
  - PROC_SPEND_AN: Purchase Orders
  - PROC_SPEND_AN: Purchase Receipts
  - PROC_SPEND_AN: Purchase Requisition
  - PROC_SPEND_AN: Spend Invoice Distribution
  - PROC_SPEND_AN: ScoreCard
  - FIN_AN: AP Transactions and Balance

- Sourcing Functional Area

    – PROC_SPEND_AN: Sourcing Negotiation

    – PROC_SPEND_AN: Sourcing Response

List of Execution Plans:

- PROC_SPEND_AN - Fusion V1 Full

- PROC_SPEND_AN - Fusion V1 Procurement

- PROC_SPEND_AN - Fusion V1 Sourcing

- PROC_SPEND_AN - Fusion V1 Expense Report

B.6 ETL Notes and Additional Information for Oracle Customer Data Management Analytics

List of DAC Subject Areas for this Offering:

- CDM_AN: Customer Data Completeness

List of Execution Plans:

- CRM Customer Data Quality Fusion

    Applies to the Functional Area: CUSTDATAMGMT_FA   Customer Data Quality

B.7 ETL Notes and Additional Information for Oracle Project Analytics

List of DAC Subject Areas for this Offering:

- PROJECT_AN: Revenue

- PROJECT_AN: Funding

- PROJECT_AN: Forecast

- PROJECT_AN: Cross Charge

- PROJECT_AN: Cost

- PROJECT_AN: Contract

- PROJECT_AN: Commitment

- PROJECT_AN: Budget

- PROJECT_AN: Billing

List of Execution Plans:

- PROJECT_AN - Fusion V1 Full:

    Applicable to all Subject Areas listed above.

- PROJECT_AN - Fusion V1 - Costing & Control:

    Applicable to Subject Areas:

    – PROJECT_AN: Budget

    – PROJECT_AN: Commitment

    – PROJECT_AN: Cost

    – PROJECT_AN: Cross Charge

- PROJECT_AN: Forecast

- PROJECT_AN - Fusion V1 - Revenue and Billing:

  Applicable to Subject Areas:

  - PROJECT_AN: Billing

  - PROJECT_AN: Contract

  - PROJECT_AN: Funding

  - PROJECT_AN: Revenue

B.8 ETL Notes and Additional Information for Oracle Partner Analytics

List of DAC Subject Areas for this Offering:

- Functional Area 'Partner Programs' can be associated with the 'PRM_AN: Partner Program' and 'SALES_AN: Opportunity Revenue' DAC Subject Areas.

- Functional Area 'Partner Performance' can be associated with DAC Subject Areas 'PRM_AN: Partner Performance', and 'SALES_AN: Opportunity Revenue'.

- Functional Area 'Partner Deals' can be associated with DAC Subject Areas 'PRM_AN: Partner Deals' and 'MARKETING_AN: Marketing Lead'.

List of Execution Plans:

- CRM PRM - Fusion

B.9 ETL Notes and Additional Information for Oracle Financial Analytics

List of DAC Subject Areas for this Offering:

- Fixed Asset functional area contains:

  - FIN_AN: Fixed Asset Balance

  - FIN_AN: Fixed Asset Transactions

- Account Payables functional area contains:

  - FIN_AN: AP Transactions and Balance

  - FIN_AN: AP Holds

- Account Receivables functional area contains:

  - FIN_AN: AR Transactions and Balance

  - Execution Plan: FIN_AN - Fusion V1 - Receivables

- Profitability functional area contains:

  - FIN_AN: Cost of Goods Sold

  - FIN_AN: GL Revenue

  - FIN_AN: Customer Expenses

  - FIN_AN: Product Expenses

    Note: If you are allocating expenses by customers, you must include the Universal Subject Area: FIN_AN: Customer Expenses. If you are allocating expenses by product, you must include the Universal Subject Area: FIN_AN: Product Expenses.

- General Ledger functional area contains:
  - FIN_AN: GL Balance
  - FIN_AN: GL Journals

List of Execution Plans:

- FIN_AN - Fusion V1 - Fixed Asset
- FIN_AN - Fusion V1 - Payables
- FIN_AN - Fusion V1 - Profitability
- FIN_AN - Fusion V1 - General Ledger

Notes

- For the General Ledger functional area, if you want to extract budget data from a non-Fusion Applications source, you must use the Subject Area named FIN_AN: GL Budget in the Universal adapter.

B.10 ETL Notes and Additional Information for Oracle HR Analytics

List of DAC Subject Areas for this Offering:

- HR_AN: Absence Event (Functional Area: Absence and Accrual)
- HR_AN: Accrual Transaction (Functional Area: Absence and Accrual)
- HR_AN: Payroll Balance (Functional Area: Payroll)
- HR_AN: Workforce Event (Functional Area: Workforce Effectiveness)
- FIN_AN: GL Journals (Functional Area: Workforce Effectiveness)

List of Execution Plans:

- HR_AN - Fusion

B.11 ETL Notes and Additional Information for Oracle Marketing Analytics

List of DAC Subject Areas for this Offering:

- Core Marketing Functional Area:
  - MARKETING_AN: Campaign History
  - MARKETING_AN: Campaign Opportunity
  - MARKETING_AN: KPI
  - MARKETING_AN: Offer Product
  - MARKETING_AN: Response
- Marketing Lead Functional Area:
  - MARKETING_AN: Marketing Lead
- Opportunity Landscape Functional Area:
  - MARKETING_AN: Customer Purchase
  - SALES_AN:Agreement Contract Item Fact
  - SALES_AN:Asset

- – MARKETING_AN: Marketing Lead
- Customer Interactions Management Functional Area:
  - – SALES_AN:Interactions
- Opportunity and Revenue Functional Area:
  - – SALES_AN: Opportunity Revenue

Note: For information about Subject Areas Customer Interactions Management and Opportunity and Revenue Management, refer to ETL Notes and Additional Information for Sales Analytics.

List of Execution Plans:

- None.

B.12 ETL Notes and Additional Information for Oracle Sales Analytics

List of Functional Areas and DAC Subject Areas for this Offering:

- Customer Interactions Management
  - – SALES_AN:Interactions
  - – SALES_AN:Interactions Relationship Heatmap
  - – SALES_AN:Interactions Coverage
- Opportunity and Revenue Management
  - – SALES_AN: Opportunity Revenue
  - – SALES_AN:Interactions Opportunity
- Opportunity and Revenue Management for Segmentation
  - – SALES_AN: Opportunity Revenue Segmentation
- Sales Prediction Engine
  - – SALES_AN:Agreement Contract Item Fact
  - – SALES_AN:Asset
  - – SALES_AN:Service Request
  - – SALES_AN: Opportunity Revenue
  - – SALES_AN: Order
  - – MARKETING_AN: Marketing Lead
- Sales Forecasting
  - – SALES_AN: Sales Forecasting
  - – SALES_AN: Opportunity Revenue
- Quota Management
  - – SALES_AN: Resource Quota
  - – SALES_AN: Territory Quota
  - – SALES_AN: Opportunity Revenue
- Territory Management
  - – SALES_AN: Opportunity Revenue

- – SALES_AN: Territory Quota

- – SALES_AN: Sales Account

- – SALES_AN: Sales Forecasting

- – MARKETING_AN: Marketing Lead

- ■ Usage Accelerator

- – CDM_AN: Customer Data Completeness

List of Execution Plans:

- ■ None.

B.13 How to Integrate Project Analytics with Procurement and Spend Analytics

You can enable Oracle Procurement and Spend Analytics to use dimension tables in Oracle Project Analytics. You can only perform this integration if you have licensed Oracle Project Analytics.

To enable the integration

1. In DAC, select the 'Fusion' source system container, for the version of the OLTP that you are using.

2. Display the Subject Areas tab.

   You need to edit the following Subject Areas: PROC_SPEND_AN: Purchase Orders, PROC_SPEND_AN: Purchase Requisition, PROC_SPEND_AN: Spend Invoice Distribution.

3. For each Subject Area, do the following:

   a. Select the Subject Area.

   b. Display the Configuration Tags tab in the upper pane.

   c. Select the Enable Project Dimensions tag.

   d. Make sure that the **Inactive** check box is cleared.

4. Assemble the Subject Areas.

5. Click the Execute button and build the Execution Plan for the Subject Areas that you updated.

B.14 How To Integrate Procurement and Spend Analytics with Spend Classification

This section contains configuration steps that apply to Oracle Procurement and Spend Analytics when deployed with Oracle Spend Classification. For implementing Oracle Spend Classification and required patches, refer to the Oracle Spend Classification product documentation.

If you are not implementing Oracle Spend Classification, you might choose to remove or hide the Oracle Spend Classification integration metadata from the Presentation layer of the BI repository (for more information about removing Oracle Spend Classification metadata, see Section B.15, "How to remove or hide Oracle Spend Classification Integration Metadata").

Note: Oracle Spend Classification is not part of the core Oracle BI Applications product suite, and is not packaged with any module of Oracle BI Applications. It is a separate solution offered by Oracle, and a separate license is required. If you are interested in licensing and implementing Oracle Spend Classification, please contact your Oracle Sales Representative.

B.14.1 Overview to Oracle Spend Classification Integration

Oracle Spend Classification is a complementary product that can be used in conjunction with Oracle Procurement and Spend Analytics to improve the accuracy of Spend by converting 'unclassified' Spend into item categories. Oracle Procurement and Spend Analytics is designed to work with or without Oracle Spend Classification.

Typical procurement systems will have many PO, Invoice, and Expense Transactions without reference to item and item categories, and in most cases they might have item descriptions in a free text format. When you implement Oracle Procurement and Spend Analytics, these transactions will come into the system as 'Unclassified' because they do not have corresponding items and/or item categories. This issue is more prominent if your organization's Spend constitutes a major portion of Indirect Spend.

Oracle Procurement and Spend Analytics is installed with infrastructure required to feed data from Oracle Business Analytics Warehouse to Oracle Spend Classification, and feed the classified data back into Oracle Business Analytics Warehouse. This Infrastructure is provided as an additional feature for those customers who would like to take the advantage of both Oracle Procurement and Spend Analytics and Oracle Spend Classification.

If you choose not to use Oracle Spend Classification, Oracle Procurement and Spend Analytics can be deployed as a stand alone solution, and the features of Procurement and Spend Analytics can be deployed without any dependency on Oracle Spend Classification.

B.14.2 About the Oracle Spend Classification Metadata

This section describes the Oracle Spend Classification metadata and repository metadata that is available for use with Oracle Spend Classification.

The following facts are integrated with Oracle Data Classification to enrich and automatically assign category codes.

- W_AP_INV_DIST_F
- W_PURCH_COST_F
- W_RQSTN_LINE_COST_F

There are five types of taxonomy supported: UNSPSC, Oracle Purchasing Categories, and three custom categories. The classification resuLogical Table Source are stored in these columns:

- AUTO_UNSPSC_WID
- AUTO_PURCHASING_CATEGORY_WID
- AUTO_CUSTOM_CATEGORY1_WID
- AUTO_CUSTOM_CATEGORY2_WID
- AUTO_CUSTOM_CATEGORY3_WID

In the Analytics metadata repository (RPD), the following is configured by default.

- UNSPSC, Oracle Purchasing Categories, and Custom Category1 are configured up to the Business Model and Mapping layer. The facts and dimension names are as follows:

  - Fact - Spend and AP Invoice Distribution

  - Fact - Purchasing – Order

  - Fact - Purchasing – Requisition

  - Dim - Auto UNSPSC

  - Dim - Auto Purchasing Category

  - Dim - Auto Custom Category1

- In the Presentation Layer, 'Procurement and Spend - Invoice Lines' contains the columns for data classification, under the following folders:

  - Data Classification

  - Auto UNSPSC

  - Auto Purchasing Category

  - Auto Custom Category 1

B.14.3 How to deploy UNSPSC, Oracle Purchasing Categories, and Custom Category1

Follow these steps if you want to expose UNSPSC, Oracle Purchasing Categories, and Custom Category1 for your Purchase Order and Purchase Requisition Subject Area.

To deploy UNSPCC, Oracle Purchasing Categories, and Custom Category1:

1. Using Oracle BI Administration Tool, open the RPD file (for example, OracleBIAnalyticsApps.rpd).

   The OracleBIAnalyticsApps.rpd file is located in the \bifoundation\OracleBIServerComponent\coreapplication_obis*n*\repository folder.

2. In the Presentation layer, do the following:

   a. Expand the folder 'Procurement and Spend - Invoice Lines'.

   b. Multi-select the following folders and right click to copy:

      Data Classification

      Auto UNSPSC

      Auto Purchasing Category

      Auto Custom Category 1

   c. To implement Oracle Spend Classification in Purchase Orders, select the folder 'Procurement and Spend - Purchase Orders' and right click to paste in the folders.

   d. To implement Oracle Spend Classification in Purchase Requisitions, select the folder 'Procurement and Spend - Purchase Requisitions' and right click to paste in the selected folders.

   e. Verify the new folders.

   f. If required, re-order the folders as you would like the folders to be displayed to business users in the Presentation Services catalog.

3. Save and close the repository.

B.14.4 How to deploy the additional Custom Category2 and Custom Category3

To deploy Custom Category2 and Custom Catogory3:

Note: This task uses the Fact_W_AP_INV_DIST_F fact as an example, though you can also apply the steps to deploy other facts.

1. Using Oracle BI Administration Tool, open the RPD file (for example, OracleBIAnalyticsApps.rpd).

    The OracleBIAnalyticsApps.rpd file is located in the \bifoundation\OracleBIServerComponent\coreapplication_obis*n*\repository folder.

2. In the Physical layer, do the following:

    a. Right click on 'Dim_W_PROD_CAT_DH_AUTO_CUSTOM_CATEGORY1' under 'Oracle Data Warehouse' and select Duplicate.

    b. Rename it as 'Dim_W_PROD_CAT_DH_AUTO_CUSTOM_CATEGORY2'.

    c. Join dimension 'Dim_W_PROD_CAT_DH_AUTO_CUSTOM_CATEGORY2' and fact 'Fact_W_AP_INV_DIST_F' using the following condition:

    ```
    Dim_W_PROD_CAT_DH_AUTO_CUSTOM_CATEGORY2.ROW_WID = Fact_W_AP_INV_DIST_
    F.'AUTO_CUSTOM_CATEGORY2_WID
    ```

3. In the Business Model and Mapping layer, do the following:

    a. Immediately below table 'Dim - Auto Custom Category1', create 'Dim - Auto Custom Category2'.

    b. Immediately below hierarchy 'Auto Custom Category1', create 'Dim - Auto Custom Category2' based on the physical table 'Dim_W_PROD_CAT_DH_ AUTO_CUSTOM_CATEGORY2'.

    c. Join 'Dim - Auto Custom Category1' to 'Fact - Spend and AP Invoice Distribution'.

    d. Edit 'Fact - Spend and AP Invoice Distribution'. Fact_W_AP_INV_DIST_F. Display the Content tab, and set the level of 'Auto Custom Category2' to 'Custom Hierarchy Base Level'.

4. In the Presentation layer, do the following:

    a. Create a sub-folder called 'Auto Custom Category 2' in the 'Procurement and Spend - Invoice Lines' folder. Edit folder and add this exact string to the Description box.

    Auto Custom Category2 becomes a sub-folder of Data Classification.

    b. Order this folder so that it is after 'Auto Custom Category 1'.

    c. Drag the 'Dim - Auto Custom Category1' columns from the Business Model and Mapping layer into the 'Auto Custom Category 2' folder in the Presentation layer.

5. Save and close the repository.

6. Repeat steps 2 - 5 for Custom Category3.

B.15 How to remove or hide Oracle Spend Classification Integration Metadata

If you are not implementing Oracle Spend Classification, Oracle recommends that you remove or hide the Oracle Spend Classification integration metadata that is included in the Presentation layer of the BI repository. Hiding or deleting this metadata avoids potential confusion among business end users.

To remove or hide Oracle Spend Classification Integration Metadata:

1. Using Oracle BI Administration Tool, open the RPD file (for example, OracleBIAnalyticsApps.rpd).

    Deployed RPD files are located in *ORACLE_HOME*\bifoundation\OracleBIServerComponent\coreapplication_obis<n>\repository.

2. In the Presentation layer pane, expand the folder 'Procurement and Spend - Invoice Lines'.

    The Oracle Spend Classification metadata in the Physical layer consists of the following objects:

    Data Classification

    Auto UNSPSC

    Auto Purchasing Category

    Auto Custom Category 1

3. To remove the metadata objects listed above, right click on the objects and select Delete.

    > **Note:**   If you decide later to implement Oracle Spend Classification, you need to do the following:
    >
    > 1. In the Business Model and Mapping layer, drag and drop the following dimensions from a copy of the 'Procurement and Spend - Invoice Lines' folder into the Presentation layer or your metadata repository:
    >
    >   - Dim - Auto UNSPSC
    >
    >   - Dim - Auto Purchasing Category
    >
    >   - Dim - Auto Custom Category1

4. To hide the objects listed above from end users, right click and select Properties, then Permissions, and clear the **Read** permission check box for the appropriate user or group.

> **Note:** If you decide later to implement Oracle Spend Classification, you need to do the following:
>
> 1. To display the following objects to end users, right click and select Properties, then Permissions, and select the **Read** permission check box for the appropriate user or group:
>   - Data Classification
>   - Auto UNSPSC
>   - Auto Purchasing Category
>   - Auto Custom Category 1

5. Save and close the metadata repository.

B.16 How to Configure the Inventory Transaction Aggregate Table for ETL Runs

Before you run the initial ETL and then the incremental ETL to load the Product Transaction aggregate table, you need to configure the Product Transaction Aggregate Table, as follows.

To configure the Product Transaction Aggregate Table

1. In DAC, go to the Design view, and select the appropriate custom container from the drop-down list.

2. Display the Tasks tab.

3. Locate the Task named PLP_ProductTransactionAggregate, display the Parameters sub-tab, and make sure that the following three parameters are set as specified:
   - REFRESH_PERIOD = 'MONTH'
   - GRAIN = 'MONTH'
   - NUM_OF_PERIOD = 3

   Note: If any of these parameters do not exist, create them as Data Type = Text with the specified Values.

To configure the Product Transaction aggregate table for the initial ETL run

1. Retrieve the records in the Product Transaction fact (`W_PRODUCT_XACT_F`) table, and aggregate the records to the Product Transaction aggregate (`W_PRODUCT_XACT_A`) table at a certain grain level.

   For example, if GRAIN=MONTH then the records in the `W_PRODUCT_XACT_F` fact table are retrieved and aggregated to the `W_PRODUCT_XACT_A` table at a monthly level.

   Running the PLP_ProductTransactionAggregate workflow implements this step.

To configure the Product Transaction aggregate table for the incremental ETL run

1. Delete the refreshed records from the Product Transaction aggregate (`W_PRODUCT_XACT_A`) table for a certain time.

   The REFRESH_PERIOD and the NUM_OF_PERIOD parameters determine the time period for the deletion.

For example, if REFRESH_PERIOD=MONTH, NUM_OF_PERIOD=1, and the date is May 15, 2005, then all records for April and the current month (May) are deleted in the `W_PRODUCT_XACT_A` table.

Running the PLP_ProductTransactionAggregate workflow implements this step.

2. Retrieve the records in the Product Transaction fact (`W_PRODUCT_XACT_F`) table, and aggregate the records to the `W_PRODUCT_XACT_A` table at a certain grain level.

For example, if GRAIN=MONTH then the records in the `W_PRODUCT_XACT_F` fact table are retrieved and aggregated to the `W_PRODUCT_XACT_A` table at a monthly level.

Running the PLP_ProductTransactionAggregate workflow implements this step.

B.17 How to Add Dates to the Order Cycle Time Table for Post-Load Processing

To add more dates, you need to understand how the Cycle Line Fact table is populated. If you want to change the dates loaded into the Cycle Line Fact (W_SALES_CYCLE_LINE_F) table, then you have to modify the PLP_SalesCycleLinesFact_Load and PLP_SalesCycleLinesFact_Load_Full mappings that take the dates from the W_* tables and load them into the Cycle Line Fact table.

To add dates to the Cycle Time table load

1. In Informatica PowerCenter Designer, open the PLP folder (or Configuration for Post Load Processing folder).

2. In Warehouse Designer, modify the table definition for the target table to verify that it has a field to store this date.

For example, if you are loading 'Validated on Date' in the `W_SALES_CYCLE_LINE_F` table, then you need to create a new column, `VALIDATED_ON_DT`, and modify the target definition of the `W_SALES_CYCLE_LINE_F` table.

3. In Source Analyzer, modify the table definition of the source table to include this new column.

Continuing with the example, you would include the `VALIDATED_ON_DT` column in the `W_SALES_CYCLE_LINE_F` source table.

4. In Mapping Designer, modify the `PLP_SalesCycleLinesFact_Load` and `PLP_SalesCycleLinesFact_Load_Full` mappings to select the new column from any of the following source tables, and load it to the `W_SALES_CYCLE_LINE_F` target table:

   – `W_SALES_ORDER_LINE_F`

   – `W_SALES_INVOICE_LINE_F`

   – `W_SALES_PICK_LINE_F`

   – `W_SALES_SCHEDULE_LINE_F`

5. Modify the Source Qualifier SQL Override for the mapping, and map the column in the transformation to the target table.

B.18 How to Add Closed Orders to Backlog Calculations

Backlog information is stored in the W_SALES_BACKLOG_LINE_F and W_SALES_ BACKLOG_HISTORY_F tables. Many types of backlog exist in the Oracle Supply Chain and Order Management Analytics application—financial backlog, operational backlog, delinquent backlog, scheduled backlog, unscheduled backlog, and blocked backlog.

By default, the Oracle Supply Chain and Order Management Analytics application only extracts open sales orders from the Sales Order Lines (W_SALES_ORDER_LINE_ F) table and Sales Schedule Lines table (W_SALES_SCHEDULE_LINE_F) for backlog calculations to populate the Backlog tables. Open sales orders are defined as orders that are not complete or not cancelled. The purpose in extracting only open orders is that in most organizations those orders that are closed are no longer a part of backlog. However, if you want to extract sales orders that are marked as closed, you may remove the default filter condition from the extract mapping.

For example, assume your customer orders ten items. Six items are invoiced and shipped, but four items are placed on operational and financial backlog. This backlog status continues until one of two things happens:

- The items are eventually shipped and invoiced.

- The remainder of the order is canceled.

If you choose to extract sales orders that are flagged as closed, then you must remove the condition in the Backlog flag. To do so, use the following procedure.

Note: The OPR_BACKLOG_FLG and FIN_BACKLOG_FLG in the W_SALES_ ORDER_LINE_F and W_SALES_SCHEDULE_LINE_F table is also used to identify which sales orders are eligible for backlog calculations.

To remove open order extract filters

1. In Informatica PowerCenter Designer, open the SDE_FUSION_V1_Adaptor.

2. Open the mplt_SA_FUSION_SalesOrderLinesFact mapplet in the Mapplet Designer.

3. Double-click on the EXP_SA_SalesOrderLinesFact, and display the Ports tab.

4. Edit the VAR_OPR_BACKLOG_FLG and remove the code `INP_ FulfillLineOpenFlag = 'Y' AND`.

5. Edit the VAR_FIN_BACKLOG_FLG and remove the code `INP_ FulfillLineOpenFlag = 'Y' AND`.

   W_SALES_SCHEDULE_LINE_FS is derived from W_SALES_ORDER_LINE_FS via SDE_FUSION_SalesScheduleLinesFact_Derive. Therefore, the changes in Steps 1 to 5 are automatically effective on the same flag columns on W_SALES_ SCHEDULE_LINE_F.

6. Validate and save your changes to the repository.

7. Open the PLP folder.

8. Open the mappings PLP_SalesBacklogLinesFact_LoadOrderLines and PLP_ SalesBacklogLinesFact_LoadScheduleLines.

9. In the Source Qualifier, remove the condition `AND W_STATUS_D.W_STATUS_ CODE <> 'Closed'` from the SQL Query.

B.19 How to Track Multiple Attribute Changes in Bookings

When you modify the default `VAR_BOOKING_ID` column, the SQL statement is configured as follows:

```
TO_CHAR(INP_FulfillLineId)||'~'||TO_CHAR(INP_FulfillLineInventoryItemId)||'~'||TO_
CHAR(INP_FulfillLineFulfillOrgId)
```

However, if you want to track changes based on more than one attribute, then in the SQL statement you must concatenate the attribute column IDs in the EXT_BOOKING_ ID column. For example, if you want to track changes in Customer Account, then concatenate the technical name IDs in the EXT_BOOKING_ID column as follows:

```
TO_CHAR(INP_FulfillLineId)||'~'||TO_CHAR(INP_FulfillLineInventoryItemId)||'~'||TO_
CHAR(INP_FulfillLineFulfillOrgId)||'~'|| TO_CHAR(INP_HeaderSoldToCustomerId))
```

To track dimensional attribute changes in bookings

1. In Informatica PowerCenter Designer, open the SDE_FUSION_V1_Adaptor.

2. Open the following mapping:
   - mplt_SA_FUSION_SalesOrderLinesFact

3. Double-click the appropriate Expression transformation to open the Edit Transformation box:
   - EXP_SA_SalesOrderLinesFact

4. In the Ports tab, edit the expression for the EXT_BOOKING_ID port, and enter the ID of the attribute for which you want to track changes.

   If you want to track changes in multiple attributes, then concatenate the IDs of all attributes and put the concatenated value in the `VAR_BOOKING_ID` column.

5. Validate and save your changes to the repository.

   Note: W_SALES_SCHEDULE_LINE_FS is derived from W_SALES_ORDER_ LINE_FS via SDE_FUSION_SalesScheduleLinesFact_Derive. Therefore, the changes in Steps 1 to 5 are automatically effective on the same column on W_ SALES_SCHEDULE_LINE_F.

B.20 How to Integrate Project Analytics with Financial Analytics

You can enable Oracle Financial Analytics to use dimension tables in Oracle Project Analytics. You can only perform this integration if you have licensed Oracle Project Analytics. You can configure the following Subject Areas in Oracle Financial Analytics to use Oracle Project Analytics tables:

- Financials - Payables
- Financials - Receivables

The following Oracle Financial Analytics fact tables integrate with Project Analytics dimensions:

- W_AP_XACT_F
- W_AP_BALANCE_F
- W_AR_XACT_F
- W_AR_AGING_INVOICE_A

To Enable Project Analytics Integration with Financial Subject Areas:

1. In DAC, select the 'Fusion' source system container.

2. Select each Subject Area, and then in the Configuration Tags tab, clear the Inactive check box for the Enable Project Dimensions configuration tag.

3. Assemble the Subject Areas.

4. Click the Execute button and build the Execution Plan for the Subject Areas that you updated.

B.21 How to Include Uncompleted Invoice Lines

By default, the Oracle Supply Chain and Order Management Analytics application is configured to extract completed sales invoices when performing the Sales Invoice data extract. Fusion Applications uses a flag to indicate whether a sales invoice is complete. In particular, completed sales invoices in Fusion Applications are those where the TransactionLinePVO.TransactionHeaderCompleteFlag = 'Y'.

To extract incomplete sales invoices, as well as complete invoices, remove the extract filter statement, as follows:

1. In Informatica PowerCenter Designer, open the SDE_FUSION_V1_Adapter folder.

2. Open the mplt_BC_FUSION_SalesInvoiceLinesFact mapplet in the Mapplet Designer.

3. Double-click the SQ_FscmTopModelAM_FinArTopPublicModelAM_ SalesInvoiceCustomerTrxLinesPVO Source Qualifier to open the Edit Transformations box.

4. Display the Properties tab.

5. For the SQL Query Transformation Attribute, select the down arrow in the Value field to display the SQL Editor box.

6. In the SQL box, remove the line 'AND `TransactionLinePVO.TransactionHeaderCompleteFlag = 'Y'`.

7. Validate and save your changes to the repository.

8. Repeat steps 2 - 7 for the mplt_BC_FUSION_SalesInvoiceLinesFact_Primary.

B.22 How to Configure Backlog Period Date

The Backlog table (`W_SALES_BACKLOG_LINE_F`) stores backlog data for the current month. In contrast, the Backlog History table (`W_SALES_BACKLOG_LINE_F`) stores snapshots of all previous months' historical backlog data. The periods for which the Backlog History table tracks backlog data is defined by the Backlog Period Date. By default, the date is set as the last calendar day of the month; however you may configure this date. You may want to view backlog history at a more detailed level, such as by day or by week, instead of by month. The following example describes how historical backlog data is stored and what the implications are for changing the backlog time period.

Example

Assume you represent a manufacturing company where financial backlog is defined as any item that is ordered, but not invoiced. On February 1, 2001, you received an order (Sales Order #1) for 30 products. 20 were shipped and invoiced and 10 were shipped, but not invoiced. At the end of the day, there is an entry in the Backlog table and in the

Backlog History table. The entry in the Backlog History table looks like that shown in Table B–1.

Table B–1   Fusion: Backlog History Table Entry as of February 1, 2001

SALES_ORDER_ NUM(Sales Order Number)	BACKLOG _ DK(Backlog Date)	BACKLOG_PERIOD_ DK(Backlog Period Date)	OPEN_QTY(Backlog Quantity)
1	02/01/2001	02/28/2001	10

On February 2, 5 of the 10 financial backlog items are invoiced and, thus, removed from the backlog. Thus, there is an update to the existing row in the Backlog History table, as shown in Table B–2.

Table B–2   Fusion: Backlog History Table Entry as of February 2, 2001

SALES_ORDER_NUM (Sales Order Number)	BACKLOG _DK (Backlog Date)	BACKLOG_PERIOD_ DK (Backlog Period Date)	OPEN_QTY (Backlog Quantity)
1	02/01/2001	02/28/2001	Old value: 10 New value: 5

No further activity happens until March 1st. On March 1st, the remaining 5 items on financial backlog are invoiced and removed from financial backlog. In addition, a new sales order (Sales Order #2) comes in for 50 new items. All of the items are put on financial backlog.

Even though all items from Sales Order #1 are cleared from financial backlog, the last backlog row remains in the Backlog History table. The purpose in retaining the last row is to indicate that there was backlog for this particular order. The quantity, in this case 5 items, does not tell you how many items were initially on backlog, which was 10.

For the 50 new financial backlog items, there is a new entry into the Backlog History table. So, as of February 28, 2001, the Backlog History table looks like the Table B–3.

Table B–3   Fusion: Backlog History Table Entry as of February 28, 2001

SALES_ORDER_NUM (Sales Order Number)	BACKLOG _DK (Backlog Date)	BACKLOG_PERIOD_ DK (Backlog Period Date)	OPEN_QTY (Backlog Quantity)
1	Old value: 02/01/2001 New value: 02/02/2001	02/28/2001	Old value: 10 New value: 5

On March 1, 30 more items are ordered (Sales Order #3), all of which are on financial backlog. The resulting Backlog History table looks like Table B–4.

Table B–4   Fusion: Backlog History Table Entry as of March 1, 2001

SALES_ORDER_NUM (Sales Order Number)	BACKLOG _DK (Backlog Date)	BACKLOG_PERIOD_ DK (Backlog Period Date)	OPEN_QTY (Backlog Quantity)
1	Old value: 02/01/2001 New value: 02/02/2001	02/28/2001	5
2	03/01/2001	03/31/2001	50
3	03/01/2001	03/31/2001	30

Because backlog history is maintained at the monthly level, you have a partial history of your backlogs. Based on the latest state of the Backlog History table shown in Table B–4, you can see that sales order number 1 ended up with 5 financial backlogged items. You do not have visibility into what the initial financial backlogged item quantities were for the sales orders; you only have their ending quantities.

If you decide that you want to track more details on how the items moved out of backlog, then you must maintain the history at a more granular level. For instance, if you want to know the number of items that were on backlog when it was first opened, then you track the backlog history by day, instead of by month.

For example, if you maintained backlog history at the daily level, then you are able to capture that sales order 1 had an initial backlog of 10 as of February 1 and the backlog quantity shrank to 5 as of February 2. So, by capturing history at the daily level, you could then compute cycle times on how long it took to move items out of backlog. However, if you decide to capture backlog history at a more detailed level, then you may compromise performance because tracking backlog history at the daily level can increase the size of the Backlog History table exponentially.

If you choose to change the time period for which historical backlog data is kept, then you must verify that all types of backlog are being stored at the same grain, which requires modification to multiple mappings. Table B–5 provides a list of all applicable mappings and their corresponding Expression transformations that you must modify.

Table B–5   Fusion: Backlog History Applicable Mappings and Expression Transformations

Mapping	Expression Transformation
PLP_SalesBacklogLinesfact_ LoadOrderLines	EXP_SALES_ORNLNS_BACKLOG
PLP_SalesBacklogLinesfact_ LoadScheduleLines	EXP_SALES_SCHLNS_BACKLOG

The backlog history period is monthly by default. The default SQL statement in the Expression transformation for the port BACKLOG_PERIOD_DK is:

```
TO_DECIMAL(TO_CHAR(LAST_DAY(CALENDAR_DATE),'YYYYMMDD'))
```

To capture a more detailed backlog history, edit the backlog period date with the following procedure. Possible periods include daily (CAL_DAY_DT), weekly (CAL_ WEEK_DT), monthly (CAL_MONTH_DT), and quarterly (CAL_QTR_DT).

B.23 How To incrementally Refresh the Inventory Monthly Balance Table

To incrementally refresh the Inventory Monthly Balance table:

1. Delete the records from the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F) aggregate table for a certain time.

   The GRAIN parameter determines the time period for the deletion. For example, if GRAIN=MONTH, and the date is May 15, 2005, then all records for April and the current month (May) are deleted in the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F) table.

   Running the PLP_InventoryMonthlyBalance workflow mapping implements this step.

2. Retrieve the records in the Inventory Balance (`W_INVENTORY_DAILY_BAL_F`) fact table and load the records to the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F) table at a certain grain level.

   For example, if GRAIN=MONTH, then the month end balance records in the `W_INVENTORY_DAILY_BAL_F` fact table are stored in and aggregated to the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F).

   Running the `PLP_InventoryMonthlyBalance` session, and the `PLP_InventoryMonthlyBalance` mapping implements this step. For the current month balance, balance records of the previous day (if it is in the same month) are deleted from W_INVENTORY_MONTHLY_BAL_F, and balance records of the current day will be loaded from W_INVENTORY_BALANCE_F to W_INVENTORY_MONTHLY_BAL_F.

   Running the PLP_InventoryMonthlyBalance workflow implements this step.

3. Remove the old records from the W_INVENTORY_DAILY_BAL_F fact table.

   To remove old records you need to use the KEEP_PERIOD and the NUM_OF_PERIOD parameters. For example, if KEEP_PERIOD=MONTH, NUM_OF_PERIOD=1, and the date is May 15, 2005, then the records for April and the current month (May) are kept and the older records are deleted.

   Running the PLP_InventoryDailyBalance_Trim workflow implements this step.

   > **Note:** The trimming process is to reduce data size in the table. It is important to emphasize that you will not be able to see the old daily balance records. But you will still be able to see the month-end balance. Therefore, please make sure that you adjust the NUM_OF_PERIOD values to reflect your data volume and data recency requirements.

To Configure Inventory Monthly Balance and the Inventory Transaction Aggregate Table:

1. Delete the records from the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F) aggregate table for a certain time.

   The GRAIN parameter determines the time period for the deletion. For example, if GRAIN=MONTH, and the date is May 15, 2005, then all records for April and the current month (May) are deleted in the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F) table.

   Running the PLP_InventoryMonthlyBalance workflow mapping implements this step.

2. Retrieve the records in the Inventory Balance (`W_INVENTORY_DAILY_BAL_F`) fact table and load the records to the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F) table at a certain grain level.

For example, if GRAIN=MONTH, then the month end balance records in the `W_INVENTORY_DAILY_BAL_F` fact table are stored in and aggregated to the Monthly Balance (W_INVENTORY_MONTHLY_BAL_F).

Running the `PLP_InventoryMonthlyBalance` session, and the `PLP_InventoryMonthlyBalance` mapping implements this step. For the current month balance, balance records of the previous day (if it is in the same month) are deleted from W_INVENTORY_MONTHLY_BAL_F, and balance records of the current day will be loaded from W_INVENTORY_BALANCE_F to W_INVENTORY_MONTHLY_BAL_F.

Running the PLP_InventoryMonthlyBalance workflow implements this step.

3. Remove the old records from the W_INVENTORY_DAILY_BAL_F fact table.

To remove old records you need to use the KEEP_PERIOD and the NUM_OF_PERIOD parameters. For example, if KEEP_PERIOD=MONTH, NUM_OF_PERIOD=1, and the date is May 15, 2005, then the records for April and the current month (May) are kept and the older records are deleted.

Running the PLP_InventoryDailyBalance_Trim workflow implements this step.

> **Note:** The trimming process reduces the amount of data in the table. It is important to emphasize that after data trimming you will not be able to see the old daily balance records. However, you will still be able to see the month-end balance. Therefore, please make sure that you adjust the NUM_OF_PERIOD values to reflect your data volume and data recency requirements.

B.24 How to Enable Project Analytics Integration with Supply Chain and Order Management Analytics

You can enable Oracle Supply Chain and Order Management to use dimension tables in Oracle Project Analytics. You can only perform this integration if you have licensed Oracle Project Analytics. You can configure the Oracle Supply Chain and Order Management Subject Areas listed below to join to certain Project Dimensions: Inventory Transactions (Project Dim, Task Dim, Financial Resource Dim).

The following Supply Chain fact table integrates with Project Analytics dimensions:

W_PRODUCT_XACT_F

Due to a limitation in Fusion Applications, the following Subject Areas of Oracle Supply Chain and Order Management Analytics are included in the configuration tag 'Enable Project Dimension', but are inactivated by default. Please note that these settings are intentional, and they should not be re-activated.

- SCOM_AN: Order Backlog
- SCOM_AN: Order Booking
- SCOM_AN: Order Credit
- SCOM_AN: Order Customer Status History
- SCOM_AN: Order Cycle

- SCOM_AN: Order Fulfillment

- SCOM_AN: Order Hold

- SCOM_AN: Order Invoice

- SCOM_AN: Order Invoice Credit

- SCOM_AN: Order Scheduling

- SCOM_AN: Order Shipping

To Enable Project Analytics Integration with Supply Chain and Order Management Analytics:

1. In DAC, display the Design view and select the appropriate custom container from the drop-down list.

2. Display the Configuration Tags tab.

3. Query for the tag Enable Project Dimensions.

4. Display the Subject Areas sub-tab.

5. Clear the **Inactive** check box for the Subject Area named 'SCOM_AN:Inventory Transactions', and save the details.

6. Assemble the Subject Area.

7. Click the Execute button and build the Execution Plan for the Subject Area that you updated.

B.25 How to Integrate Project Analytics with Procurement and Spend Analytics

You can enable Oracle Procurement and Spend Analytics to use dimension tables in Oracle Project Analytics. You can only perform this integration if you have licensed Oracle Project Analytics.

You can configure the following Procurement and Spend Subject Areas to join to the following Project Dimensions:

- Employee Expenses - Overview (Project Dim, Task Dim, Financial Resource Dim)

- Procurement and Spend - Invoice Lines (Project Dim, Task Dim)

- Procurement and Spend - Purchase Orders (Project Dim, Task Dim)

- Procurement and Spend - Purchase Requisitions (Project Dim, Task Dim)

The following Oracle Procurement and Spend Analytics fact tables integrate with Project Analytics dimensions:

- W_AP_XACT_F

- W_EXPENSE_F

- W_PURCH_COST_F

- W_RQSTN_LINE_COST_F

- W_AP_INV_DIST_F

To Enable Project Analytics Integration with Procurement and Spend Subject Areas:

1. In DAC, select the Oracle Fusion Applications source system container for the version of the OLTP that you are using.

2. Select each Subject Area, and then in the Configuration Tags tab, clear the Inactive check box for the Enable Project Dimensions configuration tag.

3. Assemble the Subject Areas.

4. Click the Execute button and build the Execution Plan for the Subject Areas that you updated.

B.26 How to Set Up Drill Down in Oracle BI EE from General Ledger to Subledger

To set up drill down in Oracle BI Answers from General Ledger to subledger:

1. Create your subledger request from 'Financials - AP Transactions' or 'Financials - AR Transactions' catalog as applicable.

2. In your request, add a filter on the column 'GL Journal ID' under the 'Document Details' subfolder for the 'AP Line Details' or 'AR Line Details' folder,=65 and then set the operator of the filter to 'Is Prompted'.

3. Build your GL Journal request from the 'Financials - GL Detail Transactions' catalog.

4. To your request, add the column 'GL Journal ID' under the 'Document Details' folder.

5. Navigate to the Column Properties of this column, and set the Value Primary Interaction property in the Column Format Interaction tab to 'NavigateAction Links'.

6. Add a navigation target and set the target location to the sub ledger request you created earlier.

You may add multiple navigation targets if your GL report shows transactions from multiple subledgers and you want to drill from GL to the appropriate Subledger report. For example, if your GL report shows transactions from AP, AR and Revenue, and you have three subledger analyses for each of these, you can add three navigation targets (by selecting the option 'Add Navigation TargetsAction Link') and set the locations to each of these analyses. Subsequently, when you run the GL report and click on the 'GL Journal ID' column value, a popup appears, where you need to click on the appropriate target based on the journal you clicked on. This will not happen automatically. For example, if you click on a journal transaction originating from AP, you need to pick the appropriate subledger report (that is, the AP report in this case) to drill into the AP report and see the details. You can add the Group Account Number attribute from GL Account Dimension to your GL report to easily identify the subledger that the GL transaction belongs to.

> **Note:** For COGS, the 'GL Journal ID' column is not exposed in any presentation catalogs. It is available in the business model layer of the RPD metadata under the logical tables 'Dim - GL COGS Details'. As a workaround, you can create presentation catalogs to report on detail level transactions for COGS and expose this column under the 'Document Details' folder in the presentation catalog. You use similar steps as described above to setup a drill-down from GL to COGS.
>
> 1. Using Oracle BI Administration Tool, open the RPD file (for example, OracleBIAnalyticsApps.rpd).
>
>   The OracleBIAnalyticsApps.rpd file is located in the \bifoundation\OracleBIServerComponent\coreapplication_ obisn\repository folder.
>
> 2. Create an empty presentation catalog (e.g. Financials – GL Cost of Goods Sold). Set properties by following other presentation catalogs.
>
> 3. Drag 'Dim – GL COGS Details' and 'Fact - Fins - GL Cost of Goods Sold Posted' to the presentation catalog.
>
> 4. Drag other dimensions.
>
> 5. Rename the presentation table 'Dim – GL COGS Details' to 'Document Details'.
>
> 6. Rename the presentation table 'Fact - Fins - GL Cost of Goods Sold Posted' to 'Facts - GL Cost of Goods Sold'. Rename other dimensions if necessary.
>
> You might also follow this same process to create a Presentation Table for Revenue to be able to drill from GL to Revenue level detail transactions.

B.27 How to Implement GL Segment and GL Account Dimensions

Follow the steps in this section to implement GL Segment and GL Segment Hierarchy Dimensions.

Guidelines

- If you need to report on only concatenated segments, then no configuration is required, and you can skip this section.

- If you want only Group Account Num (and related attributes), then at a minimum you need to configure just the Natural Account dimension.

- If you are exposing any GL Segments (including cost center, balancing, natural account), then you must go through the full configuration.

- If you are exposing any Financial fact, then at a minimum you need to configure the Natural Account dimension, because you need group account number.

B.27.1 Configuring the BI Extender

Before using BI Extender, you must perform the task named 'Performing Preconfiguration Tasks for the BI Extender' in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition)*.

B.27.2 Configuring GL Segment and GL Account Dimensions

This section explains how to configure GL Segment and GL Account Dimensions, and contains the following topics:

- Appendix B.27.2.1, "Mapping the segment labels to BI Objects in Fusion Applications"

- Appendix B.27.2.2, "Configuring GL Segments and GL Account using the BI Extension process"

- Appendix B.27.2.3, "Validating the Logical Table Source Filters for Generic Segment Divisions"

- Appendix B.27.2.4, "Re-configuring Segment Dimensions"

B.27.2.1 Mapping the segment labels to BI Objects in Fusion Applications

In order to enable GL Accounting Flexfield in Oracle BI Applications, the following configuration must be performed in the Manage Key Flexfields UI in Fusion Applications. This configuration enables the Accounting Flex Segments for BI and provides the mapping with BI Object names that should be used as dimensions for each of the Account Flexfield segments.

To map the segment labels to BI Objects in Fusion Applications:

1. In Fusion Applications, navigate to Manage Key Flexfields.

2. In each of the Accounting Flexfield segments, set the BI Enabled Flag to Y, as follows:

   a. Query for 'GL#' as Key Flexfield Code.

   b. Click Manage Structure Instances.

   c. Edit each of the segments and select the **BI enabled** check box, then save the details.

      This should be done for all segments in every Structure instance that you intend to be mapped in the BI Metadata Repository (RPD).
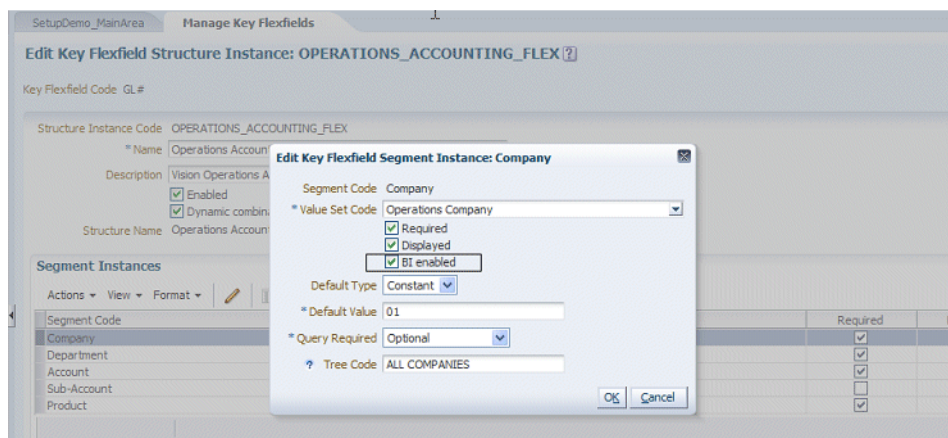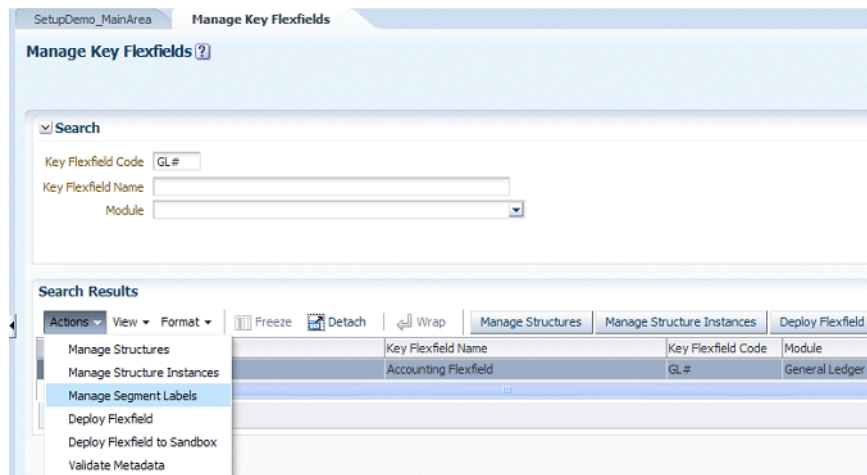

3. Populate the BI Object Name for each of the Segment Labels, as follows:

   Note: This name is the Logical table name in the RPD which is used as the dimension for the corresponding segment.

   a. In the Manage Key Flexfields UI in Fusion Applications, query for 'GL#' as Key Flexfield Code.

b. Choose Actions, then Manage Segment Labels.

c. Populate the BI Object Name for all the segment labels that you need to map in the RPD, then save the details.
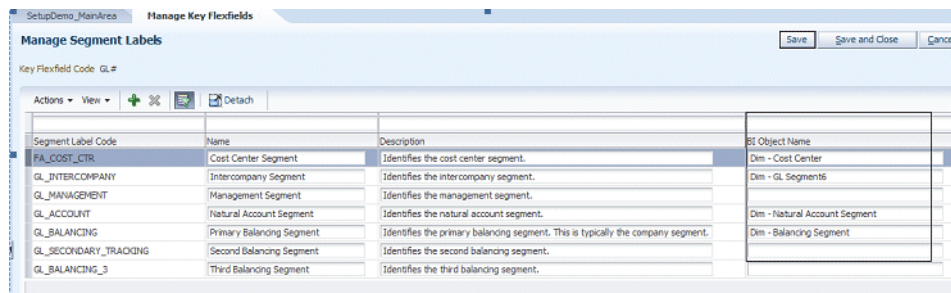
The following table shows the BI Object Names for each Qualified Segment label.

Table B–6   Mappings for Segment Label Codes to BI Object Names

Segment Label Code	BI Object Name
FA_COST_CTR	Dim - Cost Center
GL_BALANCING	Dim - Balancing Segment
GL_ACCOUNT	Dim - Natural Account Segment

For the non qualified segment labels, the BI Object Name should be populated with one of the 10 numbered Dim - Segments: Dim - GL Segment1, Dim - GL Segment2, Dim - GL Segment<*n*>, and so on, to Dim - GL Segment10.

4. Click the Deploy Flexfield option to deploy the Flexfields.

B.27.2.2 Configuring GL Segments and GL Account using the BI Extension process

This section describes how to configure the GL Accounting Segment Dimension in the BI Metadata Repository (RPD), and how to extend the ETL (Informatica and DAC) metadata to populate the corresponding tables in Oracle Business Analytics Warehouse.

> **Note:** Oracle recommends that before you start the extension process, you back up all metadata repositories. You should back up the OBIEE metadata (RPD file), Informatica Repository, and the DAC repository.

Overview on the BI Extension Process

This section provides a brief overview of the BI Extension process.

There are no default mappings to populate the segment dimension data warehouse tables (W_COST_CENTER_D, W_COST_CENTER_DH, W_NATURAL_ACCOUNT_D, W_NATURAL_ACCOUNT_DH, W_BALANCING_SEGMENT_D, W_BALANCING_SEGMENT_DH, W_GL_SEGMENT_D, W_GL_SEGMENT_DH). Informatica mappings and DAC tasks to populate these tables are generated by the BI extension process. This process is driven through the RPD metadata. The logical dimensions in the RPD metadata corresponding to these tables are 'Dim – Cost Center', 'Dim – Balancing Segment', 'Dim – Natural Account Segment' and all 'Dim – GL Segment<*n*>' dimensions. These dimension tables are populated from a Tree VO or from a Value Set VO depending on whether a tree was associated with the segment or not in the Fusion Applications setup.

For each segment associated with trees, two VOs will be generated (Tree and TreeCode) with the following naming structure:

- FscmTopModelAM.AccountBIAM.FLEX_TREE_VS_ _VI

- FscmTopModelAM.AccountBIAM.FLEX_TREECODE_VS__VI

For each segment without trees, one VO will be generated with the following naming structure:

- FscmTopModelAM.AccountBIAM.FLEX_VS_<XXX>_VI

In addition to the segment dimension tables, the BI Extension process also extends the installed ETL mapping that populates the GL Account Dimension; W_GL_ACCOUNT_D. This dimension table has a pair of columns for each segment dimension. For example, COST_CENTER_NUM and COST_CENTER_ATTRIB for Cost Center dimension, BALANCING_SEGMENT_NUM and BALANCING_SEGMENT_ATTRIB for Balancing Segment dimension, ACCOUNT_SEGn_CODE and ACCOUNT_SEGn_ATTRIB corresponding to the generic GL Segmentn dimensions. These columns will be populated from the Flex BI Flattened VO; FscmTopModelAM.AccountBIAM.FLEX_BI_Account_VI. This VO will have a pair of columns for each segment; <segmentlabel>_ and <segmentlabel>_c. For example, for your Cost Center segment which has the segment label FA_COST_CTR, there will be two columns in this VOs named FA_COST_CTR_ and FA_COST_CTR_c.

BI Extension Process Flow

- Step 1 - Import the appropriate View Objects (VOs) from the ADF data source.

- Step 2 – Verify the automatic mapping of the VOs to the logical objects in the mapping dialog.

- Step 3 - Provide connection information such as user name and password for Informatica and DAC repositories.

- Step 4 - Click finish, and the appropriate Informatica and DAC metadata are generated and updated in the respective repositories.

Prerequisite

Before you start the BI Extension process, you must enable the Extender For BIAPPS setting in Oracle BI Administration Tool. To do this, choose Tools, then Options, then General, to display the Options dialog, and select the **Extender for BIAPPS** check box.
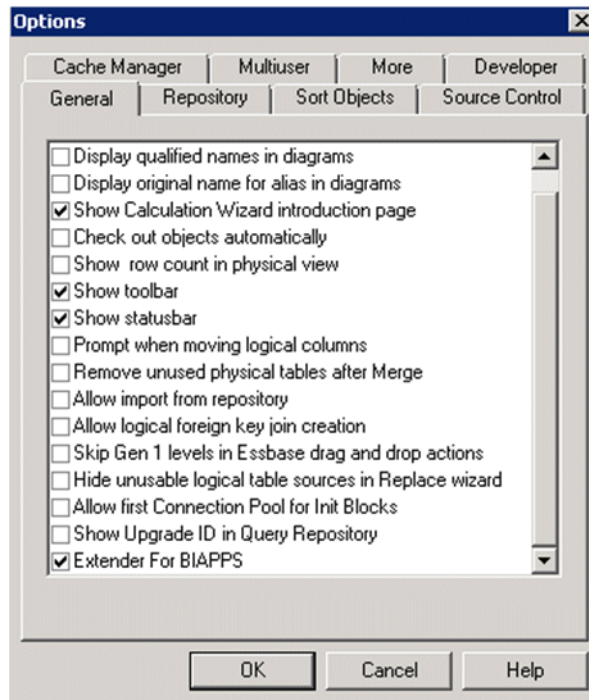

To configure GL Segments and GL Account using the BI Extension process:

1. Using Oracle BI Administration Tool, open the RPD file (for example, OracleBIAnalyticsApps.rpd).

2. Navigate to the Oracle ADF database in the physical layer:

   ```
   oracle.apps.fscm.model.analytics.applicationModule.FscmTopModelAM_
   FscmTopModelAMLocal
   ```

   Then right click on the connection pool and select Import Metadata.

3. In the 'Select Metadata Objects' dialog, do the following:

   a. Ensure that you have selected the 'Automatically include any missing joined objects' radio button.

   b. Click on the 'Synchronize with data source' icon, as shown in the example screenshot below.
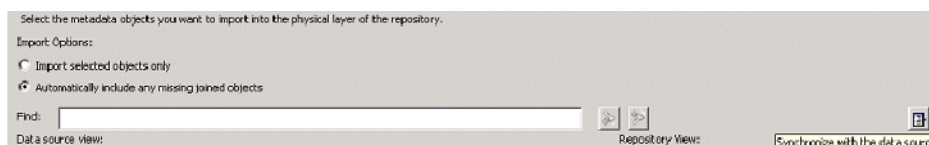
   These settings import all VOs that need to be mapped to the logical tables in the RPD based on the mapping done between the segment labels and the BI Objects as mentioned in Appendix B.27.2.1, "Mapping the segment labels to BI Objects in Fusion Applications". Click on Next after you complete the import.
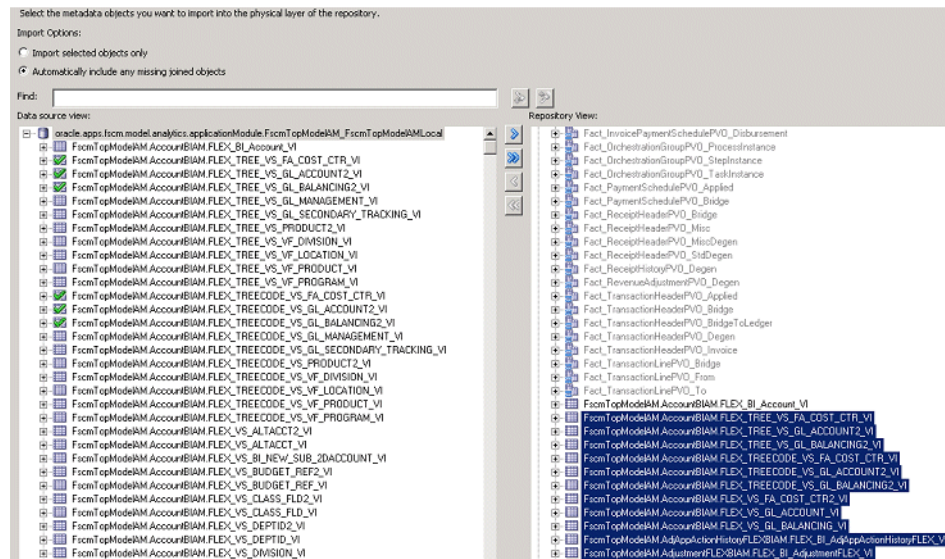
Note: When some complex Chart of Account structures are defined in Fusion Applications, more than one VO might be generated for the same segment label. In this case you will see a warning message as shown in the screenshot below. Please take a note of the information posted in the message as this may be needed to do some manual configuration in subsequent steps. Click on OK to proceed.

4. In the 'Map to Logical Model' dialog you would see that the VOs imported in Step 3 are automatically mapped to the appropriate logical tables. You would also see that the logical columns are automatically mapped to the VO columns in the bottom panel.

Validation:

■ For tree based segments, both the Tree and the Tree Code VO should be mapped to the same logical table. The 'Hierarchy' option should be checked for both.

- For non-tree based segments, 'Hierarchy' option should not be checked.

- FscmTopModelAM.AccountBIAM.FLEX_BI_Account_VI is mapped to 'Dim – GL Account'.

- For the VOs that are mapped to logical tables, the necessary VO columns are also mapped to appropriate logical columns.

  Note: If you received the warning message in step 4, then none of the VOs mentioned in the message are mapped to a logical table. If you want to map these VOs in Oracle BI Applications, then you need to map to one of the generic GL segment dimensions (Dim – GL Segmentn) manually at this stage. For each of the VOs that you manually map at this step, you also need to map the corresponding columns in FscmTopModelAM.AccountBIAM.FLEX_BI_Account_VI to the appropriate logical column in "Dim – GL Account".

5. When you have validated your mappings, click on Next and this will take you to the 'Publish to Warehouse' dialog. Provide the necessary details and click on Finish to complete the extension process.

   Data Warehouse

   <RPD physical layer database object which is 'Oracle Data Warehouse' by default>

   Informatica

   User Name – <Informatica Repository User Name>

   Password - <Informatica Repository Password>

   Database

   User Name – <Informatica Repository DB User Name>

   Password - <Informatica Repository DB Password>

   DAC

   User Name – <DAC Repository DB User Name>

   Password - <DAC Repository DB Password>

6. Validate and save your changes.

7. Validation: If you have successfully completed the extension process, you will see new mappings in the Informatica repository to populate the necessary tables. You will also see new DAC tasks corresponding to these mappings. The Informatica and DAC tasks will be named with the following naming convention SDE_<Logical Table Name>_<Physical Target Name>.

   a. If a segment is mapped from a Tree and a Tree Code VO, then there will be two mappings generated one for loading the segment dimension and the other for the hierarchy dimension. For e.g. SDE_Dim_Cost_Center_W_COST_CENTER_D and SDE_Dim_Cost_Center_W_COST_CENTER_DH.

    b. If a segment is mapped from a Value Set VO, then there will be one mapping generated for loading the segment dimension. For e.g. SDE_Dim_GL_ Segment1_W_GL_SEGMENT_D.

    c. For the GL Account extension, you would see that the Informatica mapping SDE_FUSION_GLAccountDimension will be extended to populate the new columns that were mapped in the previous steps.

8. When you have completed the BI Extension process, open your DAC repository and reassemble all the 'Financials' Subject Areas and rebuild the Execution Plan. After rebuild, validate the new DAC tasks created above are included in the 'Ordered Tasks' tab for the Execution Plan.

B.27.2.3 Validating the Logical Table Source Filters for Generic Segment Divisions

The RPD metadata contains multiple logical tables that represent the generic GL segments, such as Dim – GL Segment1, Dim – GL Segment2 and so on. Since these logical tables are mapped to the same physical table, W_GL_SEGMENT_D, a filter should be specified in the logical table source of these logical tables to restrain the output of the logical table to represent only that particular segment.These filters must be applied on the physical column SEGMENT_LOV_ID to the Value Set Codes that are applicable for that particular segment.

The extension process applies the content Logical Table Source filters for all the generic Dim – GL Segmentn dimensions mapped in the previous steps. You can validate to check if the filters are applied accordingly and save your changes.

To validate the Logical Table Source Filters for Generic Segment Divisions:

1. Using Oracle BI Administration Tool, open the OracleBIAnalyticsApps.rpd.

2. In the Business Model and Mapping layer, double click on 'Dim – GL Segmentn' and open the LTS. Navigate to the Content table and there you would be able to see the filters applied, which should look similar to the filters in the example below.
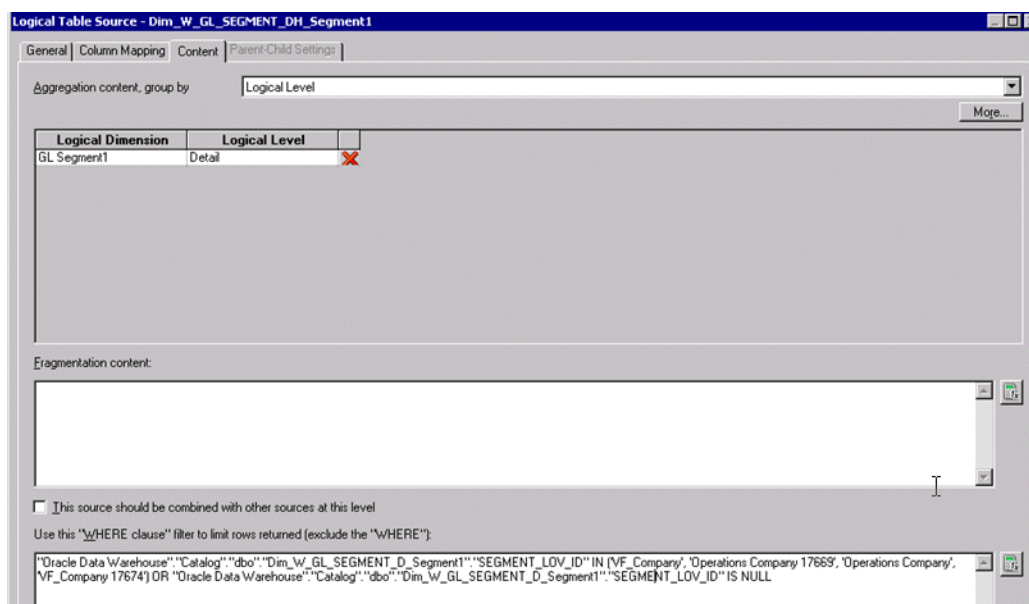
Note: You can find the list of value set codes for a particular segment by opening the segment VO table object in the physical layer of the RPD. It will be stored in the 'description' field of the table object.

B.27.2.4 Re-configuring Segment Dimensions

While configuring the segment dimensions as described in the first section, if you had mapped an incorrect VO to the segment dimension and generated the Informatica and DAC metadata, you must revert the changes and re-map using the correct VO(s).

To re-configure Segment Dimensions:

1. Delete the existing VO Logical Table Source from the corresponding logical table to bring it to the initial state.

2. Delete the Logical Table Source filters if any applied on the DW Logical Table Source (only for the generic segment dimensions).

3. Import the new VO (re-import if the VO already exists in the physical layer) and re-do the extension process as mentioned in the previous sections.

4. If the process completes successfully, then you will see that the previously created Informatica mapping will be deleted and a new mapping with the new VO will be created in its place.

B.28 How to Set Up GL Balance Segment Aggregates

Aggregated GL balances are populated in W_GL_BALANCE_A. The GL balances are not summarized by non-qualifier segments as installed by default. If you want to include the non-qualifier segments, then you must modify the SQL statements as follows.

To Set Up GL Balance Segment Aggregates

1. Open Informatica Designer.

2. Open PLP_GLBalanceAggrByAcctSegCodes.

3. Open mplt_GLBalanceAggrByAcctSegCodes.

4. In the SQL code, replace 'NULL' with the GL segments that you want to use to calculate aggregate the balances.

   For example:

```
SELECT
..
W_GL_BALANCE_F.TENANT_ID,
W_GL_BALANCE_F.TRANSLATED_FLAG,
W_GL_ACCOUNT_D.ACCOUNT_SEG1_CODE,   -- NULL
W_GL_ACCOUNT_D.ACCOUNT_SEG1_ATTRIB, -- NULL
W_GL_ACCOUNT_D.ACCOUNT_SEG2_CODE,    -- NULL
W_GL_ACCOUNT_D.ACCOUNT_SEG2_ATTRIB, -- NULL
NULL,
NULL
FROM
W_GL_BALANCE_F,
...
W_GL_ACCOUNT_D,
...
WHERE
...
GROUP BY
...
W_GL_BALANCE_F.X_CUSTOM,
W_GL_BALANCE_F.TRANSLATED_FLAG,
W_GL_ACCOUNT_D.ACCOUNT_SEG1_CODE,    -- NULL
```

```
W_GL_ACCOUNT_D.ACCOUNT_SEG1_ATTRIB, -- NULL
W_GL_ACCOUNT_D.ACCOUNT_SEG2_CODE,    -- NULL
W_GL_ACCOUNT_D.ACCOUNT_SEG2_ATTRIB, -- NULL
NULL,
NULL
```

5. Open Informatica Workflow Manager.

6. Open session PLP_GLBalanceAggrByAcctSegCodes_Full.

7. Click the Mapping tab.

8. Click Source mplt_GLbalanceAggrByAcctSegCodes.SQ_W_GL_BALANCE_F.

9. Open SQL editor for 'Sql Query'.

10. Modify the override SQL by following step 4.

B.29 How to Integrate Financial Analytics with Project Analytics

You can enable Oracle Financial Analytics to use dimension tables in Oracle Project Analytics. You can only perform this integration if you have licensed Oracle Project Analytics.

You can configure the following Oracle Financial Analytics Subject Areas to join to certain Project Dimensions:

- Financials - Payables (Project, Task, Financial Resource and Expenditure Organization Dimensions)

- Financials - Receivables (Contract Dim)

The following Oracle Financial Analytics fact tables integrate with Project Analytics dimensions:

- W_AP_XACT_F

- W_AR_XACT_F

- W_AR_AGING_INVOICE_A

To Enable Project Analytics Integration with Financial Subject Areas

1. In DAC, select the Oracle Fusion Applications source system container for the version of the OLTP that you are using.

2. Select the Subject Area listed above, and then in the Configuration Tags tab, clear the Inactive check box for the Enable Project Dimensions configuration tag.

3. Assemble the Subject Area.

4. Click the Execute button and build the Execution Plan for the Subject Area that you updated.

B.30 How to Perform RPD Modifications for Cost and Revenue Time Grain Changes

This topic explains how to configure the grain of Cost aggregate (W_PROJ_COST_A) and Revenue aggregate (W_PROJ_REVENUE_A) to Period, Quarter, or Year. As installed by default, the grain of the cost aggregate and revenue aggregate are set at Fiscal Quarter. However, you can modify the grain of the aggregate to either Period or Quarter or Year. This is done by configuring the Parameter COST_TIME_GRAIN and

REVENUE_TIME_GRAIN to Period or Quarter or Year. In addition, you must make the metadata repository changes that are included in this section.

Note: This section only covers the metadata repository changes that you must make using Oracle BI Administration Tool.

This task involves the following sub-tasks:

1. "Changing the Time Grain of the Cost Aggregate table to Fiscal/Project/Enterprise Period".

2. "Changing the Time Grain of the Revenue Aggregate table to Fiscal/Project/Enterprise Period".

3. "Changing the Time grain of the Cost Aggregate table to Fiscal/Project/Enterprise Quarter".

4. "Changing the Time grain of the Revenue Aggregate table to Fiscal/Project/Enterprise Quarter".

5. "Changing the Time grain of the Cost Aggregate table to Fiscal/Project/Enterprise Year".

6. "Changing the Time grain of the Revenue Aggregate table to Fiscal/Project/Enterprise Year".

Changing the Time Grain of the Cost Aggregate table to Fiscal/Project/Enterprise Period

If the grain of Cost aggregate is at period level, the following metadata changes should be made for the Fiscal, Project, and Enterprise calendars:

1. Delete the joins to Dim_W_MCAL_QTR_D_Fiscal_Quarter/ Dim_W_MCAL_ QTR_D_Project_Quarter/Dim_W_ENT_QTR_D.
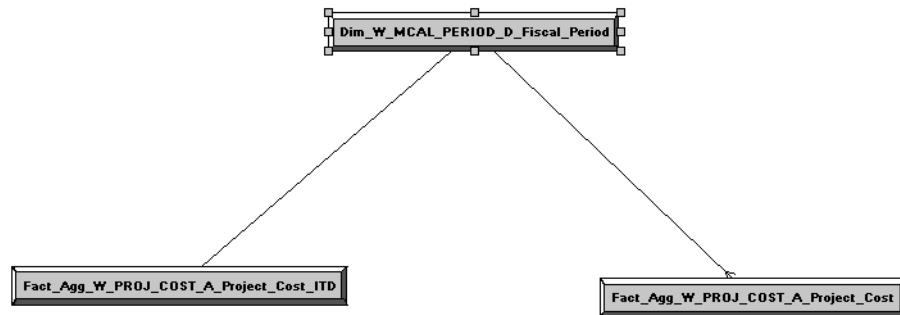
   Delete the existing physical joins between Fact_Agg_W_PROJ_COST_A_Project_ Cost (under logical fact 'Fact – Project Cost') to Dim_W_MCAL_QTR_D_Fiscal_ Quarter (under logical dimension 'Dim – Date Fiscal Calendar'), Dim_W_MCAL_ QTR_D_Project_Quarter (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_QTR_D (under logical dimension 'Dim - Date').

   Keeping these joins will result in a loss of records when the report is at the Quarter level, because some records in the Aggregate table have their GL_ACCT_PERIOD_ START_DAY_WID and ENT_QTR_START_DT_WID coincide not with the Quarter Start Date but rather with a Period Start Date.

2. Create joins to Dim_W_MCAL_PERIOD_D_Fiscal_Period.

   Additional physical joins need to be created between following facts:

   ```
   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD
   Fact_Agg_W_PROJ_COST_A_Project_Cost
   Dim_W_MCAL_PERIOD_D_Fiscal_Period
   ```

In the Business Model and Mapping layer, select the 'Dim_W_MCAL_PERIOD_D_
Fiscal_Period' Logical Table Source from the 'Dim - Date Fiscal Calendar' and the
'Fact_Agg_W_PROJ_COST_A_Project_Cost' and 'Fact_Agg_W_PROJ_COST_A_
Project_Cost_ITD' Logical Table Sources in 'Fact - Project Cost and then right click
and select 'physical diagram->selected objects only' and create the following
physical join, then click OK.

```
Dim_W_MCAL_PERIOD_D_Fiscal_Period.MCAL_PERIOD_START_DAY_WID = Fact_Agg_W_PROJ_
COST_A_Project_Cost.GL_ACCT_PERIOD_START_DAY_WID
```


Create the following complex join between Dim_W_MCAL_PERIOD_D_Fiscal_
Period and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD:

```
Dim_W_MCAL_PERIOD_D_Fiscal_Period.MCAL_PERIOD_END_DAY_WID >= Fact_Agg_W_PROJ_
COST_A_Project_Cost_ITD.GL_ACCT_PERIOD_END_DAY_WID
AND Dim_W_MCAL_PERIOD_D_Fiscal_Period.MCAL_CAL_WID = Fact_Agg_W_PROJ_COST_A_
Project_Cost_ITD.GL_MCAL_CAL_WID
```

3. Create joins to Dim_W_MCAL_PERIOD_D_Project_Period.

Create additional physical joins between following facts:

```
Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD
Fact_Agg_W_PROJ_COST_A_Project_Cost
Dim_W_MCAL_PERIOD_D_Project_Period
```


Create the following physical join between Dim_W_MCAL_PERIOD_D_Project_
Period and Fact_Agg_W_PROJ_COST_A_Project_Cost:

```
Dim_W_MCAL_PERIOD_D_Project_Period.MCAL_PERIOD_START_DAY_WID = Fact_Agg_W_PROJ_
COST_A_Project_Cost.PROJ_ACCT_PERIOD_START_DAY_WID
```

Create the following complex join between Dim_W_MCAL_PERIOD_D_Project_
Period and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD:

```
Dim_W_MCAL_PERIOD_D_Project_Period.MCAL_PERIOD_END_DAY_WID >= Fact_Agg_W_PROJ_
COST_A_Project_Cost_ITD.PROJ_ACCT_PERIOD_END_DAY_WID
AND Dim_W_MCAL_PERIOD_D_Project_Period.MCAL_CAL_WID = Fact_Agg_W_PROJ_COST_A_
Project_Cost_ITD.PROJ_MCAL_CAL_WID
```

4. Create joins to Dim_W_ENT_PERIOD_D.

   Additional physical joins need to be created between following facts:

   ```
   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD
   Fact_Agg_W_PROJ_COST_A_Project_Cost
   Dim_W_ENT_PERIOD_D
   ```


   In the Business Model and Mapping layer, select the 'Dim_W_ENT_PERIOD_D'
   Logical Table Source from the 'Dim - Date' and the 'Fact_Agg_W_PROJ_COST_A_
   Project_Cost' and 'Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD' Logical Table
   Sources in 'Fact - Project Cost and then right click and select 'physical
   diagram->selected objects only' and create the following physical join, then click
   OK.

   ```
   'Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_PERIOD_D"."ENT_PERIOD_START_
   DT_WID" = "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_COST_A_
   Project_Cost"."ENT_PERIOD_START_DAY_WID"
   ```

Create the following complex join between Dim_W_ENT_PERIOD_D and Fact_
Agg_W_PROJ_COST_A_Project_Cost_ITD:

```
"Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_COST_A_Project_Cost_
ITD"."ENT_PERIOD_END_DAY_WID" <= "Oracle Data Warehouse"."Catalog"."dbo"."Dim_
W_ENT_PERIOD_D"."ENT_PERIOD_END_DT_WID"
```


5. Change the Content Aggregation Level in the Business Model and Mapping layer.

As installed by default, the grain for cost aggregate is set to Fiscal Quarter against
the dimensions Dim-Date Fiscal Calendar and Dim-Date Project Calendar. This
must be modified to Period for both Fact_Agg_W_PROJ_COST_A_Project_Cost
and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD Logical Table Sources. In the
Business Model and Mapping layer open these two Logical Table Sources in 'Fact –
Project Cost'.

Ensure that there are joins between Fact_Agg_W_PROJ_COST_A_Project_Cost_
ITD Logical Table Source in 'Fact - Project Cost' to Dim_W_MCAL_QTR_D_Fiscal_
Quarter/ Dim_W_MCAL_YEAR_D_Fiscal_Year Logical Table Sources from the
'Dim - Date Fiscal Calendar', Dim_W_MCAL_QTR_D_Project_Quarter/ Dim_W_
MCAL_YEAR_D_Project_Year Logical Table Sources from the 'Dim - Date Project
Calendar' and Dim_W_ENT_PERIOD_D /Dim_W_ENT_YEAR_D Logical Table
Sources from 'Dim - Date'. These are done by default.

6. Save the changes.

When these changes are complete, run the Consistency Check and ensure that
there are no errors, save the RPD file, and clear Oracle BI Enterprise Edition Cache.
If you are making the changes in offline mode, then restart the Oracle BI Server
and Oracle BI Presentation Services.

**Changing the Time Grain of the Revenue Aggregate table to
Fiscal/Project/Enterprise Period**

If the grain of the Revenue aggregate is at period level, then the following metadata
changes should be made for the Fiscal, Project, and Enterprise calendars:

1. Delete the joins to Dim_W_MCAL_QTR_D_Fiscal_Quarter/ Dim_W_MCAL_
QTR_D_Project_Quarter.

Delete the existing physical joins between Fact_Agg_W_PROJ_REVENUE_A_
Revenue (under logical fact 'Fact – Project Revenue') to Dim_W_MCAL_QTR_D_
Fiscal_Quarter (under logical dimension 'Dim – Date Fiscal Calendar') and Dim_
W_MCAL_QTR_D_Project_Quarter (under logical dimension 'Dim – Date Project
Calendar').

Keeping these joins will result in a loss of records when the report is at the Quarter
level, because some records in the Aggregate table have their GL_ACCT_PERIOD_
START_DAY_WID coincide not with the Quarter Start Date but rather with a
Period Start Date.

2. Create joins to Dim_W_MCAL_PERIOD_D_Fiscal_Period.

Create additional physical joins between following facts:

```
Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD
Fact_Agg_W_PROJ_REVENUE_A_Revenue
Dim_W_MCAL_PERIOD_D_Fiscal_Period
```

In the Business Model and Mapping layer, select the 'Dim_W_MCAL_PERIOD_D_Fiscal_Period' Logical Table Source from the 'Dim - Date Fiscal Calendar' and the 'Fact_Agg_W_PROJ_REVENUE_A_REVENUE' and 'Fact_Agg_W_PROJ_REVENUE_A_Revenue _ITD' Logical Table Sources in 'Fact - Project Revenue and then right click and select 'physical diagram->selected objects only' and create the following physical join, then click OK.

```
Dim_W_MCAL_PERIOD_D_Fiscal_Period.MCAL_PERIOD_START_DAY_WID = Fact_Agg_W_PROJ_
REVENUE_A_Revenue.GL_ACCT_PERIOD_START_DAY_WID
```

Create the following complex join between Dim_W_MCAL_PERIOD_D_Fiscal_
Period and Fact_Agg_W_PROJ_ REVENUE_A_Revenue _ITD:

```
Dim_W_MCAL_PERIOD_D_Fiscal_Period.MCAL_PERIOD_END_DAY_WID >= Fact_Agg_W_PROJ_
REVENUE_A_Revenue_ITD.GL_ACCT_PERIOD_END_DAY_WID
AND Dim_W_MCAL_PERIOD_D_Fiscal_Period.MCAL_CAL_WID = Fact_Agg_W_PROJ_REVENUE_A_
Revenue_ITD.GL_MCAL_CAL_WID
```


3. Create joins to Dim_W_MCAL_PERIOD_D_Project_Period.

Create additional physical joins between following facts:

```
Fact_Agg_W_PROJ_ REVENUE_A_Revenue _ITD
Fact_Agg_W_PROJ_ REVENUE_A_Revenue
Dim_W_MCAL_PERIOD_D_Project_Period
```


Create the following physical join between Dim_W_MCAL_PERIOD_D_Project_
Period and Fact_Agg_W_PROJ_REVENUE_A_Revenue:

```
Dim_W_MCAL_PERIOD_D_Project_Period.MCAL_PERIOD_START_DAY_WID = Fact_Agg_W_PROJ_
REVENUE_A_Revenue.PROJ_ACCT_PERIOD_START_DAY_WID
```

Create the following complex join between Dim_W_MCAL_PERIOD_D_Project_
Period and Fact_Agg_W_PROJ_REVENUE_A_Revenue _ITD:

```
Dim_W_MCAL_PERIOD_D_Project_Period.MCAL_PERIOD_END_DAY_WID >= Fact_Agg_W_PROJ_
REVENUE_A_Revenue_ITD.PROJ_ACCT_PERIOD_END_DAY_WID
AND Dim_W_MCAL_PERIOD_D_Project_Period.MCAL_CAL_WID = Fact_Agg_W_PROJ_REVENUE_
A_Revenue_ITD.PROJ_MCAL_CAL_WID
```


4. Create joins to Dim_W_ENT_PERIOD_D.

Create additional physical joins between following facts:

```
Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD
```

```
Fact_Agg_W_PROJ_REVENUE_A_Revenue
Dim_W_ENT_PERIOD_D
```


In the Business Model and Mapping layer, select the 'Dim_W_ENT_PERIOD_D'
Logical Table Source from the 'Dim - Date' and the 'Fact_Agg_W_PROJ_
REVENUE_A_Revenue' and Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD'
Logical Table Sources in 'Fact - Project Revenue', and then right click and select
'physical diagram->selected objects only' and create the following physical join,
then click OK:

```
"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_PERIOD_D"."ENT_PERIOD_START_
DT_WID" = "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_
Revenue"."ENT_PERIOD_START_DAY_WID"
```


Create the following complex join between Dim_W_ENT_PERIOD_D and Fact_
Agg_W_PROJ_REVENUE_A_Revenue_ITD:

```
"Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_Revenue_
ITD"."ENT_PERIOD_END_DAY_WID" <=
"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_PERIOD_D"."ENT_PERIOD_END_
DT_WID"
```

Physical Foreign Key - Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD_FKey#6

Name: Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD_FKey#6

Table: Dim_W_ENT_PERIOD_D

Table: Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD

Column:
Name
ENT_PERIOD_END_DT_WID
ROW_WID
ENT_HALF
ENT_PERIOD

Operator: =

Column:
Name
CONTRACT_LINE_WID
ROW_WID
PROJECT_WID
GL_ACCT_PERIOD_START_DAY_WID

Driving table: None   Type: Inner

Cardinality: N 0,1 1 Unknown   1 0,1 N Unknown

Hint:

Expression:
"Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD"."ENT_PERIOD_END_DAY_WID" <=
"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_PERIOD_D"."ENT_PERIOD_END_DT_WID"

5. As installed by default, the grain for cost aggregate is set to Fiscal Quarter against the dimensions Dim-Date Fiscal Calendar and Dim-Date Project Calendar. This must be modified to Period for both Fact_Agg_W_PROJ_REVENUE_A_Revenue and Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD Logical Table Sources. In the Business Model and Mapping layer open these two Logical Table Sources in 'Fact – Project Revenue'.

Logical Table Source - Fact_Agg_W_PROJ_REVENUE_A_Revenue

General | Column Mapping | Content | Parent-Child Settings

Aggregation content, group by   Logical Level

Show mapped   Show unmapped   More...

Logical Dimension	Logical Level
Customer	Customer Detail
Customer Account	
Date	Enterprise Period
Date - Fiscal Calendar	Fiscal Period
Date - Projects Calendar	Project Period
Date - Receiver Fiscal Calendar	
Department	

Logical Table Source - Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD

General | Column Mapping | Content | Parent-Child Settings

Aggregation content, group by   Logical Level

Show mapped   Show unmapped   More...

Logical Dimension	Logical Level
Currency	Aggregate Level
Customer	Customer Detail
Customer Account	
Date	Enterprise Period
Date - Fiscal Calendar	Fiscal Period
Date - Projects Calendar	Project Period
Date - Receiver Fiscal Calendar	

Fragmentation content:

Make sure that there are joins between Fact_Agg_W_PROJ_REVENUE_A_ Revenue_ITD Logical Table Source in 'Fact - Project Revenue' to Dim_W_MCAL_ QTR_D_Fiscal_Quarter/ Dim_W_MCAL_YEAR_D_Fiscal_Year Logical Table Sources from the 'Dim - Date Fiscal Calendar', Dim_W_MCAL_QTR_D_Project_ Quarter/ Dim_W_MCAL_YEAR_D_Project_Year Logical Table Sources from the 'Dim - Date Project Calendar', and Dim_W_ENT_PERIOD_D /Dim_W_ENT_ YEAR_D Logical Table Sources from 'Dim - Date'. These are done by default.

6. Save the changes.

When these changes are complete, run the Consistency Check and ensure that there are no errors, save the RPD file, and clear the Oracle BI Enterprise Edition Cache. If you are making the changes in offline mode, then restart the Oracle BI Server and Oracle BI Presentation Services.

Changing the Time grain of the Cost Aggregate table to Fiscal/Project/Enterprise Quarter

This is default configuration. Please ensure that the following joins are in place.

1. Verify the joins to Fiscal Calendar (Dim-Date Fiscal Calendar).

   In the Business Model and Mapping layer, select the Dim_W_MCAL_QTR_D_ Fiscal_Quarter/ Dim_W_MCAL_YEAR_D_Fiscal_Year Logical Table Source from the 'Dim - Date Fiscal Calendar' and the Fact_Agg_W_PROJ_COST_A_Project_ Cost and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD Logical Table Sources in 'Fact - Project Cost and then right click and select 'physical diagram->selected objects only' and verify the following default joins between the following objects:

   ```
   Dim_W_MCAL_QTR_D_Fiscal_Quarter.MCAL_QTR_START_DAY_WID = Fact_Agg_W_PROJ_COST_
   A_Project_Cost.GL_ACCT_PERIOD_START_DAY_WID

   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.GL_ACCT_PERIOD_END_DAY_WID <= Dim_W_
   MCAL_QTR_D_Fiscal_Quarter.MCAL_QTR_END_DAY_WID AND
   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.GL_MCAL_CAL_WID=Dim_W_MCAL_QTR_D_
   Fiscal_Quarter.MCAL_CAL_WID

   Dim_W_MCAL_YEAR_D_Fiscal_Year.MCAL_YEAR_END_DAY_WID
   >=Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.GL_ACCT_PERIOD_END_DAY_WID
   AND Dim_W_MCAL_YEAR_D_Fiscal_Year.MCAL_CAL_WID =
   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.GL_MCAL_CAL_WID
   ```

2. Verify the joins to Project Calendar (Dim-Date Project Calendar).

   In the Business Model and Mapping layer, select the Dim_W_MCAL_QTR_D_ Project_Quarter/ Dim_W_MCAL_YEAR_D_Project_Year Logical Table Source from the 'Dim - Date Project Calendar' and the Fact_Agg_W_PROJ_COST_A_ Project_Cost and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD Logical Table Sources in 'Fact - Project Cost and then right click and select 'physical diagram->selected objects only' and verify the following default joins between the following objects.

   ```
   Dim_W_MCAL_QTR_D_Project_Quarter.MCAL_QTR_START_DAY_WID = Fact_Agg_W_PROJ_COST_
   A_Project_Cost.PROJ_ACCT_PERIOD_START_DAY_WID

   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.PROJ_ACCT_PERIOD_END_DAY_WID
   <= Dim_W_MCAL_QTR_D_Project_Quarter.MCAL_QTR_END_DAY_WID AND
   Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.PROJ_MCAL_CAL_WID=Dim_W_MCAL_QTR_D_
   Project_Quarter.MCAL_CAL_WID

   Dim_W_MCAL_YEAR_D_Project_Year.MCAL_YEAR_END_DAY_WID >= Fact_Agg_W_PROJ_COST_A_
   Project_Cost_ITD.PROJ_ACCT_PERIOD_END_DAY_WID AND Dim_W_MCAL_YEAR_D_Project_
   Year.MCAL_CAL_WID = Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD.PROJ_MCAL_CAL_WID
   ```

3. Verify the joins to Enterprise Calendar (Dim-Date).

   In the Business Model and Mapping layer, select the Dim_W_ENT_QTR_D / Dim_W_ENT_YEAR_D Logical Table Source from the 'Dim - Date' and the Fact_ Agg_W_PROJ_COST_A_Project_Cost and Fact_Agg_W_PROJ_COST_A_Project_ Cost_ITD Logical Table Sources in 'Fact - Project Cost and then right click and

select 'physical diagram->selected objects only' and verify the following default joins between the following objects.

```
"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_QTR_D"."ENT_QTR_START_DT_
WID" = "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_COST_A_Project_
Cost"."ENT_PERIOD_START_DAY_WID"
```

```
"Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_COST_A_Project_Cost_
ITD"."ENT_PERIOD_END_DAY_WID" <=  "Oracle Data Warehouse"."Catalog"."dbo"."Dim_
W_ENT_QTR_D"."ENT_QTR_END_DT_WID"
```

```
"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_YEAR_D"."ENT_YEAR_END_DT_
WID" >= "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_COST_A_
Project_Cost_ITD"."ENT_PERIOD_END_DAY_WID"
```

4. Removing extraneous joins.

Delete any additional time dimension joins between Time Dimension physical table aliases in Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim - Date to Project Cost Fact physical table aliases Fact_Agg_W_PROJ_COST_A_ Project_Cost and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD under Logical cost 'Fact - Project Cost'.

5. Change the Content Aggregation Level in the Business Model and Mapping layer.

By default, the grain for cost aggregate is set to Quarter against the dimensions Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim - Date.

Instead of Fiscal/Project Period you must set this to Fiscal Quarter for Dim – Date Fiscal Calendar, Project Quarter for Dim – Date Project Calendar and Enterprise Quarter for Dim - Date.

6. Save the changes.

When these changes are complete, run the Consistency Check and ensure that there are no errors, save the RPD file, and clear the Oracle BI Enterprise Edition Cache. If you are making the changes in offline mode, then restart the Oracle BI Server and Oracle BI Presentation Services.

Changing the Time grain of the Revenue Aggregate table to Fiscal/Project/Enterprise Quarter

This is default configuration. Please ensure that the following joins are in place.

1. Verify the joins to Fiscal Calendar (Dim-Date Fiscal Calendar).

In the Business Model and Mapping layer, select the Dim_W_MCAL_QTR_D_ Fiscal_Quarter/ Dim_W_MCAL_YEAR_D_Fiscal_Year Logical Table Source from the 'Dim - Date Fiscal Calendar' and the Fact_Agg_W_PROJ_REVENUE_A_ Revenue and Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD Logical Table Sources in 'Fact - Project Revenue and then right click and select 'physical diagram->selected objects only' and verify the following joins between the following objects.

```
Dim_W_MCAL_QTR_D_Fiscal_Quarter.MCAL_QTR_START_DAY_WID =
Fact_Agg_W_PROJ_REVENUE_A_Revenue.GL_ACCT_PERIOD_START_DAY_WID
```

```
Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.GL_ACCT_PERIOD_END_DAY_WID
<= Dim_W_MCAL_QTR_D_Fiscal_Quarter.MCAL_QTR_END_DAY_WID AND
Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.GL_MCAL_CAL_WID=Dim_W_MCAL_QTR_D_Fiscal_
Quarter.MCAL_CAL_WID
```

```
Dim_W_MCAL_YEAR_D_Fiscal_Year.MCAL_YEAR_END_DAY_WID
```

```
>=Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.GL_ACCT_PERIOD_END_DAY_WID
AND Dim_W_MCAL_YEAR_D_Fiscal_Year.MCAL_CAL_WID =
Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.GL_MCAL_CAL_WID
```

2. Verify the joins to Project Calendar (Dim-Date Project Calendar).

   In the Business Model and Mapping layer, select the Dim_W_MCAL_QTR_D_
   Project_Quarter/ Dim_W_MCAL_YEAR_D_Project_Year Logical Table Source
   from the 'Dim - Date Project Calendar' and the Fact_Agg_W_PROJ_REVENUE_A_
   Revenue and Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD Logical Table
   Sources in 'Fact - Project Revenue and then right click and select 'physical
   diagram->selected objects only' and verify the following joins between the
   following objects.

```
Dim_W_MCAL_QTR_D_Project_Quarter.MCAL_QTR_START_DAY_WID =
Fact_Agg_W_PROJ_REVENUE_A_Revenue.PROJ_ACCT_PERIOD_START_DAY_WID

Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.PROJ_ACCT_PERIOD_END_DAY_WID
<= Dim_W_MCAL_QTR_D_Project_Quarter.MCAL_QTR_END_DAY_WID AND
Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.PROJ_MCAL_CAL_WID=Dim_W_MCAL_QTR_D_
Project_Quarter.MCAL_CAL_WID

Dim_W_MCAL_YEAR_D_Project_Year.MCAL_YEAR_END_DAY_WID >= Fact_Agg_W_PROJ_
REVENUE_A_Revenue_ITD.PROJ_ACCT_PERIOD_END_DAY_WID AND Dim_W_MCAL_YEAR_D_
Project_Year.MCAL_CAL_WID = Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD.PROJ_MCAL_
CAL_WID
```

3. Verify the joins to Enterprise Calendar (Dim-Date).

   In the Business Model and Mapping layer, select the Dim_W_ENT_QTR_D /
   Dim_W_ENT_YEAR_D Logical Table Source from the 'Dim - Date' and the Fact_
   Agg_W_PROJ_REVENUE_A_Revenue and Fact_Agg_W_PROJ_REVENUE_A_
   Revenue_ITD Logical Table Sources in 'Fact - Project Revenue and then right click
   and select 'physical diagram->selected objects only' and verify the following joins
   between the following objects.

```
"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_QTR_D"."ENT_QTR_START_DT_
WID" = "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_
Revenue"."ENT_PERIOD_START_DAY_WID"

"Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_Revenue_
ITD"."ENT_PERIOD_END_DAY_WID" <= "Oracle Data Warehouse"."Catalog"."dbo"."Dim_
W_ENT_QTR_D"."ENT_QTR_END_DT_WID"

"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_YEAR_D"."ENT_YEAR_END_DT_
WID" >= "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_
Revenue_ITD"."ENT_PERIOD_END_DAY_WID"
```

4. Removing extraneous joins.

   Delete any additional time dimension joins between Time Dimension physical
   table aliases in Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim –
   Date to Project Revenue Fact physical table aliases Fact_Agg_W_PROJ_
   REVENUE_A_Revenue and Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD
   under Logical cost 'Fact - Project Revenue'.

5. Change the Content Aggregation Level in the Business Model and Mapping layer.

   By default, the grain for revenue aggregate is set to Quarter against the
   dimensions Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim - Date.

Instead of Fiscal/Project Period, you must set this to Fiscal Quarter for Dim – Date Fiscal Calendar, Project Quarter for Dim – Date Project Calendar and Enterprise Quarter for Dim - Date.

6. Save the changes.

When these changes are complete, run the Consistency Check and ensure that there are no errors, save the RPD file, and clear the Oracle BI Enterprise Edition Cache. If you are making the changes in offline mode, then restart the Oracle BI Server and Oracle BI Presentation Services.

Changing the Time grain of the Cost Aggregate table to Fiscal/Project/Enterprise Year

1. Delete joins to Dim_W_MCAL_QTR_D_Fiscal_Quarter/ Dim_W_MCAL_QTR_D_ Project_Quarter/ Dim_W_ENT_QTR_D.

   Delete the existing physical joins between Fact_Agg_W_PROJ_COST_A_Project_ Cost (under logical fact 'Fact – Project Cost') to Dim_W_MCAL_QTR_D_Fiscal_ Quarter (under logical dimension 'Dim – Date Fiscal Calendar'), Dim_W_MCAL_ QTR_D_Project_Quarter (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_QTR_D (under logical dimension 'Dim – Date').

   Delete the existing physical joins between Fact_Agg_W_PROJ_COST_A_Project_ Cost_ITD (under logical fact 'Fact – Project Cost') to Dim_W_MCAL_QTR_D_ Fiscal_Quarter (under logical dimension 'Dim – Date Fiscal Calendar'), Dim_W_ MCAL_QTR_D_Project_Quarter (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_QTR_D (under logical dimension 'Dim – Date').

2. Create joins to Dim_W_MCAL_YEAR_D_Fiscal_Year/ Dim_W_MCAL_YEAR_D_ Project_Year/ Dim_W_ENT_YEAR_D.

   Additional physical joins need to be created between following Logical Table Source fact Fact_Agg_W_PROJ_COST_A_Project_Cost (under logical fact 'Fact – Project Cost') and Dim_W_MCAL_YEAR_D_Fiscal_Year (under logical dimension 'Dim – Date Fiscal Calendar'), Dim_W_MCAL_YEAR_D_Project_Year (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_YEAR_D (under logical dimension 'Dim – Date').

   The following joins need to be created (foreign key join):

   ```
   Dim_W_MCAL_YEAR_D_Fiscal_Year.MCAL_YEAR_START_DAY_WID =
   Fact_Agg_W_PROJ_COST_A_Project_Cost.GL_ACCT_PERIOD_START_DAY_WID

   Dim_W_MCAL_YEAR_D_Project_Year.MCAL_YEAR_START_DAY_WID =
   Fact_Agg_W_PROJ_COST_A_Project_Cost.PROJ_ACCT_PERIOD_START_DAY_WID

   "Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_YEAR_D"."ENT_YEAR_START_DT_
   WID" = "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_COST_A_Project_
   Cost"."ENT_PERIOD_START_DAY_WID"
   ```

3. Verify the joins to Dim_W_MCAL_YEAR_D_Fiscal_Year/ Dim_W_MCAL_YEAR_ D_Project_Year/ Dim_W_ENT_YEAR_D.

   Ensure that there are joins between Fact_Agg_W_PROJ_COST_A_Project_Cost_ ITD Logical Table Source in 'Fact - Project Cost' to Dim_W_MCAL_YEAR_D_ Fiscal_Year Logical Table Source from the 'Dim - Date Fiscal Calendar', Dim_W_ MCAL_YEAR_D_Project_Year Logical Table Source from the 'Dim - Date Project Calendar' and Dim_W_ENT_YEAR_D Logical Table Source from the 'Dim - Date'.These are done by default.

Delete any additional time dimension joins between Time Dimension Logical Table Sources in Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim – Date to Logical Table Sources Fact_Agg_W_PROJ_COST_A_Project_Cost and Fact_Agg_W_PROJ_COST_A_Project_Cost_ITD under Logical cost 'Fact - Project Cost'.

4. Changing Content Aggregation Level in the Business Model and Mapping layer

As installed by default, the grain for cost aggregate is set to Quarter against the dimensions Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim - Date.

Instead of Fiscal/Project Period you must set this to Fiscal Year for Dim – Date Fiscal Calendar, Project Year for Dim – Date Project Calendar and Enterprise year for Dim - Date.

5. Save the changes.

When these changes are complete, run the Consistency Check and ensure that there are no errors, save the RPD file, and clear the Oracle BI Enterprise Edition Cache. If you are making the changes in offline mode, then restart the Oracle BI Server and Oracle BI Presentation Services.

Changing the Time grain of the Revenue Aggregate table to Fiscal/Project/Enterprise Year

1. Delete the joins to Dim_W_MCAL_QTR_D_Fiscal_Quarter/ Dim_W_MCAL_QTR_D_Project_Quarter.

Delete the existing physical joins between Fact_Agg_W_PROJ_REVENUE_A_Revenue (under logical fact 'Fact – Project Revenue') to Dim_W_MCAL_QTR_D_Fiscal_Quarter (under logical dimension 'Dim – Date Fiscal Calendar'), Dim_W_MCAL_QTR_D_Project_Quarter (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_QTR_D (under logical dimension 'Dim – Date').

Delete the existing physical joins between Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD (under logical fact 'Fact – Project Revenue') to Dim_W_MCAL_QTR_D_Fiscal_Quarter (under logical dimension 'Dim – Date Fiscal Calendar', Dim_W_MCAL_QTR_D_Project_Quarter (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_QTR_D (under logical dimension 'Dim – Date').

2. Create joins to Dim_W_MCAL_YEAR_D_Fiscal_Year/ Dim_W_MCAL_YEAR_D_Project_Year.

Additional physical joins need to be created between following Logical Table Source fact Fact_Agg_W_PROJ_REVENUE_A_Revenue (under logical fact 'Fact – Project Cost') and Dim_W_MCAL_YEAR_D_Fiscal_Year (under logical dimension 'Dim – Date Fiscal Calendar'), Dim_W_MCAL_YEAR_D_Project_Year (under logical dimension 'Dim – Date Project Calendar') and Dim_W_ENT_YEAR_D (under logical dimension 'Dim – Date').

The following joins need to be created (foreign key join):

```
Dim_W_MCAL_YEAR_D_Fiscal_Year.MCAL_YEAR_START_DAY_WID =
Fact_Agg_W_PROJ_REVENUE_A_Revenue.GL_ACCT_PERIOD_START_DAY_WID

Dim_W_MCAL_YEAR_D_Project_Year.MCAL_YEAR_START_DAY_WID =
Fact_Agg_W_PROJ_REVENUE_A_Revenue.PROJ_ACCT_PERIOD_START_DAY_WID

"Oracle Data Warehouse"."Catalog"."dbo"."Dim_W_ENT_YEAR_D"."ENT_YEAR_START_DT_
WID" = "Oracle Data Warehouse"."Catalog"."dbo"."Fact_Agg_W_PROJ_REVENUE_A_
Revenue"."ENT_PERIOD_START_DAY_WID"
```

3. Verify the joins to Dim_W_MCAL_YEAR_D_Fiscal_Year/ Dim_W_MCAL_YEAR_D_Project_Year.

Ensure that there are joins between Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD Logical Table Source in 'Fact - Project Revenue' to Dim_W_MCAL_YEAR_D_Fiscal_Year Logical Table Source from the 'Dim - Date Fiscal Calendar', Dim_W_MCAL_YEAR_D_Project_Year Logical Table Source from the 'Dim - Date Project Calendar' and Dim_W_ENT_YEAR_D Logical Table Source from the 'Dim - Date'. These are done by default.

Delete any additional time dimension joins between Time Dimension Logical Table Source in Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim – Date to Logical Table Sources Fact_Agg_W_PROJ_REVENUE_A_Revenue and Fact_Agg_W_PROJ_REVENUE_A_Revenue_ITD under Logical cost 'Fact - Project Revenue'.

4. Changing Content Aggregation Level in the Business Model and Mapping layer

As installed by default, the grain for revenue aggregate is set to Quarter against the dimensions Dim-Date Fiscal Calendar, Dim-Date Project Calendar and Dim - Date.

Instead of Fiscal/Project Period you must set this to Fiscal Year for Dim – Date Fiscal Calendar and Project Year for Dim – Date Project Calendar and Enterprise year for Dim - Date.

5. Save the changes.

When these changes are complete, run the Consistency Check and ensure that there are no errors, save the RPD file, and clear the Oracle BI Enterprise Edition Cache. If you are making the changes in offline mode, then restart the Oracle BI Server and Oracle BI Presentation Services.

B.31 How to Assign Group Account Numbers to Natural Accounts for Human Resources Analytics

You must assign the following group account numbers in Fusion Applications for HR: (You can skip this task if you have already completed this task for General Ledger.)

- 'CONT EXP' for 'Contracting Expenses'
- 'COGS' for 'Cost of Goods Sold'
- 'DEPCN' for 'Depreciation Expenses'
- 'EMP BENFT' for 'Employee Benefits Related Expenses'
- 'EMP OVERTIME' for 'Employee Overtime Expenses'
- 'EMP SUPP' for Employee Support Expenses'
- 'GEN PAYROLL' for General Admin and Other Payroll'
- 'MISC OPER EXP' for 'Miscellaneous Operating Expenses'
- 'MKTG PAYROLL' for 'Payroll Expenses'
- 'SLS PAYROLL' for 'Payroll Expenses'
- 'R&D PAYROLL' for 'Payroll Expenses' (GEN PAYROLL is already listed)
- 'VARIANCE EXP' for 'Product Variance Expenses'
- 'REVENUE' for 'Revenue'

Note: 'Other Operating Expenses' is a derived column. It does not need a group account number assignment.

How to Assign Group Account Numbers to Natural Accounts:

1. Login to Fusion Applications.

2. Click the Applcore menu.

3. Identify the value set used for your natural account.

4. Open the window to maintain value set values.

5. Assign financial categories to each natural account from the list of values.

The following group account numbers (financial categories) are seeded:

```
ACC DEPCN - Accumulated Depreciation
ACC LIAB - Accrued Liabilities
AP - Account Payables
AR - Account Receivables
CASH - Cash
CMMN STOCK - Common Stock
COGS - Cost Of Goods Sold
CONT EXP - Contracting Expenses
DEFERRED COGS - Deferred Cost of Goods Sold
DEFERRED REVENUE - Deferred Revenue
DEPCN - Depreciation Expenses
EMP BENFT - Employee Benefits Related Expenses
EMP OVERTIME - Employee Overtime
EMP SUPP - Employee Support and Cafeteria Expenses
FG INV - Finished Goods Inventory, FREIGHT - Freight Expenses
GEN PAYROLL - General Admin And Other Payroll
GOODWILL - Goodwill
INC TAX - Income Tax
INT EXP - Interest Expenses
LT DEBT - Long Term Debt
MISC OPER EXP - Miscellaneous Operating Expenses
MKTG PAYROLL - Marketing Payroll
OTHER ASSET - Other Assets
OTHER CA - Other Current Assets
OTHER CL - Other Current Liabilities
OTHER EQUITY - Other Equity Related
OTHER INC - Other Income
OTHER LIAB - Other Liabilities
OTHER MKTG EXP - Other Marketing Expenses
OTHER R&D EXP - Other R&D Expenses
OTHER SLS EXP - Other Sales Expenses
PPAID EXP - Prepaid Expenses
PPE - PPE
PREF STOCK - Preferred Stock
PURCH - Purch
R&D PAYROLL - R&D Payroll
RET EARNING - Retained Earning
REVENUE - Sales Revenue
RM CONS - RM Cons
RM INV - Raw Material Inventory
SLS PAYROLL - Sales Payroll
ST BORR - ST Borr
TAX LIAB - Tax Liabilities
TRAVEL & ENT EXP - Travel & Entertainment Expenses
VARIANCE EXP - Product Variance Expenses
WIP INV - WIP Inventory
```

B.32 How to Configure Group Account Numbers

Assign Financial Categories (Group Account Num) to natural accounts as follows. You need access to Fusion Applications - Application Core Setup.

1. In Fusion Applications, go to Application Core Setup.

2. Click Manage Key Flexfields.

3. Search Key Flexfield Code 'GL#'

4. Click Manage Structure Instance.

5. Find a structure instance for your chart of accounts.

6. Select the structure instance and click Edit.

7. Click Value Set Code for the Account segment to open Manage Value Sets.

8. Click Manage Values.

9. Search a natural account to which you want to assign financial categories.

10. Select a value and click Edit.

11. Assign a financial category from the list of values.

12. Save the changes.

B.33 How To Customize Extended Cross Functional Security for Accounts Payables

Note: The following section describes a post-installation and optional configuration task.

To facilitate procurement users (such as Category Managers and Procurement Managers) to perform deeper and cross functional analysis apart from their regular duty, Oracle Procurement and Spend Analytics includes configured data and functional security to access the accounts payable transactions (such as invoices, payments, payment schedules) through extended duty roles. To implement these duties, please follow the instructions in this task.

Understanding Extended Duty Roles: Seeded security roles for Oracle BI Applications for Fusion Applications include the following additional duty roles. These extended roles are not mapped to any enterprise job roles by default, but they are pre-configured within Oracle BI Applications to enforce object and data level security for Accounts Payables.

■   'Procurement Managerial Extended Analysis Duty' role (Role name: OBIA_ PROCUREMENT_MANAGERIAL_ANALYSIS_DUTY) – This BI Duty role enables users to perform cross functional analysis outside of Category Management. Internally, data security on Oracle BI Applications is implemented using 'Extended Procurement and Payable Business Unit Data Security' (Role name: OBIA_EXTENDED_PROCUREMENT_AND_PAYABLE_BUSINESS_UNIT_ DATA_SECURITY).

■   'Category Manager Extended Analysis Duty' role (Role name: OBIA_CATEGORY_ MANAGER_ANALYSIS_DUTY) – This BI Duty role enables to perform cross functional analysis outside of Procurement Management. Internally, data security on Oracle BI Applications is implemented using 'Extended Procurement and

Payable Business Unit Data Security' (Role name: OBIA_EXTENDED_
PROCUREMENT_AND_PAYABLE_BUSINESS_UNIT_DATA_SECURITY).

- ■ 'Procurement Executive Analysis Duty' role (Role name: OBIA_PROCUREMENT_
  EXECUTIVE_ANALYSIS_DUTY) to act also like a Spend Analyst/ Executive duty.
  Internally, data security on Oracle BI Applications is implemented using 'Extended
  Procurement and Spend Business Unit Data Security' (Role name: OBIA_
  EXTENDED_PROCUREMENT_AND_SPEND_BUSINESS_UNIT_DATA_
  SECURITY). This data security role enables cross functional analysis by manage
  spend Business Unit Data Security.

Follow the steps below to implement 'Procurement Executive Analysis Duty' role:

1. Assign BI duty 'Procurement Managerial Extended Analysis Duty' to Fusion
   Applications job role, 'Procurement Manager' or similar.

2. Assign BI duty 'Category Manager Extended Analysis Duty' to Fusion
   Applications job role, 'Category Manager' or similar.

3. Create 'VP of Procurement' or similar executive job role in your Fusion
   Applications deployment and assign BI duty 'Procurement Executive Analysis
   Duty' to 'VP of Procurement'.

4. Assign appropriate Fusion Applications duty roles to the job role - 'VP of
   Procurement' and assign BU privileges. Data security of 'Procurement Executive
   Analysis Duty' (OBIA duty role) is controlled by the BUs assigned to the user in
   the agent access 'manage spend' action.

5. Customize Presentation catalog permissions (for Supplier Performance – AP
   Transactions related content) and Subject Area permissions as desired for above
   mentioned roles.

For more information on how to create and manage job roles in Fusion Applications,
refer to section 'Understanding How to Secure Oracle Fusion Applications' in *Oracle
Fusion Applications Administrator's Guide*. For more information on how to define and
customize security in Oracle BI Applications, refer to *Oracle Fusion Middleware
Reference Guide for Oracle Business Intelligence Applications*.

B.34 How To Customize Extended Cross Functional Security for Employee Expenses

Note: The following section describes a post-installation and optional configuration
task.

To facilitate procurement users (such as Procurement VP or Spend Analyst) to perform
deeper and cross functional analysis apart from their regular duty, Oracle Procurement
and Spend Analytics have configured data and functional security to access the
employee expenses transactions (such as expense report, credit card transaction and
expense violation) through extended duty roles. If you would like to provision such
duty to the procurement and spend users, please follow the instructions in this task.

Understanding Extended Duty Roles: BI seeded duty roles for Fusion Applications
includes a 'Procurement Executive Analysis Duty' role (Role name: OBIA_
PROCUREMENT_EXECUTIVE_ANALYSIS_DUTY) to act like a Spend Analyst/
Executive duty. This extended role is not mapped to any enterprise job roles by
default, but it is pre-configured within Oracle BI Applications to enforce object and
data level security for Employee Expenses. Internally, data security is implemented
using 'Extended Procurement and Spend Business Unit Data Security' (Role name:
OBIA_EXTENDED_PROCUREMENT_AND_SPEND_BUSINESS_UNIT_DATA_

SECURITY). This data security role enables cross functional analysis by manage spend Business Unit Data Security.

Follow the steps below to implement 'Procurement Executive Analysis Duty' role:

1. Create 'VP of Procurement' or similar executive job role in your Fusion Applications deployment and assign BI duty 'Procurement Executive Analysis Duty' to 'VP of Procurement'.

2. Assign appropriate Fusion Applications duty roles to the job role - 'VP of Procurement' and assign BU privileges. Data security of 'Procurement Executive Analysis Duty' (OBIA duty role) is controlled by the BUs assigned to the user in the agent access 'manage spend' action.

3. Customize Presentation catalog permissions (for Employee Expense dashboard and related analyses) and Subject Area permissions as desired for 'Procurement Executive Analysis Duty' role.

For more information on how to create and manage job roles in Fusion Applications, refer to section 'Understanding How to Secure Oracle Fusion Applications' in *Oracle Fusion Applications Administrator's Guide.* For more information on how to define and customize security in Oracle BI Applications, refer to *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*.

B.35 How To Customize Security for Procurement Executive / Spend Analyst

Note: The following section describes a post-installation and optional configuration task.

To enable procurement users (such as Procurement VP or Spend Analyst) to perform deeper and cross functional analysis apart from their regular duties, Oracle Procurement and Spend Analytics includes data and functional security to access the employee expenses transactions (such as expense report, credit card transaction and expense violation) through extended duty roles. If you would like to provision such duty to the procurement and spend users, then please follow the instructions in this task.

Understanding Extended Duty Roles: BI seeded duty roles for Fusion Applications includes 'Procurement Executive Analysis Duty' role (Role name: OBIA_PROCUREMENT_EXECUTIVE_ANALYSIS_DUTY) to act also like a Spend Analyst/Executive duty. This extended role is not mapped to any enterprise job roles by default, but it is pre-configured within Oracle BI Applications to enforce object and data level security for Spend Analysis. Internally, data security on Oracle BI Applications is implemented using 'Extended Procurement and Spend Business Unit Data Security' (Role name: OBIA_EXTENDED_PROCUREMENT_AND_SPEND_BUSINESS_UNIT_DATA_SECURITY). This data security role enables cross functional analysis by manage spend Business Unit Data Security.

Follow the steps below to implement 'Procurement Executive Analysis Duty' role:

1. Create 'VP of Procurement' or similar executive job role in your Fusion Applications deployment and assign BI duty 'Procurement Executive Analysis Duty' to 'VP of Procurement'.

2. Assign appropriate Fusion Applications duty roles to the job role - 'VP of Procurement' and assign BU privileges. Data security of 'Procurement Executive Analysis Duty' (OBIA duty role) is controlled by the BUs assigned to the user in the agent access 'manage spend' action.

3. Customize Presentation catalog permissions (for Spend Analyzer dashboard and related analyses) and Subject Area permissions as desired for 'Procurement Executive Analysis Duty' role.

For more information on how to create and manage job roles in Fusion Applications, refer to section 'Understanding How to Secure Oracle Fusion Applications' in *Oracle Fusion Applications Administrator's Guidee*. For more information on how to define and customize security in Oracle BI Applications, refer to *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*.

B.36 How to Grant Cross Functional Access to Order Management Users

Note: The following section describes a post-installation and optional configuration task.

To facilitate OM users (such as Order Manager and Shipping Manager) to perform deeper and cross functional analysis apart from their regular duty, Oracle Supply Chain and Order Management Analytics has configured data and functional security to access cross functional information (such as inventory, backlog , shipping information) through extended duty roles. If you would like to provision such a duty to the Order Management users, please follow the instructions in this task.

Understanding Extended Duty Roles: Seeded security roles for Oracle BI Applications for Fusion Applications includes the following additional duty roles. These extended roles are not mapped to any enterprise job roles by default, but they are pre-configured within Oracle BI Applications to enforce object and data level security for Inventory transactions.

- 'Extended Order Management Analysis Duty' role (Role name: OBIA_ EXTENDED_ORDER_MANAGEMENT_ANALYSIS_DUTY) – This duty role provides cross-module access to the order manager job role for stand-alone Oracle BI Applications content. The cross-module access will include invoice, inventory, backlog, AR and shipping information. Data security on Oracle BI Applications is implemented using OBIA_ORDER_FULFILLMENT_ORCHESTRATION_ BUSINESS_UNIT_DATA_SECURITY.

- 'Extended Shipping Management Analysis Duty' role (Role name: OBIA_ EXTENDED_SHIPPING_MANAGEMENT_ANALYSIS_DUTY) – This duty role provides cross-module access to the shipping manager job role for stand-alone Oracle BI Applications content. The cross-module access will include inventory, backlog and orders information. Data security on Oracle BI Applications is implemented using 'OBIA_INVENTORY_ORGANIZATION_SHIPMENT_DATA_ SECURITY'.

Follow the steps below to implement Extended Duty roles in Supply Chain and Order Management Analytics

1. Assign BI duty 'OBIA_EXTENDED_ORDER_MANAGEMENT_ANALYSIS_ DUTY' to Fusion Applications job role, 'Order Manager' or similar.

2. Assign BI duty 'OBIA_EXTENDED_SHIPPING_MANAGEMENT_ANALYSIS_ DUTY' to Fusion Applications job role, 'Shipping Manager' or similar.

3. Assign appropriate Fusion Applications duty roles to the job role - 'Order Manager' and assign BU privileges. Data security of 'OBIA_ORDER_ MANAGEMENT_ANALYSIS_DUTY' (OBIA duty role) is controlled by the BUs assigned to the user in the agent access 'manage orders' action.

4. Customize Presentation catalog permissions for subject areas including cross functional content (for example Sales - Inventory and Backlog) and Subject Area permissions as desired for above mentioned roles.

For more information on how to create and manage job roles in Fusion Applications, refer to section 'Understanding How to Secure Oracle Fusion Applications' in *Oracle Fusion Applications Administrator's Guide.* For more information on how to define and customize security in Oracle BI Applications, refer to *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*.

B.37 How To Grant GL Data Role to HR VP Users

In Oracle Business Intelligence Applications, in order for a BI user with VP of HR job role to see GL data, he/she needs to be provisioned with GL data role pertaining to a Financial Analyst job role. The GL data role provisioned will control the data security that will be enforced upon the GL data the user is trying to view. To understand more details on how GL data are provisioned in Fusion Applications, please refer to *Oracle General Ledger User's Guide* for more information.

B.38 How to Implement Asset Category and Asset Location Dimension

This topic explains how to configure Fixed Asset Category and Fixed Asset Location Dimensions, and contains the following sections:

- Appendix B.38.1, "Mapping Segment Labels to BI Objects in Fusion Applications"
- Appendix B.38.2, "Configuring Fixed Asset Category and Location Dimension Using BI Extender"

> **Note:** For information about how to configure BI Extender, refer to Appendix B.27.1, "Configuring the BI Extender"

B.38.1 Mapping Segment Labels to BI Objects in Fusion Applications

In order to enable Fixed Asset Key Flexfields in Oracle BI Applications, you must perform the following steps in the Manage Key Flexfields UI in Fusion Applications. These steps enable the Flex Segments for BI and provide the mapping with BI Object names that should be used as the logical column for each of the Flexfield segments.

To map segment labels to BI Objects in Fusion Applications:

1. In the Fusion Applications seed data UI, navigate to the Manage Key Flexfields dialog.

2. Enable the BI Enabled flag for each Flexfield segment, as follows:

   a. Query for the appropriate Key Flexfield Code.

      For Fixed Asset Location, the Key Flexfield Code is LOC#.

      For Fixed Asset Category, the Key Flexfield Code is CAT#.

   b. Click the Manage Structure Instances button.

   c. Edit each of the segments and select the **BI enabled** check box, then save the details.

      The example below shows the Edit Key Flexfield Segment Instance dialog for Major Category.
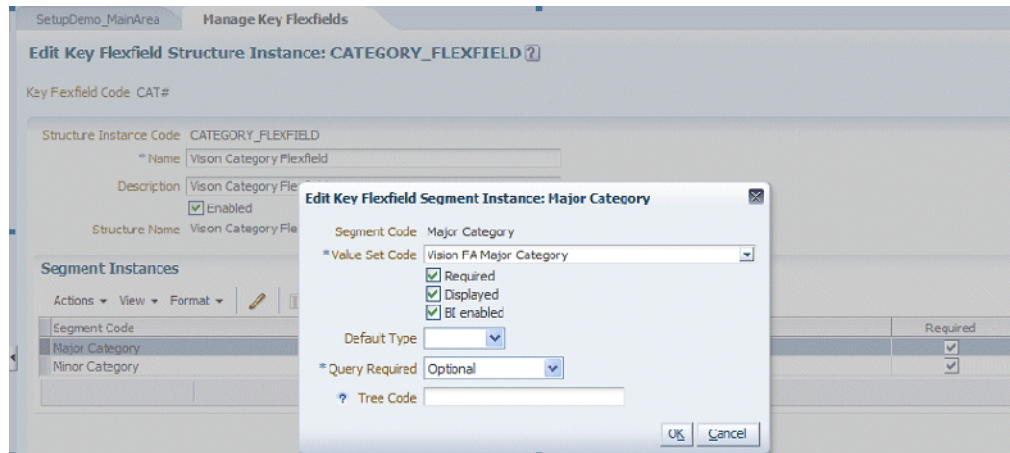
You must do this for all segments in every Structure instance that you intend to be mapped in the RPD.

3. Populate the BI Object Name for each of the Segment Labels, as follows:

Note: This name is the Logical column name in the RPD which is used to report the corresponding segment.

a. In the Manage Key Flexfields dialog, query for the appropriate Key Flexfield Code.

For Fixed Asset Location, the Key Flexfield Code is LOC#.

For Fixed Asset Category, the Key Flexfield Code is CAT#.

b. From the Actions menu, select Manage Segment Labels.
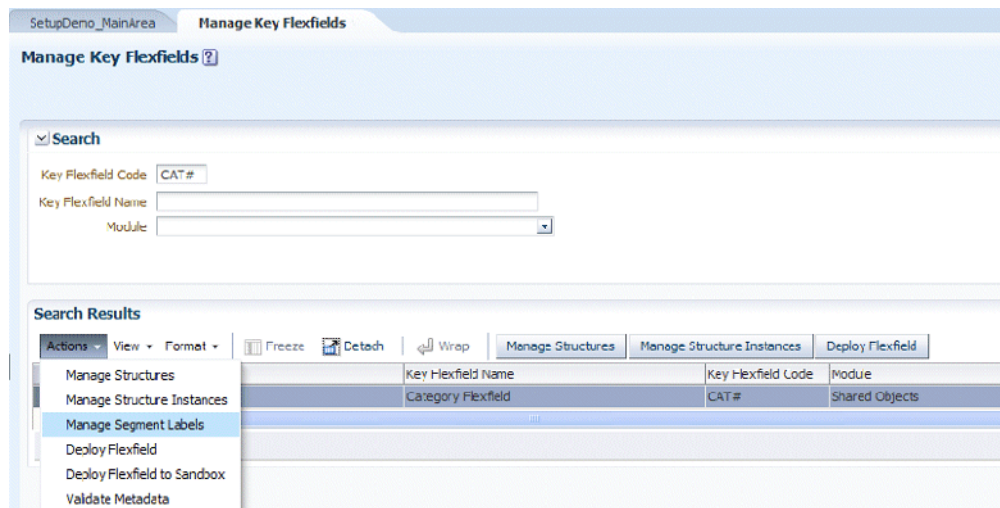
c. Use the BI Object Name field to specify a value for all the segment labels that need to be mapped in the RPD, and save the details.

The example below shows the mapping for Fixed Asset Location (LOC#).

Notes

For BASED_CATEGORY specify Major Category. For MINOR_CATEGORY specify Minor Category. For all other segment labels use Segment1 to Segment7.

4. Click the Deploy Flexfield button.

B.38.2 Configuring Fixed Asset Category and Location Dimension Using BI Extender

This section describes how to configure the Fixed Asset Category and Location dimensions in the OBIEE metadata (RPD) and how to extend the ETL (Informatica and DAC) metadata to populate the corresponding tables in the Oracle Business Analytics Warehouse.

Note: Oracle recommends that you backup all metadata repositories prior to invoking the extension process, including the OBIEE metadata (RPD file), Informatica Repository, and the DAC repository.

Prerequisite

Before you start the BI Extension process, you must enable the Extender For BIAPPS setting in Oracle BI Administration Tool. To do this, choose Tools, then Options, then General, to display the Options dialog, and select the **Extender for BIAPPS** check box.

To perform the BI Extension

1. In Oracle BI Administration Tool, open the RPD file (for example, OracleBIAnalyticsApps.rpd)

2. In the physical layer, navigate to the Oracle ADF database:

   ```
   oracle.apps.fscm.model.analytics.applicationModule.FscmTopModelAM_
   FscmTopModelAMLocal
   ```

   Right click on the connection pool under this and select Import Metadata to display the Select Metadata Objects dialog.

3. In the Select Metadata Objects dialog, do the following:

   a. Ensure that the **Automatically include any missing joined objects** option is selected.

   b. Click the **Synchronize with the data source** option on the right hand corner of the import dialog.

Selecting these options imports the necessary Flex BI Flattened View Objects and Segment columns for Fixed Asset Category and Location. The view object names in this case are FscmTopModelAM.CategoryBIAM.FLEX_BI_Category_VI and FscmTopModelAM.LocationBIAM.FLEX_BI_Location_VI.

Click on Next after you complete the import.

The example below shows the import process for FLEX_BI_Category_VI.
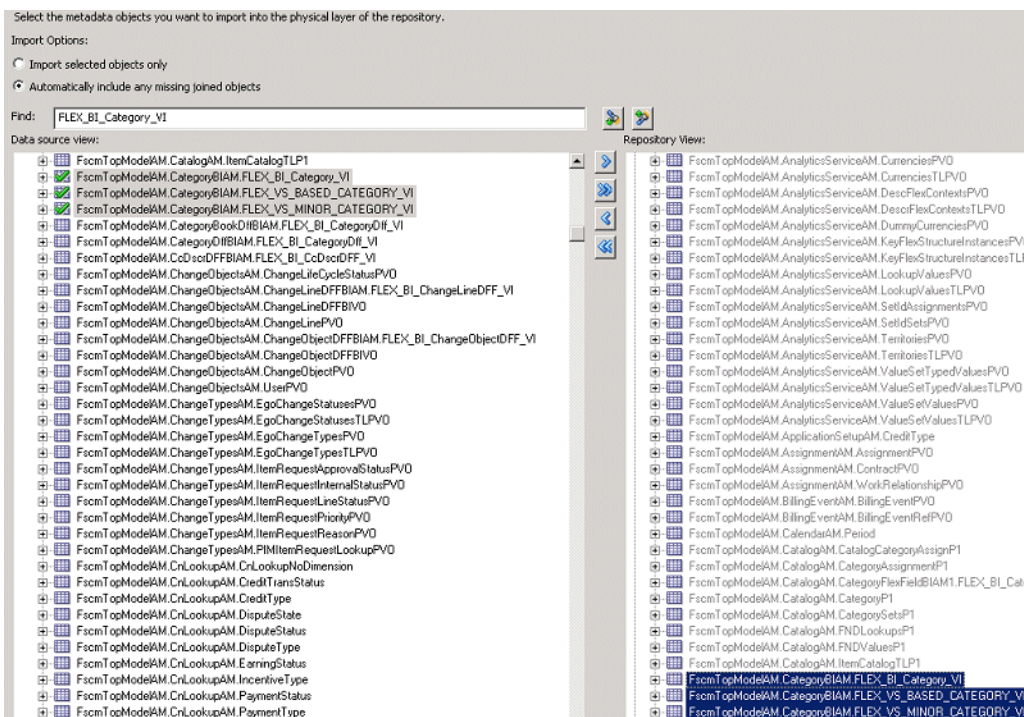
The example below shows the import process for FLEX_BI_Category_VI.
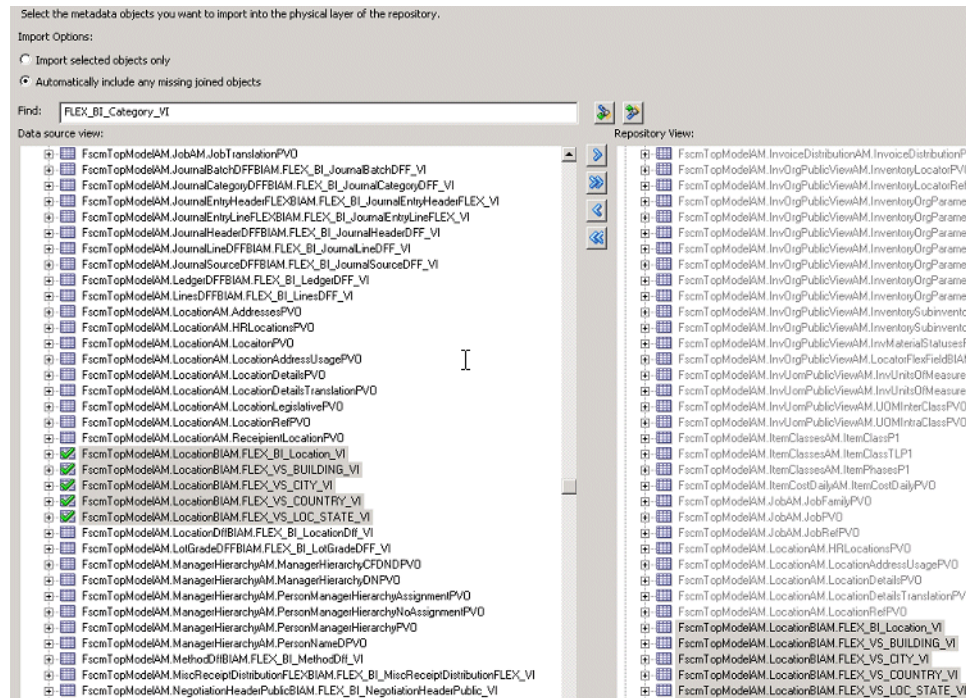
4. In the Map to Logical Model dialog, note that the VOs imported in Step 3 are automatically mapped to the appropriate logical tables. Note also that the logical columns are automatically mapped to the VO columns in the bottom panel.

Validation: You can validate at this stage that all the automatic mappings have happened as expected using the guidelines below:

- FLEX_BI_Category_VI is mapped to logical table 'Dim – Asset Category' and FLEX_BI_Location_VI is mapped to logical table 'Dim – Asset Location'.

- The segment columns in these VOs are mapped to the appropriate logical columns in these dimensions based on the Segment Label Code to BI Object Name mapping done in the Manage Key Flexfields UI in Fusion Applications as explained in Appendix B.38.1, "Mapping Segment Labels to BI Objects in Fusion Applications".

5. When you have validated your mappings, click on Next and this will take you to the 'Publish to Warehouse' dialog. Provide the necessary details, then click on Finish to complete the extension process.

Notes:

Data Warehouse

<RPD physical layer database object which is 'Oracle Data Warehouse' by default>

Informatica

User Name – <Informatica Repository User Name>

Password - <Informatica Repository Password>

Database

User Name – <Informatica Repository DB User Name>

Password - <Informatica Repository DB Password>

DAC

User Name – <DAC Repository DB User Name>

Password - <DAC Repository DB Password>

6. Validate and save the changes.

7. If you have successfully completed the extension process, then you will see the Informatica mapping SDE_FUSION_FixedAssetCategoryDimension and SDE_FUSION_FixedAssetLocationDimension in your repository extended to populate these new columns that you have just mapped.

B.39 How to Perform System Setups and Post Install Tasks for Oracle BI Applications

Make sure that you have followed the setup steps in Section 3.1.1, 'High-level Tasks for Setting Up Oracle BI Applications' of *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications*.

B.40 How to Reload the Time Dimension Tables After the Data Warehouse Is Loaded

Steps for 11g Fusion Applications:

1. You will see a Subject Area named Common Dimension and Extend Day Dimension. If you have Multiple calendar in your Day Dimension, then choose the configuration tag Extend Day Dimension Multiple Calendar Support, or else remove it. Then assemble the Subject Area.

2. Choose the Task Sil_DayDimension_XTND. Choose a new START_DATE (= @ END_DATE +1) and new END_DATE and set the parameter values.

3. Choose the Task SDE_FUSION_TimePeriodMCalPeriod_XTND. Retain the START_DATE and choose a new END_DATE.

4. Build the corresponding Execution Plan with same name.

5. Remember to change FILE_MCAL_CAL_D, FILE_MCAL_CONTEXT_G, FILE_MCAL_PERIOD_DS (these 3 in universal) and FILE_MCAL_CONFIG_G, in case you use them as source.

Execution Plan steps for an Fusion Applications container:

1. You will see a subject Area 'Common-Extend Day Dimension'. If you have Multiple calendars in your Day Dimension, then choose the configuration tag Extend Day Dimension Multiple Calendar Support, or else remove it. Then, assemble the Subject Area.

2. Choose the Task Sil_DayDimension_XTND. Choose a new START_DATE (= END_DATE +1) and new END_DATE and set the parameter values.

3. Choose the Task SDE_FUSION_TimePeriodMCalPeriod_XTND. Retain the START_DATE and choose a new END_DATE.

4. Build the corresponding Execution Plan named 'Common-Extend Day Dimension Fusion'.

B.41 How To Configure Scorecard Target Before Running ETL

Open <INFASRC>\file_purch_scorecard_target.csv to specify the target for the KPI. The supported dimensions are time dimension and procurement Business Unit dimension. You must specify the following values in the source file in the required data format:

- Quarter start date
- Procurement business Unit ID
- KPI name
- KPI target value

The following KPIs are supported for KPI target value:

- # of Negotiation Lines Awarded Per Category
- # of POs Per Buyer
- # of Suppliers Per Category
- % of Fulfilled Requisition Lines Past Expected Date
- % of Late Receipts
- % of Processed Requisition Lines Past Expected Date
- % of Realized Savings
- % of Supplier Diversity Spend
- % of Unfulfilled Requisition Lines Past Expected Date
- Average Negotiation Cycle Time
- Average Requisition to Receipt Cycle Time
- Electronic Invoice %
- Manual Requisition Lines Rate
- Non-Agreement Purchase Rate
- Overall Accepted %
- Perfect Invoices %
- Purchase Order Schedule Line Return Rate
- Received On Time %

B.42 How to Configure Order Item and Service Request Flat Files For ETL

Background

In Fusion Applications, there are several entities that are sourced from non-Fusion Applications systems. Fusion Applications CRM is leveraging OBIA (Oracle Business Intelligence Applications) to integrate data from Fusion Applications and non-Fusion Applications source systems. The Oracle BI Applications metadata layer consolidates disparate physical data sources and makes it ready for analysis by Fusion Applications users. Sales Prospector (SPE) is a brand new Fusion application for sales users helping them to manage their pipeline and whitespace effectively. SPE expects Order Item and Service Request data to be supplied from non-Fusion applications.

ETL from Flat Files

Non-Fusion Applications data such as Order Item and Service Request can be directly loaded into Oracle Business Analytics Warehouse as long as the data can be presented in the specified flat file format. The ETL process loads the non-Fusion Applications data from the flat files and Fusion Applications data from Fusion Applications database tables into staging tables; then loads data from the staging tables into Oracle Business Analytics Warehouse.

SPE ETL Preparation

SPE needs non-Fusion Applications data for Order Item Fact, Service Request Fact and Service Request Dimension. The data should be presented in flat files according to the following specifications:

- Data should be in CSV files (*.csv).

- For full ETL, the files should contain all initial records that are supposed to be loaded into Oracle Business Analytics Warehouse; for incremental ETL, the files should contain only new or updated records.

- The files are specially formatted for Fusion Sales Prediction Engine (SPE) data mining use only. All columns in the files should follow Fusion application data model terms and standards, and all ID columns in the files are expected to have corresponding Fusion Integration ID.

- Data should start from line six of each file. The first five lines of each file will be skipped during ETL process.

- Each row represents one record in staging table.

- All date values should be in the format of YYYYMMDDHH24MISS. For example, 20071231140300 should be used for December 31, 2007, 2:03 pm.

- Columns DATASOURCE_NUM_ID and INTEGRATION_ID in all flat files cannot be NULL.

- Column DATASOURCE_NUM_ID needs to be fixed to 200, which is also the Fusion Applications data source number.

The Flat files for Order Item Fact, Service Request Fact and Service Request Dimension are:

- file_orderitem_fs.csv - for more information about the structure of this file, see Section B.42.1, "Flat file file_orderitem_fs.csv".

- file_srvreq_fs.csv - for more information about the structure of this file, see Section B.42.2, "Flat file file_srvreq_fs.csv".

- file_srvreq_ds.csv - for more information about the structure of this file, see Section B.42.3, "Flat file file_srvreq_ds.csv".

Before starting the ETL run, the flat files should be prepared based on the formats provided in sections below and then copied to the SrcFiles directory on the Informatica server machine (for example, C:\Program Files\Informatica\PowerCenter8.6.x\server\infa_shared\SrcFiles).

B.42.1 Flat file file_orderitem_fs.csv

The file is generic and therefore does not support any source order system specific features, such as recurring order lines and etc. Each line in this file will contribute to the total order amount. The granularity of this file is each order line.

The file is specially formatted for Fusion Sales Prediction Engine (SPE) data mining use only.

Table B–7   File Structure for file_orderitem_fs.csv

Column Name	Data Type	Sample Data	Description
CUSTOMER_ID	VARCHAR(80)	999997551042159	Customer Party Id. There could be more than one customer IDs in an order. Among the possible customer IDs of bill to, ship to, invoice to and so on; this is the primary ID for BI analysis use.
			Foreign key to HZ_PARTIES.PARTY_ID.
CURCY_CD	VARCHAR(20)	USD	Currency Code, the currency that the order line amounts are based on.
CRM_CURR_EXCHANGE_RATE	NUMBER(28,10)	1.00	CRM Currency Exchange Rate, which is for the conversion of the order line amounts to the CRM common currency.
CRM_CORP_CURR_CODE	VARCHAR(20)	USD	CRM Common Currency Code.
ORDER_ID	VARCHAR(80)	4171787	Order header ID.
PROD_ID	VARCHAR(80)	999997500678718	Product Inventory Item ID.
			Foreign key to EGP_SYSTEM_ITEMS_B.INVENTORY_ITEM_ID.
PROD_GROUP_ID	VARCHAR(80)	Null	Product Group ID. Optional for SPE ETL use. Leave null.
RESOURCE_ID	VARCHAR(80)	123445623	Resource ID, order owner Resource ID for order.
			Foreign key to HZ_PARTIES.PARTY_ID

Table B–7   (Cont.) File Structure for file_orderitem_fs.csv

Column Name	Data Type	Sample Data	Description
RESOURCE_ORG_ID	VARCHAR(80)	3453453453	Resource Organization ID, order owner's organization ID.
			Foreign key to HR_ALL_ ORGANIZATION_ UNITS_ F.ORGANIZATION_ID.
SOURCE_ID	VARCHAR(80)	100000016742344	Marketing campaign source code defined in MKT_SC_SOURCE_ CODES.
ORDER_DT	DATE	20061220000000	Order Date in the format of YYYYMMDDHH24MISS. It is the date when order is placed. This date is used in ETL as canonical date for resolving dimensional FKs.
DATASOURCE_NUM_ ID	NUMBER(10)	200	Data Source Number ID. Need to be fixed to 200, which is the same value for Fusion Applications data source in ETL.
INTEGRATION_ID	VARCHAR(80)	12149813	Integration ID, the order Line ID. Typically, each order may have one order header and multiple order lines.
DISCNT_AMT	NUMBER(28,10)	2.33	Discount Amount, deduction made to the unit price.
NET_PRI	NUMBER(28,10)	45.752	Net Price of order line item. This is the final price after deducting discount amount.
QTY_REQ	NUMBER(28,10)	12	Quantity Ordered for the line item.
PR_TERR_ID	VARCHAR(80)	1000000000023112	Primary Territory ID, ID of primary sales territory where order is placed.
			Territory ID is defined in MOT_TERRITORIES.
CREATED_BY_ID	VARCHAR(80)	SALES_ADMIN	Created By ID, Login ID of the user who created the row.
CREATED_ON_DT	DATE	20071231140300	Created On Date in the format of YYYYMMDDHH24MISS.
CHANGED_BY_ID	VARCHAR(80)	SALES_ADMIN	Changed By ID, Login ID of the user who modified the row.

Table B–7   (Cont.) File Structure for file_orderitem_fs.csv

Column Name	Data Type	Sample Data	Description
CHANGED_ON_DT	DATE	20071231140300	Changed On Date in the format of YYYYMMDDHH24MISS.
DELETE_FLG	VARCHAR(1)	Null, Y or N	Delete Flag, indicates if the record is deleted since last ETL. Default to N if null.
X_CUSTOM	VARCHAR(10)	Null	ETL reserved. Leave null.

B.42.2 Flat file file_srvreq_fs.csv

The columns listed below are required for SPE ETL use. The granularity of this file is each Service Request. The file is specially formatted for Fusion Sales Prediction Engine (SPE) data mining use only.

Table B–8   File Structure for Flat file file_srvreq_fs.csv

Column Name	Data Type	Sample Data	Description
DATASOURCE_NUM_ID	NUMBER(10)	200	Data Source Number ID. Data Source Number ID needs to be fixed to 200, the same value for Fusion Applications data source in ETL.
INTEGRATION_ID	VARCHAR(80)	12149813	Integration ID, unique IDentifier ID for each Service Request.
CLOSE_DT	DATE	20030616174947	Closed Date, date in the format of YYYYMMDDHH24MISS when service request was closed.
OPEN_DT	DATE	20020516174947	Open Date, date in the format of YYYYMMDDHH24MISS when service request was open.
DELETE_FLG	VARCHAR(1)	Null, Y or N	Delete Flag, indicates if the record is deleted since last ETL. Default to N if null.
CREATED_BY_ID	VARCHAR(80)	SALES_ADMIN	Created By ID, Login ID of user who created the row.
CREATED_ON_DT	DATE	20071231140300	Created On Date in the format of YYYYMMDDHH24MISS.
CHANGED_BY_ID	VARCHAR(80)	SALES_ADMIN	Changed By ID, Login ID of the user who modified the row.
CHANGED_ON_DT	DATE	20071231140300	Changed On Date in the format of YYYYMMDDHH24MISS.

Table B–8   (Cont.) File Structure for Flat file file_srvreq_fs.csv

Column Name	Data Type	Sample Data	Description
X_CUSTOM	VARCHAR(10)	Null	ETL reserved. Leave null.
CUSTOMER_ID	VARCHAR(80)	999997551042159	Customer Party Id.
			Foreign key to HZ_ PARTIES.PARTY_ID.
PROD_ID	VARCHAR(80)	999997500678718	Product Inventory Item ID.
			Foreign key to EGP_ SYSTEM_ITEMS_ B.INVENTORY_ITEM_ ID.

B.42.3 Flat file file_srvreq_ds.csv

The columns listed below are required for SPE ETL use. The granularity of this file is each Service Request. The file is specially formatted for Fusion Sales Prediction Engine (SPE) data mining use only.

Table B–9   File Structure for Flat file file_srvreq_ds.csv

Column Name	Data Type	Sample Data	Description
DATASOURCE_NUM_ ID	NUMBER(10)	200	Data Source Number Id. Data Source Number Id needs to be fixed to 200, the same value for Fusion Applications data source in ETL.
INTEGRATION_ID	VARCHAR(80)	1-10E-5	Integration ID, unique Identifier ID for each Service Request.
CLOSE_DT	DATE	20020516174947	Closed Date, date in the format of YYYYMMDDHH24MISS when service request was closed.
OPEN_DT	DATE	20020516174947	Open Date, date in the format of YYYYMMDDHH24MISS when service request was open.
SEV_CD	VARCHAR(80)	SR_ SEVERITY~3-Medium	Severity Code of the Service Request.
			Possible values are: SR_ SEVERITY~1-Critical,
			SR_SEVERITY~2-High,
			SR_ SEVERITY~3-Medium,
			SR_SEVERITY~4-Low.

Table B–9 (Cont.) File Structure for Flat file file_srvreq_ds.csv

Column Name	Data Type	Sample Data	Description
STATUS	VARCHAR(80)	SR_STATUS~Open	Service Request Status. Possible values are: SR_STATUS~Approved, SR_STATUS~Cancelled, SR_STATUS~Closed, SR_STATUS~Completed, SR_STATUS~Open, SR_STATUS~Pending.
DELETE_FLG	VARCHAR(1)	Null, Y or N	Delete Flag, indicates if the record is deleted since last ETL. Default to N if null.
CREATED_BY_ID	VARCHAR(80)	SALES_ADMIN	Created By ID, Login ID of the user who created the row.
CREATED_ON_DT	DATE	20071231140300	Created On Date in the format of YYYYMMDDHH24MISS.
CHANGED_BY_ID	VARCHAR(80)	SALES_ADMIN	Changed By ID, Login ID of the user who modified the row.
CHANGED_ON_DT	DATE	20071231140300	Changed On Date in the format of YYYYMMDDHH24MISS.
X_CUSTOM	VARCHAR(10)	Null	ETL reserved. Leave null.

B.43 How to Add Balances to BI Balance Group

In order to extract payroll balances into Oracle Business Analytics Warehouse, the balances must be assigned to the BI Balance Group in the Fusion Applications system. By limiting the balances extracted, the performance of ETL and analyses will be improved. In addition, only certain types of balance are suitable for including in the warehouse. You should only extract run balances, as other types of balances may not be fully additive (for example year-to-date balances cannot be added together).

Pre-requisites

- Access to Fusion Applications Payroll Administration area.

- Office 2007 with Oracle ADF 11g Plug In.

- List of defined balances required to add to BI Balance Group:

    - Listed by Balance Dimension (which must be Run) and Balance Type.

    - Listed by Legislative Data Group.

Instructions

These instructions cover the steps required to add balances to the BI Balance Group for inclusion in Oracle Business Analytics Warehouse. There will be more details on the Payroll Administration documentation which will cover exceptions and verification reports to validate any setup.

Steps to create a batch:

1. Log into Fusion Applications and navigate to the Payroll Administration area (Navigator => Payroll => Payroll Administration).

2. In the Task pane select Batch Processing => Batch Loader.

3. Click the Download button to open the Batch Loader Spreadsheet, re-entering your login details as requested.

4. In the Batch Header Sheet tab, enter a name for the batch and the Legislative Data Group and Save.

5. Double-click the batch name to select the batch and open the Batch Content Sheet tab.

6. Click the Add button and select the 'Add a Defined Balance' action.

7. Enter the details for each defined balance to be added to the BI Balance Group:

   ■ Line Sequence.

   ■ Attribute Definition – 'Global BI Attribute'.

   ■ Legislative Data Group – as entered in step 4.

   ■ Balance Dimension – balance dimension name; this should be a simple run balance without any contexts.

   ■ Balance Type – balance type name for the defined balance

8. Save the details.

Steps to transfer the batch:

1. Back in Fusion Applications navigate to the Checklists page (Navigator => Payroll => Checklists).

2. In the Task pane select Payroll Flows => Submit a Process or Report.

3. Select the Legislative Data Group for the batch.

4. Select the 'Transfer Batch' process and click Next.

5. Enter the details:

   ■ Give a name for the Payroll Flow.

   ■ For the batch parameter, select the batch name entered in step 4 of 'Steps to create a batch'.

6. Submit the process

B.44 Getting Started With Functional Configuration

To get started with Functional Configuration, see Section 5.2, "Roadmap for Functional Configuration".

A BI Application Offering and one or more Functional Areas are selected during the creation of an Implementation Project. A list of Functional Setup tasks is generated based on the selected Oracle BI Applications Offering and Functional Area(s).

There are four types of Functional Task:

■ Tasks to configure Data Load Parameters - Clicking on the Go To Task button for these tasks launches Oracle BI Applications Configuration Manager and the Manage Data Load Parameter setup user interface is displayed with the appropriate set of Data Load Parameters required to perform a task.

- Tasks to manage Domains and Mappings - Clicking on the Go To Task button for these tasks launches Oracle BI Applications Configuration Manager and the Manage Domains and Mappings setup user interface is displayed with the appropriate set of Domain Mappings.

- Tasks to configure Reporting Parameters - Clicking on the Go To Task button for these tasks launches Oracle BI Applications Configuration Manager and the Manage Reporting Parameter setup user interface is displayed with the appropriate set of Reporting Parameters required to perform a task.

- Tasks which are informational - These tasks provide instructions for configurations which are to be performed outside of Oracle BI Applications Configuration Manager either in Informatica PowerCenter, DAC, or the RPD using the BI Administration Tool. Some Informational Tasks provide conceptual, background or supporting information and do not describe a functional configuration.

B.45 Overview of Oracle Sales Analytics

Oracle Sales Analytics helps sales executives, sales managers and front line sales representatives obtain insight into sales performance at all stages of the sales cycle. Coverage includes all key areas in Sales, namely, Opportunity and Pipeline management, Win Loss Analysis, Quota Management, Forecasting and Customer Analytics.

B.46 Overview of Oracle Marketing Analytics

Oracle Marketing Analytics is a comprehensive analytical solution that provides timely fact-based insight into the marketing activities of the entire organization. It provides new levels of information richness, usability, and reach to marketing professionals throughout the enterprise. It provides actionable intelligence in the following Marketing areas: Marketing Effectiveness, Customer Insight, and Lead Analysis.

The main functional areas within Marketing Analytics are:

- Core Marketing - Helps to analyze customer and prospect responses to campaigns, marketing activities and marketing offers.

- Marketing Leads - Helps to do a detailed analysis on leads as they move through the lifecycle. Analysis includes lead to opportunity conversion, what percentage of the leads are getting rejected and retired by the sales team, what are the main reasons, how effective the sales force in converting the leads and so on.

- Customer Interactions Management - Helps to analyze various marketing and sales interactions a company has had with their customers and prospects.

- Opportunity Revenue Management - Helps to analyze the opportunity revenue generated from marketing activities, helping marketers to calculate the Return on Marketing Investment (ROMI).

Please note that you must implement the above 4 functional areas for a complete end to end analysis of marketing campaigns and other activities.

Opportunity Landscape - Opportunity Landscape is a functional area included within Marketing Analytics, but is not necessary for the Marketing Analytics to function. This module provides analysis for Fusion Opportunity Landscape application. Please refer to the product documentation of Fusion Opportunity Landscape for more details.

B.47 Overview of Oracle Partner Analytics

Partner Analytics helps channel and partner account managers assess partner and program performance on all key fronts - lead generation, deals registered, revenue and enrollments. It also enables partner organization sales representatives and managers to assess their own sales performance.

B.48 Overview of Oracle Human Resources Analytics

Oracle Human Resources contains information for workforce effectiveness, workforce deployment, headcount gain and losses, compensation, payroll, absence and leave accrual. The Oracle HR Analytics application has the following functional areas:

Workforce Effectiveness Functional Area

The Workforce Effectiveness functional area allows senior HR executives to monitor key HR effectiveness metrics at the enterprise level. It combines key HR metrics with the organization's financial data to analyze industry benchmark metrics such as revenue per employee, contribution per employee, and so on. The correlation of workforce and financial metrics provides insight into how workforce trends directly impact the organization's operations and financial strength.

Sample questions answered by the Workforce Effectiveness area are:

- What is the trend in revenue per employee?

- What is the gross margin contribution per headcount?

- What is the ratio of workforce cost to gross margin contribution per headcount?

- What are the trends in workforce costs, e.g. employee benefit expenses, payroll expenses, contracting expenses, and so on.?

Workforce Deployment Functional Area

The Workforce Deployment functional area is the foundation for workforce analysis including three subject areas, Workforce Deployment, Headcount Gain and Loss, and Compensation.

- **Workforce Deployment Subject Area**

  Workforce Deployment provides the comprehensive data analysis on head count, retention, workforce diversity, employee performance, and contingent labor utilization. Key workforce profile information such as employee, organization, supervisor, performance band, and service band are shared with other HR functional areas. Sensitive personal attributes like birth date, age, and marital status are organized in a separate folder to allow for restricted access. Configured HR event analysis is another key feature of Workforce Deployment functional area. Customers can configure various employee assignment actions to support analysis in voluntary/involuntary termination, hires, transfers, promotions, or layoffs, and so on. In addition, changes in an employee's job, organization, location, supervisor and salary are tracked to support workforce movement analysis.

  Specifically, the following analysis is supported by Workforce Profile functional area:

  – Headcount analysis

  – Workforce diversity

  – Employee attrition and retention

– Employee performance

– Span of control

– Internal mobility

Sample questions answered by the Workforce Deployment area are:

– What is the current staffing level? What is the headcount growth rate?

– What is the contingent labor usage trend by job, by location, by department?

– Is employee turnover within the target? How is employee turnover trending?

– Is there a retention hotspot?

– What is the workforce demographics by ethnicity, by age group, by education, and so on?

– Who are the top performers and under performers?

– Is there a shortage of workers by job by geography?

- **Workforce Gain and Loss Subject Area**

An internal transfer can result in a headcount gain or loss or no change effect at different levels of an organization. The Workforce Gain and Loss area is designed to track headcount and FTE change at all levels of the supervisor hierarchy. At any supervisor level, it can easily identify net headcount changes and analyze inter- and intra-unit headcount gain and loss resulting from external or internal movements such as hires, transfer-ins/outs, reorganizations and terminations.

Sample questions answered by the Headcount Gain and Loss area are:

– How is current headcount changed from last year?

– What is the headcount gain from new hire?

– What is the headcount loss from termination?

– What is the net headcount change due to reorganization?

– What is the internal headcount movement due to inter-unit transfers?

- **Compensation Subject Area**

The delivered compensation metrics allow Compensation analysts and managers to analyze employee salary and compensation-ratio analysis at different levels of granularity. It proactively detects over or under-compensated employees, which can have big impact on your company's ability to maintain a competitive edge. It provides the capability to correlate employee pay with performance and perform pay equity and salary compression analysis by various factors e.g. job, grade and length of service.

Sample questions answered by the Compensation area are:

– What it the average salary based on length of service?

– How do the average salaries and compa-ratios compare for each job/job family?

– How many workers are outside their salary range?

– What is the salary compression between jobs, between grades and between new hire and seasoned workers?

Payroll Functional Area

HR Payroll functional area captures employee pay period balances as well as monthly, quarterly and annual payroll balances. It allows you to analyze earnings, deductions, employee and employer-paid benefits and taxes that may comprise an employee's total compensation. It helps HR and line managers to assess employees' total compensation and the relationship between base salary and variable pay components. It analyzes total payroll spend and monitors cost differences by geography. It also helps line managers and finance to monitor overtime usage and overtime spend and keep employees' payroll costs within budget.

Sample questions answered by the Payroll area are:

- How has the total cost of compensation changed over time?

- How does the total cost of compensation this month compare to the same month 1, 2, 3, and 4 years ago?

- How has the proportion of base pay versus bonus versus other types of compensation changed over the years?

- What is the trend in overtime hours and cost? Where is the biggest overtime spending?

- What is the total payroll cost trend by country?

Absence and Leave Accrual Functional Area

Absence and Leave Accrual includes two components, absence tracking and accrual balance tracking. Unplanned absenteeism impedes workforce productivity and increase workforce cost. Absence and Leave Accrual functional area analyzes employee leave accrual balances and absence trends, working days lost, and hot spots in absenteeism.

Sample questions answered by the Absence and Leave Accrual area are:

- What is the trend in employee absences by absence type?

- What is the lost productivity due to employee absences?

- Which employees have a high Bradford score (a measure to track habitual sick leave)?

- What is the current leave accrual liability cost?

- Which departments and/or locations have the highest leave accrual balances?

- What is the percentage of employees who have reached their maximum accrual balance?

B.49 Overview of Oracle Customer Data Management Analytics

Fusion Customer Data Management Analytics provides insight into the data quality of an organization's customer data.This solution provides a set of data completeness analyses which allow you to monitor, measure, and analyze the completeness of the underlying party information of your enterprise, including organization and person information.

B.50 Overview of Oracle Financial Analytics

Oracle Financial Analytics comprises the following Functional Areas:

- **Employee Expenses** - The Oracle Employee Expenses Analytics application has been designed to provide visibility into an organization's employee related expenditures, including corporate card usage, expense policy violations, and the overall submission and approval process. Gain control of the drivers of employee expenses by isolating top spenders across expense categories and identifying recurring policy violations. Visibility into overall expense trends improve ability to negotiate with key merchants. The default configuration for the Oracle Employee Expenses Analytics application is based on what is identified as the most-common level of detail or granularity. However, you may configure and modify the extracts to best meet your business requirements.

- **Fixed Assets** - The Oracle Fixed Assets Analytics application provides finance controllers, asset managers, and cost center managers with a complete picture of the asset's life cycle from acquisition through to retirement. Fixed assets comprise approximately 40 to 50% of the balance sheet and are a key component for both the commercial and public sector customers. Tracking asset life cycle value and measuring returns on some of the key assets are important to increase the overall return of the organization. The default configuration for the Oracle Fixed Assets Analytics application is based on what is identified as the most-common level of detail or granularity. However, you may configure and modify the extracts to best meet your business requirements.

- **General Ledger** - The General Ledger Analytics application has been designed to provide insight into key financial areas of performance, including balance sheet, cash flow, expenses, budget vs. actual, working capital, liquidity. Identify root cause of discrepancies for more timely, informed decisions at all levels of the organization. Gain access to reporting and analysis from intra-period financial information before books are closed. The default configuration for the Oracle General Ledger Analytics application is based on what is identified as the most-common level of detail or granularity. However, you may configure and modify the extracts to best meet your business requirements.

- **Payables** - The Oracle Payables Analytics application has been designed to provide an overview of the health of the payables side of the business and enables Finance to best manage its cash outflows and ensure timely payments to its suppliers. The need for analysis is increasingly important because suppliers are becoming strategic business partners with the focus on increased efficiency for just in time, and quality purchasing relationships. The default configuration for the Oracle Payables Analytics application is based on what is identified as the most-common level of detail, or granularity. However, you can configure or modify the extracts to best meet your business requirements.

- **Profitability** - The Oracle Profitability Analytics application has been designed to provide key data pertaining to profitability, including Profit and Loss Statements, Customer and Product profitability, Margin Analysis, ROA, and ROE. Insight into Revenue and Cost Drivers help drive financial accountability, and proactive behavior. The default configuration for the Oracle Profitability Analytics application is based on what is identified as the most-common level of detail or granularity. However, you may configure and modify the extracts to best meet your business requirements.

- **Receivables** - The Receivables Analytics application has been designed to provide key data pertaining to receivables, including receivables due, credit risk, payments, collector efficiency and enables Finance to best manage cash inflows and their ability to collect debt. Each day that your receivables are past the due date represents a significant, opportunity-cost to your company. Keeping a close eye on the trends, and clearing of AR is one way to assess the efficiency of your sales operations, the quality of your receivables, and the value of key customers.

The default configuration for the Oracle Receivables Analytics application is based on what is identified as the most-common level of detail or granularity. However, you may configure and modify the extracts to best meet your business requirements.

B.51 Overview of Oracle Product Information Management Analytics

Oracle Product Information Management (PIM) Data Hub is an enterprise data management solution that enables customers to centralize all product information from heterogeneous systems. It allows organizations to create a single, enterprise view of their product information, by integrating, standardizing and synchronizing fragmented product data from multiple source systems into a central, operational, data repository ('Hub').

PIM Data Hub solution centralizes the disparate sources of product information and provides a full, 360-degree view of products across all channels. It enables articulated management and communication of product information both within the organization as well as externally to customers and value-chain partners.

Oracle Product Information Management Analytics application comprises the following Subject Areas:

- PIM - Item: This subject area provides information on creation and approval activities related to items of different Item class, type, phase and status.

- PIM - Change Orders: This subject area provides information on activities related to Change Orders such as number of change orders in different age range, average age of change orders, different stages of change order life cycle, e.g. approved, rejected, draft, pending effective.

- PIM - New Item Request: This subject area provides information on activities related to New Item requests such as number of new item requests in different age range, average age of new item requests, New Item Request Cycle Time and different stages of new item request life cycle, e.g. new, approved, rejected.

- PIM - Item Catalog : This subject area provides information on activities related to Item Catalogs like number of new catalogs, categories, and shared categories.

- PIM - Item Batch : This subject area provides information on activities related to Item Import from any external system such as number of items excluded, partially imported, successfully imported, and so on during the batch import process.

B.52 Overview of Oracle Project Analytics

Project Analytics includes the following Subject Areas:

- **Commitments**: A detailed subject area that provides the ability to report on the obligations for future expenditures that a project has made. Reporting can be done across organizations, projects, tasks, resources and periods. There are metrics showing raw and burdened amounts for requisitions, purchase orders and supplier invoices.

- **Cross Charges**: A detailed subject area that provides the ability to report on expenditures that projects or organizations charge to each other for resources that they share. Reporting is possible across periods, organizations, projects, task and resources. Metrics include charges generated by the Intercompany Billing or the Borrowing and Lent methods. There are metrics for current and previous periods.

Note: For reporting on Cross Fact Analysis, Oracle recommends that you always have a filter on Canonical BU (Business Unit Name column under Organizations folder in presentation area).

Table B–10   List of Facts and Business Units/Organization

Fact	Canonical BU (Business Unit)/Organization
Project Billing	Contract BU / Organization
Project Budget	Project BU / Organization
Project Budget - Linear Spread	Project BU / Organization
Project Commitment	Project BU / Organization
Project Commitment Snapshot	Project BU / Organization
Project Contract	Contract BU / Organization
Project Cost	Expenditure BU / Organization
Project Cross Charge - Invoice	Project BU / Organization
Project Cross Charge - Provider	Expenditure BU / Organization
Project Cross Charge - Receiver	Project BU / Organization
Project Cross Charge - Revenue	Contract BU / Organization
Project Forecast	Project BU / Organization
Project Funding	Contract BU / Organization
Project Revenue	Contract BU / Organization

Notes

The Canonical BU (Canonical Organization) is the Common Logical BU (Organization) against which data is analyzed across different fact tables. From each fact table one main BU (Org) is selected to be used for analyzing data in that fact table (for example, for Cost Fact, the canonical BU is the Expenditure BU; for Revenue Fact, the canonical BU is the Contact BU) and use the corresponding foreign key to join to the logical dimension Dim - Business Unit (Dim - Project Organization). These dimensions Dim - Business Unit and Dim - Project Organization are called Canonical BU and Canonical Project Organizations dimensions respectively. For example, for the Cost Fact the join would be

```
Dim_W_INT_ORG_D_Business_Unit.SCD1_WID = Fact_W_PROJ_COST_LINE_
F_Project_Cost.EXPENDITURE_OPER_UNIT_WID
```

For Revenue Fact the join would be:

```
Dim_W_INT_ORG_D_Business_Unit.SCD1_WID =Fact_W_PROJ_REVENUE_
LINE_F_Revenue_Lines.CONTRACT_BU_WID
```

In addition the Canonical BU calendar is used when forming the foreign key to the Fiscal Calendar Day dimension (W_MCAL_DAY_D).For Cross Fact Analysis, you must always ensure that you have a filter on Canonical BU (Business Unit Name column under Organizations folder in presentation area). This filter on Canonical BU is required in all dashboards because it ensures the calendar is unique and prevents double counting.

The table below lists the Canonical BU (Canonical Organizations) that are available for the Logical Facts supported in Project Analytics solution.

Table B–11   List of Facts, Canonical BUs, and Canonical Organization

Fact	Canonical BU (Business Unit)/Organization	Canonical Organization
Project Billing	Contract BU	Contract Organization
Project Budget	Project BU	Project Organization
Project Budget - Linear Spread	Project BU	Project Organization
Project Commitment	Project BU	Project Organization
Project Commitment Snapshot	Project BU	Project Organization
Project Contract	Contract BU	Contract Organization
Project Cost	Expenditure BU	Expenditure Organization
Project Cross Charge - Invoice	Project BU	Project Organization
Project Cross Charge - Provider	Expenditure BU	Expenditure Organization
Project Cross Charge - Receiver	Project BU	Contract Organization
Project Cross Charge - Revenue	Contract BU	Contract Organization
Project Forecast	Project BU	Project Organization
Project Funding	Contract BU	Contract Organization
Project Revenue	Contract BU	Contract Organization

B.53 Overview of Oracle Procurement and Spend Analytics

Oracle Procurement and Spend Analytics comprises of Procurement Analytics, Sourcing Analytics, and Employee Expense Analytics.

Oracle Procurement and Spend Analytics enable organizations to optimize their supply side performance by integrating data from across the enterprise value chain and enabling executives, managers, and frontline employees to make more informed and actionable decisions. Organizations using Oracle Procurement and Spend Analytics benefit from increased visibility into the Corporate Spend and complete source-to-pay process, including comprehensive sourcing and procurement analysis, supplier performance analysis, supplier payables analysis, and Employee Expenses analysis. Through complete end-to-end insight into the savings, spend patterns, and supplier performance, organizations can significantly reduce costs, enhance profitability, increase customer satisfaction, and gain competitive advantage. Oracle Procurement and Spend Analytics also integrates with the other applications in the Oracle Business Intelligence Applications product line, such as Oracle Financial Analytics. They deliver this insight across the organization to increase the company's effectiveness in managing its customers, suppliers, and financial decisions.

Oracle Procurement and Spend Analytics provides visibility into sourcing, direct and indirect spending across the enterprise, payment, and employee expenses. Oracle Procurement and Spend Analytics comprises the following Subject Areas:

- **Procurement and Spend - Change Orders**: This subject area provides the ability to report on changes to purchasing documents post approval, showing count of changes/ cancellations and processing time by Supplier, BU, Buyer, and Change Order attributes such as method, type, initiator, and so on.

- **Procurement and Spend - Invoice Lines**: This is a detailed subject area that provides the ability to report on total spend of an organization across suppliers, products, item categories, business units, cost centers, buying locations, supplier

locations and associated hierarchy. In addition, this subject area also provides detailed information at invoice distribution level.

- **Procurement and Spend - Procure to Pay**: This is a summary subject area that provides the ability to do comparative analysis and report on requested spend, committed spend and actual spend and receipts across business units, buying locations, suppliers, products, item categories and associated hierarchies for both direct and indirect spend (indirect spend being MRO and employee expenses) in detail to allow complete visibility of spending across your organization.

- **Procurement and Spend - Purchase Agreement**: This subject area provides ability to report on Purchase Agreements, showing agreement amount, its consumption and expiration, number of different agreement types, buyers, supplier and supplier sites, agreement lines across Supplier, Supplier Site, Buyer, Item, BUs, and Agreement details.

- **Procurement and Spend - Purchase Cycle Lines**: This is a summary subject area that provides the ability to report cycle time performance, such as requisition to purchase order lead time, purchase order to receipt lead time, P2P lead time of the suppliers of an organization.

- **Procurement and Spend - Purchase Orders**: This is a detailed subject area that combines the information from Purchase Orders, Purchase Order Costs and Purchase Schedules with the ability to report on committed spend, contract compliance and Purchase orders of the suppliers of an organization across suppliers, company, products, item categories and associated hierarchies at purchase order line level.

- **Procurement and Spend - Purchase Orders BU Summary**: This is the same as 'Procurement and Spend - Purchase Orders' Subject Area, except that they do not have data security enabled, and is used in Fusion Applications embedded analyses only by explicit data filter.

- **Procurement and Spend - Purchase Receipts**: This is a detailed subject area that provides the ability to report on actual spend and purchase receipts of the suppliers of an organization across suppliers, company, location, products, item categories and associated hierarchies at purchase receipt line level, including reporting based on receiving time.

- **Procurement and Spend - Purchase Requisition BU Summary**: This is the same as 'Procurement and Spend - Purchase Receipts' Subject Area, except that they do not have data security enabled, and is used in Fusion Applications embedded analyses only by explicit data filter.

- **Procurement and Spend - Purchase Requisition Status**: This is a summary subject area that provides the ability to report on requisition status along the approval cycle of purchase requisitions of the suppliers of an organization. This subject area is only populated by the Universal adapter.

- **Procurement and Spend - Purchase Requisitions**: This is a detailed subject area that provides the ability to report on requested spend and purchase requisitions (including cyclic requisitions) of the suppliers of an organization across suppliers, company, products, item categories and associated hierarchies at purchase requisition line level.

- **Procurement and Spend - Scorecard**: This subject area supports Procurement Scorecard. It includes metrics/ KPIs and its targets that provide the ability to monitor and analyze trends of procurement organization's performance. It provides performance and goal attainment information, across time and business

units, from different perspectives such as finance, internal customer, operations and supplier.

- **Supplier Performance - Supplier AP Transactions**: This is a summary subject area that provides the ability to analyze payment performance and payment due analysis of the suppliers of an organization across suppliers, company, location, products, commodities and associated hierarchies. (Note: In order to populate Supplier Payables component, you must implement the Accounts Payables module of Oracle Financial Analytics. If you do not implement the Accounts Payables module, then some of the Supplier Payables analyses will not be populated.)

- **Supplier Performance - Supplier Performance**: This subject area (built on Purchase Cycle Lines) contains targeted metrics that allow users to analyze the timeliness, reliability, cost, and quality of goods provided by the suppliers. It helps you to understand how well suppliers are contributing to the success of your organization.

- **Sourcing - Award**: This subject area provides the ability to report on Sourcing Awards, showing projected and realized savings, award amount, quantity, price, PO amount, number of suppliers and BUs awarded across sourcing negotiation types, BUs, Suppliers, Buyers and Categories.

- **Sourcing - Negotiation**: This subject area provides the ability to report on Sourcing Negotiations, showing negotiation amounts, header/ line counts and cycle times across sourcing negotiation types, BUs, Suppliers, Buyers and Categories.

- **Sourcing - Overview**: This is a detailed subject area that provides the ability to report on supplier participation and response to sourcing documents, projects and realized savings, award amount, quantity, price, PO amount, number of suppliers and BUs awarded, and various cycle times across Sourcing negotiation types, BUs, Suppliers, Buyers and Categories.

- **Sourcing - Response**: This subject area provides the ability to report on Sourcing Responses, showing supplier response and participation across sourcing negotiation types, BUs, Suppliers, Buyers and Categories.

- **Employee Expenses - Credit Card**: This subject area provides the ability to report on the corporate card spend of an organization, showing the number and amount of outstanding transactions by business unit, employee, and expense categories.

- **Employee Expenses - Overview**: This is a detailed subject area that provides the ability to report on employee spend of an organization across employees, company, cost center and associated hierarchies, including Approvers and cycle time measurements related to Approval, and Employee Expenses by various expense types.

- **Employee Expenses - Violations**: This subject area provides the ability to report on policy violations for submitted employee expenses of an organization, across employee and business.

B.54 Overview of Oracle Supply Chain and Order Management Analytics

The Oracle Supply Chain and Order Management Analytics application for Fusion Applications allows you to analyze:

- Bookings

- Financial and Operational Backlogs

- Invoices

- The movement of sales orders through different stages of the sales cycle

- Orchestration orders analysis

- Order Hold analysis

- Inventory held by an organization

- Inventory movements in, out, and through manufacturing plants, distribution centers, or storage locations

- Inventory Valuation

- Inventory cycle count with Hit or Miss and Exact Match analysis

- Product Information Management covering analytics for Item, Item-Batch and Item-Catalog attributes

- Product Information analytics to support New Item Requests and Change Order processes in Fusion

The Oracle Supply Chain and Order Management Analytics application consists of orders, invoices, order orchestration, backlogs, inventory, logistics and product information management. Sales orders are the entry point for the sales process. Invoices are the exit point from the fulfillment process. Backlogs are points of congestion in your fulfillment process. This coverage includes insight into orchestration orders and process durations and which items are booked, backlogged, and invoiced. This allows you to evaluate the sales performance of individual sales representatives or departments. Oracle Supply Chain and Order Management Analytics application also provides you with information on Inventory Transactions, Inventory Balances and Customer and Supplier Returns. This enables companies to monitor inventory levels trend to Sales performance to improve cost exposure, increase turnover through inventory level reduction and increased velocity, properly deploy inventory at the right place / right time and better understand Customer and Supplier Returns to maintain quality.

In addition to the above, the Oracle Supply Chain and Order Management Analytics for has new content for Fusion Applications source that includes new subject areas in Costing, Distributed Order Orchestration, Logistics and Product Information Management.

B.55 About Configuring Initial Extract Date

Initial Extract Date is required when you extract data for a full load. It reduces the volume of data in the initial load. The specified initial extract date will be used as a filter on the creation date of the transactional data in the selected full extract mapping. The default date is January 01, 1970.

When you set the Initial Extract Date parameter, ensure that you set it to the beginning of an accounting period and not a date in the middle of an accounting period. For example, if you decide to extract data from June 2005, and the June 2005 accounting period starts on June 5, set the date to June 5, 2005.

The following tables use INITIAL_EXTRACT_DATE:

```
PROJECTS:
W_PROJ_BUDGET_F
W_PROJ_COMMITMENT_F
W_PROJ_COMMITMENT_SNP_F
W_PROJ_CONTRACT_LINE_F
```

```
                    W_PROJ_COST_LINE_F
                    W_PROJ_CROSS_CHARGE_DIST_F
                    W_PROJ_FORECAST_F
                    W_PROJ_FUNDING_LINE_F
                    W_PROJ_INVOICE_DIST_F
                    W_PROJ_REVENUE_LINE_F

                    FINANCE:
                    W_GL_OTHER_F
                    W_GL_BALANCE_F
                    W_GL_REVN_F
                    W_GL_COGS_F
                    W_GL_COST_REVN_F
                    W_AP_HOLDS_F
                    W_FA_BALANCE_F
                    W_FA_XACT_F

                    OM:
                    W_SALES_ORDER_LINE_F
                    W_SALES_INVOICE_LINE_F
                    W_SALES_SCHEDULE_LINE_F
                    W_SALES_PICK_LINE_F
                    W_SALES_ORDER_HOLD_F
                    W_SALES_ORDER_HOLD_1_F
                    W_DOO_PROCESS_F
                    W_SALES_ORDER_CREDIT_F
                    W_SALES_INVOICE_CREDIT_F

                    PIM:
                    W_ITEM_REQUEST_F
                    W_ITEM_REQUEST_STATUS_SNP_F
                    W_ITEM_INTERFACE_F
                    W_ITEM_F

                    PRM:  No INITIAL_EXTRACT_DATE usage

                    Procurement:
                    W_PURCH_RQSTN_LINE_F
                    W_RQSTN_LINE_COST_F
                    W_PURCH_AGREEMENT_HEADER_F
                    W_PURCH_AGREEMENT_LINE_F
                    W_PURCH_SCHEDULE_LINE_F
                    W_PURCH_COST_F
                    W_PURCH_RCPT_F
                    W_AP_INV_DIST_F
                    W_PURCH_CHANGE_ORDER_F

                    Sourcing:
                    W_NEG_INVITATIONS_F
                    W_NEG_LINES_F
                    W_NEG_RESPONSES_F

                    Expense:
                    W_EXPENSE_F
                    W_EXPENSE_CC_F
                    W_EXPENSE_VIOLATION_F

                    SCM:

                    W_CST_ITEM_COST_DAILY_F
```

```
W_CST_INTRANSIT_DAILY_F
W_CST_INTRAN_ACCNTED_DAILY_F
W_CST_ONHAND_ACCNTED_DAILY_F
W_CST_ONHAND_DAILY_F
W_INVENTORY_CYCLE_COUNT_F
W_PRODUCT_XACT_F


HCM:
W_WRKFC_EVT_F
```

Note: HR needs a specific extract date (used in place of 'initial') such as HR_WRKFC_ EXTRACT_DATE, HR_ABSENCE_EXTRACT_DATE, HR_PAYROLL_EXTRACT_ DATE and HR_ACCRUAL_EXTRACT_DATE. The requirement for setting up the common INITIAL_EXTRACT_DATE parameter is only that this date should be earlier than any of our four specific extract date values.

B.56 About Configuring Global Currencies

Currency conversions are required because your business might have transactions involving multiple currencies. To create a meaningful report, you have to use a common currency. Oracle Business Analytics Warehouse stores amounts in the following currencies:

- Document currency. The document currency is the currency of the transaction. For example, if you purchase a chair from a supplier in Mexico, the document currency is probably the Mexican peso. Or, if you made a business trip to the United Kingdom and filed an expense report for meal expenses in the UK, the document currency of the expense report will most likely be in GBP.

- Local currency. The local currency is the base currency of your ledger, or the currency in which your accounting entries are recorded in.

- Global currencies. Oracle BI Applications provides three global currencies, which are the common currencies used by Oracle Business Analytics Warehouse. For example, if your organization is a multinational enterprise that has its headquarters in the United States, you probably want to choose US dollars (USD) as one of the three global currencies.

  The global currency is useful when creating enterprise-wide analyses. For example, a user might want to view enterprise-wide data in other currencies. For every monetary amount extracted from the source, the load mapping loads the document and local amounts into the target table. It also loads the exchange rates required to convert the document amount into each of the three global currencies. For fact tables, there are two amount columns covering the Local currency amount and the Document currency amount. In addition, there are three columns covering the Global currency (for example, global _amount1) and their corresponding exchange rate columns.

  In most cases, the source system provides the document currency amount, which is the default setting for handling currency. If the source system provides only the document currency amount, the source adapter performs lookups to identify the local currency codes based on the source system the appropriate currencies are assigned. After the lookups occur, the extract mapping provides the load mapping with the document currency amount and the document and local currency codes. The load mapping will then use the provided local currency codes and perform currency conversion to derive the local amount. The load mapping will also fetch the global currencies setup from DAC parameters and look up the corresponding exchange rates to each of the three global currencies.

To specify global currencies, use the parameters GLOBAL1_CURR_CODE, GLOBAL2_CURR_CODE, and GLOBAL3_CURR_CODE.

B.57 About Configuring Calendars

This topic contains additional information about using FSM to configure calendars for Oracle BI Applications Offerings.

B.57.1 About Specifying a Gregorian Calendar Date Range

This task is a pre-requisite for all types of calendars. Calendar date ranges are set by parameters START_DATE and END_DATE. These parameters decide the dates that will be loaded into W_DAY_D and in related calendar tables like W_MCAL_PERIOD_D and W_MCAL_DAY_D. The START_DATE needs to be set to the oldest transaction date that will be brought into the warehouse. The END_DATE needs to be large enough to cover at least 10 years in the future from the date of implementation. Example values can be 1980 to 2020. If your oldest transaction starts in 2000, then there is no need to go as far as 1980. The parameters START_DATE and END_DATE need to include all dates covered by any of the fiscal calendars brought into the warehouse as well. These parameters are the boundaries for the date dimension and related tables.

B.57.2 About Defining an Enterprise Calendar

An Enterprise calendar (or reporting calendar) enables cross Subject Area analysis. Enterprise calendars can be set to one of the OLTP sourced fiscal calendars or to one of the warehouse generated calendars. This can be done by setting the parameters GBL_CALENDAR_ID and GBL_DATSOURCE_NUM_ID. The Enterprise Calendar is set to one of the pre-seeded generated calendars.

If you are setting a warehouse generated calendar as the Enterprise calendar:

GBL_CALENDAR_ID: Should be the CALENDAR_ID of the Generated Calendar (4-4-5 or 13 period type of Calendars). By default the 4-4-5 calendar has a CALENDAR_ID of '10000' and the 13-period calendar has a CALENDAR_ID of '10001'.

GBL_DATASOURCE_NUM_ID: If Global Calendar is Generated Calendar, then it should be the DATASOURCE_NUM_ID value of the OLAP (Oracle Business Analytics Warehouse).

If you are setting a Fusion Applications source calendar as the Enterprise calendar:

GBL_CALENDAR_ID: This parameter should be the MCAL_CAL_NAME~MCAL_PERIOD_TYPE for Non-Generated Calendars. For example GBL_CALENDAR_ID will be 'Accounting~41', if the Enterprise Calendar id='Accounting' and the calendar period_type='41'. The value of GBL_CALENDAR_ID for Fusion Applications will be GL_PERIODS .period_name~ GL_PERIODS .period_type of the particular calendar chosen as the Enterprise calendar. This calendar will be the one used for cross reporting. In this release the Enterprise calendar and CRM calendar are synonymous. Please make sure you set the value to the same value as the CRM calendar in Fusion.

GBL_DATASOURCE_NUM_ID: If Enterprise Calendar is not a Generated Calendar: It should be the DATASOURCE_NUM_ID of the source system from where the Calendar definition is taken. For example, if you have two Fusion Applications data sources, and the Global Calendar is from an data source 1, then this parameter value should specify the value of source 1.

B.57.3 About Configuring Data Load Parameters for File Based Calendars (also known as Generated Calendars)

Oracle Business Intelligence Applications 11*g* Release 1 (11.1.1) supports the following types of generated calendars:

- 13 period calendars.

- 4-4-5 calendars (and variants).

When you set up calendars, note the following:

- The W_MCAL_CONFIG_G table controls how generated calendars are created.

- If generating the 4-4-5 or 13 period calendars, W_MCAL_CONFIG_G needs to have at least one row for the 4-4-5 period or 13 period. There is no entry needed in this table for Oracle Fusion Applications source calendars.

- W_MCAL_WEEK_D will be populated only for the generated calendars (i.e. 13 period or 4-4-5 type of calendars), and hence the W_DAY_D week Enterprise columns will be null for non-generated calendars (known as OLTP sourced fiscal calendars). W_ENT_WEEK_D will not be populated if a non-generated calendar is chosen as the Enterprise Calendar.

- For 13 period calendars, there is no concept of Quarter, therefore all Quarter columns in W_MCAL_WEEK_D, W_MCAL_PERIOD_D, W_MCAL_YEAR_D will be null. W_ENT_QTR_D will not be populated if a 13 period calendar is chosen as the Enterprise Calendar.

- The following table outlines columns in the W_MCAL_CONFIG_G table, which is loaded from the file_mcal_config_g.csv.

Table B–12   Columns in configuration table W_MCAL_CONFIG_G

Column Name	Column Description
CALENDAR_ID	The ID of the calendar that is being configured. This is the primary key for this table.
CALENDAR_NAME	The name of the calendar that is being configured.
CALENDAR_CLASS	Automatically generated.
PERIOD_TYPE	The type of the calendar period that is being configured e.g. '4-4-5'.
CAL_ST_DT	The date from which the calendar generation begins. **Note**: This should cover the date range required in Oracle Business Analytics Warehouse.
CAL_END_DT	The date at which the calendar generation ends. **Note**: This should cover the date range required in Oracle Business Analytics Warehouse.
CAL_OFFSET	The offset that identifies the start date of the calendar. Valid Start Day and Offset values are: ■ Monday 0 ■ Tuesday 1 ■ Wednesday 2 ■ Thursday 3 ■ Friday -3 ■ Saturday -2 ■ Sunday -1

Table B–12   (Cont.) Columns in configuration table W_MCAL_CONFIG_G

Column Name	Column Description
REFERENCE_DATE	The date (in MMDD format) that the fiscal year starts. For example, if an organization's fiscal year is from October to September, then the REFERENCE_DATE value should be '0929'.
	The previous fiscal year must end between three days before and three days after the date specified by REFERENCE_DATE (that is, within the range (REFERENCE_DATE - 3) to (REFERENCE_DATE + 3). In other words, if a REFERENCE_DATE is '0131' for 31st January, the previous fiscal year cannot exceed 3rd February.
WEEK_ALLOCATION_RULE	This parameter determines how weeks are allocated in the calendar that is being configured. E.g. '4-4-5' or '5-4-4' or '4-5-4' or '13 period'.
Other standard columns	W_INSERT_DT, W_UPDATE_DT, TENANT_ID, X_CUSTOM etc.

- The following table outlines Parameters needed for Generated Calendars.

Table B–13   Parameters required for Generated Calendars

Parameter Name	DAC Parameter Description
13P_CALENDAR_ID	SIL_TimeDimension_MCalWeek13Period. Required if you want to populate the 13 period type of calendar in Oracle Business Analytics Warehouse. The value should be the CALENDAR_ID as defined in the W_MCAL_CONFIG_G table for the 13 period type of calendar.
445P_CALENDAR_ID	Task: SIL_TimeDimension_MCalWeek445. Required if you want to populate the 445 period type of calendar in Oracle Business Analytics Warehouse. The value should be the CALENDAR_ID as defined in the W_MCAL_CONFIG_G table for the 445 period type of calendar.

- If there is a week (starting on a Sunday and ending on a Saturday) that falls across two calendar years, the week is counted in both years. For example, the week that starts on 12/30/2007 will be counted in both 2007 and 2008. In 2007, the week start date will 12/30/2007 and the end date will be 12/31/2007. In 2008, this will be the first week with start date as 01/01/2008 and end date as 01/05/2008.

- W_DAY_D stores 31 records for each month regardless of whether the month actually has 31 days. If the month has a fewer number of days, there will be records with null values in the Calendar Date and Day Date columns. These extra records are loaded for the calculation of Period Ago metrics in the Oracle BI Repository and will not affect the ETL or reporting.

- There are some attributes on the W_DAY_D table that are not mapped in the Physical layer of the Oracle BI Repository. Therefore, before creating any new attribute in the repository, check whether the attribute is already available in the Physical layer and if it can be mapped directly.

- If your fiscal calendar contains more than 12 months, the extra months will be assigned a value of 0 for the fiscal quarter. The same holds for the fiscal trimester and fiscal half values.

- By default, Oracle BI Applications can generate up to 65536 rows. If you need more than 65536 rows, you can increase the capacity to 262144 rows (718 years) by doing the following:

1. Duplicate 'SIL_DayDimension_GenerateRows7'.

2. Rename it 'SIL_DayDimension_GenerateRows8'.

3. Run this immediately after 'SIL_DayDimension_GenerateRows7'.

B.58 About Configuring Slowly Changing Dimensions

Oracle Business Analytics Warehouse provides Category 2 slowly changing dimension (SCD) functionality, which allows you to track the history of updates to dimension records. When a record in Oracle Business Analytics Warehouse has an update, the updated information is posted into a new row and the old information is kept for historical reporting purposes.

Oracle Business Analytics Warehouse identifies and applies the slowly changing dimension logic chosen by the user after data has been extracted and transformed to be source-independent. Users may configure Oracle BI Applications to support both Category 1 SCDs, in which data is overwritten with updates, and Category 2 SCDs, in which the original records are maintained while a new record stores the updated data. Choosing Category 1 or Category 2 SCDs depends on identifying your historically significant attributes.

Users can choose Category 1 or Category 2 by setting the value for $$TYPE2_FLG to Y or N in Oracle BI Applications Configuration Manager.

The following tables have **TYPE2 defined by default** (by default, it's ON):

Common Dimensions:

W_PRODUCT_D

W_INVENTORY_PRODUCT_D

W_POSITION_D

W_USER_D

W_INT_ORG_DH

W_PARTY_ORG_D

W_PARTY_PER_D

HCM:

W_HR_PERSON_LEG_D

W_HR_POSITION_D

W_JOB_D

W_PAY_GRADE_D

W_SUPERVISOR_D and W_SUPERVISOR_STATUS_D:

Note: These are not the classical Type-2 dimensions. They have EFFECTIVE_FROM_ DT and EFFECTIVE_TO_DT and are set to Type2.

However, HCM handles the dates internally and does not rely on the SCDUpdate mappings for these two. These tables are used to build the Supervisor Hierarchy, and not exposed in RPD after the physical layer.

Finance:

W_FIXED_ASSET_D

The following tables have TYPE2 supported in applications but not set by default (by default, it is OFF; it can be turned ON if required).

Common Dimensions:

W_COST_CENTER_D

W_COST_CENTER_DH

W_BUSN_LOCATION_D

W_TERR_DH

Finance:

W_AP_TERMS_D

W_BALANCING_SEGMENT_D

W_BANK_D

W_ASSET_BOOK_D

W_ASSET_CATEGORY_D

W_ASSET_LOCATION_D

W_GL_ACCOUNT_D

W_GL_SEGMENT_D

W_NATURAL_ACCOUNT_D

W_PAYMENT_TERMS_D

CRM/OM/PIM:

No SCD2 dims

SCM/Procurement/Sourcing/Expense:

No SCD2 dims

B.59 About Configuring Reporting Parameters for Year Prompting

There is no information available for this task.

B.60 How to Configure the Customer Cost Lines and Product Cost Lines for Profitability Analytics

The following steps are required if you would like to enable and use the Customer Cost Lines and Product Cost Lines feature in Profitability Analytics. Oracle provides default configuration files and the sample data for Customer Cost lines and Product Cost lines in comma separated value (CSV) files. The content of data CSV files must be replaced with your actual cost line data.

To configure the Customer Costs Lines and Product Costs Lines:

1. Copy the following configuration and sample data files from the ORACLE_
   HOME\biapps\etl\biapps_etl\data_files\universal_sample_data folder to the
   INFA_HOME\server\infa_shared\SrcFiles folder on the Informatica Server
   machine.

   ■   file_customer_cost_line.csv

- file_product_cost_line.csv

- file_xact_type.csv

- file_chnl_type.csv

- FILE_DOMAIN.csv

Note: You can omit this step if the configuration files have already been copied to the INFA_HOME\server\infa_shared\SrcFiles folder on the Informatica Server machine.

2. Open the file_customer_cost_line.csv file in the INFA_HOME\server\infa_shared\SrcFiles folder.

3. Replace and fill in the customer costing transactions that you want to load into the Customer Cost fact table, and save.

4. Open the file_product_cost_line.csv file in the INFA_HOME\server\infa_shared\SrcFiles folder.

5. Replace and fill in the product costing transactions that you want to load into the Product Cost fact table, and save.

   The other files file_xact_type.csv, file_chnl_type.csv, and FILE_DOMAIN.csv file contain the default values of transaction types, channel types, and related domain values. You can leave these files unchanged unless you want to customize the values.

Note: These CSV files are the source data for the Customer / Product Cost Line mappings in SDE_Universal_Adaptor. The mappings will fail if above configuration steps are not completed.

C

Dialog Reference for Oracle BI Applications Configuration Manager

This section contains reference information for the User Interface for Oracle BI Applications Configuration Manager and FSM deployed with Oracle BI Applications.

This chapter contains the following topics:

C.1 Add Warehouse Domain Member/Add Target Domain Member button

Use the Add Warehouse Domain Member or Add Target Domain Member button to display the "Add Warehouse Domain Member/Add Target Domain Member dialog", which enables you to add a Domain member. For example, you might want to create a salary category called 'Range 5' so that you can map a range of values in the Source Instance to 'Range 5'.

Note: If this button is grayed out, then the selected Domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").

C.2 Add Warehouse Domain Member/Add Target Domain Member dialog

Use this dialog to create a new Domain member in Oracle Business Analytics Warehouse. For example, you might want to create a band category called 'Range 5' so that you can map a range of values in the Source Instance to the band category 'Range 5'.

Element	Description
Code	Use this field to specify a unique value that identifies the new Domain member. For example 'Range_5'.
Name	Use this field to specify a short name up to 255 characters long for the new Domain member. For example, 'Range_100,000_Plus'.
Description	(Optional) Use this field to provide additional information up to 2000 characters long about the Domain Member, to assist Functional Developers.

C.3 Define Business Intelligence Applications Instance: Source System tab

Use this tab to register a Source Instance, edit a registered Source Instance, or disable a Source Instance (for a definition of Source System, see Chapter 4.1, "Terminology"). An Oracle BI Applications Instance is an Oracle BI Applications environment consisting of one Oracle Business Analytics Warehouse and one or more Source Instances.

In a new Oracle Business Intelligence Applications installation, you need to specify at least one Source Instance.

Note: For information about using icons and menu options, Section 4.8.3, "About the Icons".

Element	Description
Source Systems	A list of currently registered Source Instances for the current deployment.
	In a new installation of Oracle BI Applications, this table is empty. To register a new Source Instance, click the Add icon to display the "Register Source dialog". For more information about the Add icon, see Section 4.8.3, "About the Icons".
	Use the Edit icon to edit an existing Source Instance using the "Edit Source dialog". You can only edit 'enabled' Source Instances. To enable or disable a Source Instance, use the options on the Actions menu.
	Columns:
	■ **Source Instance Name** - the name given by the BI Applications System Administrator to a transactional system that serves as a source of data for BI Applications data load. The Source Instance is important because functional setup tasks, and setting of values of objects such as Parameters and Domain Maps are performed in relation to a source instance. If you have more than one Source Instance (not supported in Version 11.1.1.6.0), this allows you to support heterogeneous sources and multiple instances of the same source.) A source instance has a unique Data Source Number.
	■ **Description** - additional information specified by the BI Applications System Administrator about the Source Instance.
	■ (Read-only) **Product Line** - a value that is assigned automatically when you register a Source Instance.
	■ **Data Source Number** - the unique ID used to identify the Source Instance.
	Note: The Data Source Number that you enter here must exactly match the value of the Data Source Number specified in DAC for this source.
	■ **Enabled** - When you register a Source Instance, it is enabled by default. If you disable a Source Instance, then you cannot edit setup data for that Source Instance .
	Note: If you make a mistake in the Source Instance definition that you are not able to correct, you must disable that Source Instance and register a new Source Instance. For example, if the Data Source Number is incorrect, then you cannot edit a Source Instance and correct the Data Source Number. In this scenario, you must disable the incorrect Source Instance and register a new Source Instance.
	Note: To enable a Source Instance, select the Source Instance, then choose Enable from the Actions menu. To disable a Source Instance, select the Source Instance and select Disable from the Actions menu.
Fusion Applications for: *<Source Instance Name>*	Displays the Fusion Applications (Customer Relationship Management, Financials and Supply Chain Management, or Human Capital Management) that are part of the selected Source Instance.
	Columns:
	■ **Fusion Applications** - the name of the Fusion application.
	■ **Product Line Version** - product line version of Fusion applications.
	Note: This table is read-only. You select Fusion Applications when you register a Source Instance using the Add icon or Register New Source Instance menu option.

Element	Description
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ Actions - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ View - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ Edit - for information about this option, see Section 4.8.3, "About the Icons". ■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

C.4 Define Business Intelligence Applications Instance: Target Warehouse tab

Use this tab to specify a Name and Description for Oracle Business Analytics Warehouse.

Element	Description
Name	The Name of Oracle Business Analytics Warehouse. To update this value, click Edit.
Product Line Version	The product version used to verify the Oracle Business Intelligence Applications version that is deployed.
Description	Use this field to specify additional information about Oracle Business Analytics Warehouse. To update this value, click Edit.
Data Source Number	The Warehouse Data Set Name (DSN) number, which is always 999.
J2EE Data Source Name	The unique ID to identify the source data in Oracle Business Analytics Warehouse.
Edit	Use this option to display the "Edit Business Analytics Warehouse dialog", which enables you to change the Name and Description.

C.5 Edit Business Analytics Warehouse dialog

Use this dialog to specify a name and description for Oracle Business Analytics Warehouse in your Oracle BI Applications instance.

Element	Description
Name	(Mandatory) Use this field to specify a short name to identify Oracle Business Analytics Warehouse.
Data Source Number	(Read-only) The Warehouse Data Source Number (DSN).
Description	(Optional) Use this field to provide additional information about your Business Analytics Warehouse. The Description field supports xxx characters in length.

C.6 Edit Domain Member Mappings dialog

Use this dialog to map source domain members to target domain members. You can also use this dialog to change the default member mappings to meet your business

requirements. For example, in a Country source domain, you might want 'AE United Arab Emirates' to map to the code value 'AE' in Oracle Business Analytics Warehouse.

For ranged (or banded) domains, you can use this dialog to edit and create ranges. For example, you might want to increase an existing performance range from 1 - 100 to 1 - 200. Or, you might want to create a new performance range called Performance_ Range_6 and map values between 500 and 1000 to Performance_Range_6.

For more information about how to edit domain mappings and domain ranges, see Section 6.4.5, "About Setting Up Domain Member Mappings".

> **Tip:** If you start to add a new Range Member Mapping then click Cancel, before you can cancel the dialog you must specify a value in every field. In other words, if you specify a value in the **Range Start** field and **Range End** field but not the **Code** field, an error message is displayed when you click Cancel. To cancel the dialog, you must first specify a value in every field before you click Cancel.

Element	Description
Sync to Source	Use this option to synchronize the domain members in Oracle Business Analytics Warehouse with the domain members in the source system. This process automatically inserts target members from the source domain, and automatically generates 1:1 mappings for the currently selected domain. This option does not override existing member mappings.
	When you click Sync to Source, a Warning dialog is displayed that prompts you to either click OK to continue or click Cancel to stop the synchronization process. If you click OK to continue, then you commit changes to the target Domain members, even if you do not click 'Save' or click 'Save and Close'.
	This option becomes grayed out if you start to edit mappings (for example, if you select a value from the **Code** drop down list), until you click Save, or you click Cancel then relaunch the dialog.
	Note: This option is only available for extensible non-ranged target Domains (for more information, see Section 6.4.7, "Why are some domains non-extensible?").
	For more information about using Sync to Source, see Section 6.4.5.6, "How to use Sync to Source to synchronize a Target Domain with a Source Domain".
Batch Edit	Use this field (and the accompanying Change button) to update multiple Target Domain Member Mappings with the same value.
	Select one or more member mappings in the Source Domain Member list, select a value from the **Batch Edit** drop down list, then click Change. All Target Domain Members will be updated with the value selected.
	For more information about using Batch Edit, see Section 6.4.5.7, "How to use Batch Edit to update multiple Target Domain Member Values".
Change	Use this option to apply the value selected in the **Batch Edit** drop down list to all target domains members currently selected.
	Note: If you click Change erroneously, then click Cancel to 'undo' the changes and close the Edit Domain Member Mappings dialog.

Element	Description
Add Range Member Mapping (+ icon).	Use this option to add a new Domain Member Mapping. Click the + icon to display a new row in the table, containing blank fields, which you use to specify the details of the new range.
	Note: This option is only available for Domains for which you can specify ranges. For example, a Performance Percentile domain in a HR application might have the ranges: 0 - 100 as Performance Range 1, and 101 - 200 as Performance Range 2.
	If this option is grayed out, then the target Domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").
	Note: If you start to add a new Range Member Mapping then click Cancel, before you can cancel the dialog you must specify a value in every field. In other words, if you specify a value in the **Range Start** field and **Range End** field but not the **Code** field, an error message is displayed when you click Cancel. To cancel the dialog, you must first specify a value in every field, then click Cancel again.
	If you want to assign a new range to a new category (for example, to Performance Range 3), then before you create the new range you must first use the **Add Warehouse Member** option to create the new category.
Source Members	**Note**: This option is only available for non-ranged Domain Member Mappings.
	Use this drop down list to change the type of domains displayed. Select 'Unmapped' to display only source domains that are not mapped to target domains. Select 'Mapped' to display only source domains that are mapped to target domains. The default value 'All' displays both mapped and unmapped domains.
	Note: If you selected a value in the **Source Members** field on the "Manage Domains and Mappings: Domain Mappings tab", then that selection does not persist to this dialog. For example, if you selected Not Mapped in the **Source Members** field on the "Manage Domains and Mappings: Domain Mappings tab", then by default this dialog displays both Mapped and Unmapped values.
Add Target Domain Member	Use this option to display the "Add Warehouse Domain Member/Add Target Domain Member dialog", which enables you to create a new target mapping value. The "Add Warehouse Domain Member/Add Target Domain Member dialog" prompts you to specify a Code, Name, and Description (optional) for the new target mapping.
	For example, if you have a Performance Percentile domain in a HR application with the ranges: 0 - 100 as Performance Range 1, and 101 - 200 as Performance Range 2, then you might want to create a category (or Range) called Performance Range 3, to which you can map a new range. Or, you might want to create a new domain called Payment Method in a Financial application.
	If this option is grayed out, then the target Domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").
Source Domain Member	The list of Source Domain Members for the Source Domain.

Element	Description
Target Domain Member	This column value displays the mapped Warehouse or Target member value for the Source Domain Member displayed in the Source Domain Member column.
	Use the lists in the **Code** column to change the target domain member to which the adjacent **Source Domain Member** value is mapped.
	For example, if you have used the Add Warehouse Member button to create a new target category (or Range) called '> 2000', you might want to select '> 2000' for the range that is specified with **Range Start** = 2000 and **Range End** =10,000.
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following:
	■ View - for information about this menu, see Section 4.8.2, "About the Menu Options".
	■ Detach - for information about this option, see Section 4.8.3, "About the Icons".

Related Topics

Section 4.8.1, "About the Work Area"

Section 4.8.2, "About the Menu Options"

Section 4.8.3, "About the Icons"

Section 6.4, "About Working With Domains and Domain Mappings"

C.7 Edit Parameter Value dialog (for Data Load Parameters)

Use this dialog to specify a value or set of values for the selected Data Load Parameter. For example, you might set the value of the global currency parameter GBL_CURRENCY to 'USD'.

The fields that are displayed on this dialog are different depending on the type of parameter being edited (for example, boolean, date, multi-value select list of values, number, single-value select list of values, string).

This dialog uses the following selection field types:

■ Boolean

Use the list to select either Yes or No.

■ Date

Use the field to type a date in the format MM/DD/YYYY, or use the Date Picker to select a date.

- **Table for Single Values**

  Use the table to select a single value, then click Save and Close to complete the selection.

  If the table is empty, click Retrieve Source Values first to populate the table.

- **Multi-value Select List of Values**

  Move the items that you want to select from the **Available Values** list to the **Selected Values** list. Use Ctrl + click to select multiple non-contiguous values. Use Shift + click to select multiple contiguous values.

  Click Retrieve Source Values to refresh the **Available Values**. For example, if you have access to more than one data source, you might use the **Source Instance** field to select a different data source, then you might click Retrieve Source Values to populate the **Available Values** list.

- **Number**

  Use the field to type a number, or use the spinner controls to increase or decrease the number.

- Single-value Select List of Values

  Click a value in the list, click Select, then click Save and Close. **Note**: If you do not click Select before first clicking Save and Close, the changes are not saved.

- String

  Use the field to type a value.

Related Topics

Section 4.8.1, "About the Work Area"

Section 4.8.2, "About the Menu Options"

Section 4.8.3, "About the Icons"

C.8 Edit Parameter Value dialog (for Reporting Parameters)

Use this dialog to specify a value or set of values for the selected Reporting Parameter.

The fields that are displayed on this dialog are different depending on the type of parameter being edited (for example, boolean, date, multi-value select list of values, number, single-value select list of values, string).

This dialog uses the following selection field types:

- Boolean

Use the list to select either Yes or No.

- Date

  Use the field to type a date in the format MM/DD/YYYY, or use the Date Picker to select a date.

- Table for Single Values

  Use the table to select a single value, then click Save and Close to complete the selection.

  If the table is empty, click Retrieve Source Values first to populate the table.

- Multi-value Select List of Values

  Move the items that you want to select from the **Available Values** list to the **Selected Values** list. Use Ctrl + click to select multiple non-contiguous values. Use Shift + click to select multiple contiguous values.

  Click Retrieve Source Values to refresh the **Available Values**. For example, if you have access to more than one data source, you might use the **Datasourcename** field to select a different data source, then you might click Retrieve Source Values to populate the **Available Values** list.

- Number

  Use the field to type a number, or use the spinner controls to increase or decrease the number.

- Single-value Select List of Values

  Click a value in the list, click Select, then click Save and Close. **Note**: If you do not click Select before first clicking Save and Close, the changes are not saved.

- String

  Use the field to type a value.

Related Topics

Section 4.8.1, "About the Work Area"

Section 4.8.2, "About the Menu Options"

Section 4.8.3, "About the Icons"

C.9 Edit Preferred Currency Name dialog

Use this dialog to edit the Preferred Currency names that are displayed on BI dashboards in the Currency dropdown on the My Account dialog\Preferences tab for a user logged into Oracle Business Intelligence. You can either use the default currency display names or if required you can specify different currency display names.

Element	Description
Preferred Currency Code	A read-only code that identifies a preferred currency type.

Element	Description
Preferred Currency Name	The currency label that is displayed in the Currency dropdown on the My Account dialog\Preferences tab for a user logged into Oracle Business Intelligence. If required, you can change the default values. For example, if your organization prefers to use the term 'Local Currency' instead of 'Ledger Currency', then you might use this field to change the value 'Ledger Currency' to 'Local Currency'.

Related Topics

Section 4.8.1, "About the Work Area"

Section 4.8.2, "About the Menu Options"

Section 4.8.3, "About the Icons"

C.10 Export Data dialog

Use this dialog to specify the objects that you want to export. For example, you might want to export only changes to Data Load Parameters.

Element	Description
Export File Name	Use this field to specify a name for the export (for example, June_1_setup_data). This name is displayed in the *<Export List>* on the Export Setup Data dialog, and is used as the default ZIP file name that is generated by Oracle BI Applications Configuration Manager. Alternatively, you can also use the default export file name that is displayed. The name that you specify in this field can be different to the ZIP file name that you specify when the export is complete. For example, you might specify 'ExportSetupData' in this field, but when the export is complete and you are prompted to specify a file name you might change the file name to ExportSetupData_2010-05-14.zip. **Note**: Do not specify a ZIP file extension or a file location in this field. A ZIP file extension is appended automatically during the export. You are prompted for a file location later if you choose to save the ZIP that is generated.
Setup Objects to Export	Use these check boxes to specify the objects that you want to export. For more information about what data is exported, see Section 6.13.1, "What Data is Exported?".
Export	Use this button to start the data export using the specified details to create a ZIP file. When the File Download dialog is displayed, either click **Open** to open the ZIP file in your default ZIP file tool or click **Save** to save the ZIP file to a location that you specify. Oracle BI Applications Configuration Manager creates a default ZIP file name using the value that you specified in the **Export File Name** field appended with '.zip'. **Tip**: If you Save the ZIP file, Oracle recommends that do not change the ZIP file name. If you change the ZIP file name, then the ZIP file name will be out of sync with the file name that is displayed in the **Name** field in the Export List on the "Export Setup Data dialog".

C.11 Export Setup Data dialog

Use this dialog to view details of previous exports of setup data, and to export your setup data to a ZIP file for data migration. For example, you might want to view recent exports, or migrate your setup data from a test environment to a production environment.

Note: When you export setup data, Oracle BI Applications Configuration Manager does not record the location of ZIP files that are created. In other words, you need to manually keep a note of ZIP file locations for each export.

Element	Description
Export Data	Use this button to display the "Export Data dialog", which enables you to export the setup data from the current environment to a ZIP file.
<Export List>	Use this list to view and manage previous exports. Use the **Status** column to make sure that an export was a Success. If the **Status** column reports an Error, repeat the export process and make sure that you specify a name only in the **Export File Name** field. Do not specify a ZIP extension or a file location.
Export Details	Use this pane to view the Setup Objects, Table Names, and Row Count values for currently selected export in the **Export Setup Data** list above.

Related Topics

Section 6.13, "About Exporting and Importing Setup Data for Oracle BI Applications Configuration Manager"

Section 6.13.1, "What Data is Exported?"

C.12 Edit Source dialog

Use the Edit Source dialog to edit the details for the currently selected Source Instance.

You need to specify one Source Instance for the database instance in your Oracle Fusion Applications deployment. In a new Oracle Business Intelligence Applications installation, you need to specify one Source Instance.

For example, if you are deploying Oracle Fusion Applications in one Fusion Applications instance, you might specify a Source Instance called 'Oracle Fusion Applications', specify a unique data source number, and select all three Fusion Applications in the **Fusion Applications** list.

Element	Description
Product Line	(Read-only) A value that is assigned automatically when you register a Source Instance.
Data Source Number	(Read-only) A unique ID for the Data Source that identifies data in the Business Analytics Warehouse. The Data Source Number specified here must match the Data Source Number specified in the Physical Data Source connection in DAC for the Source with the Connection Type of 'Warehouse'. **Note**: If the Data Source Number is incorrect, then you must disable this incorrect Source Instance and register a new Source Instance with a corrected Data Source Number.

Element	Description
Source Instance Name	Specify a Source Instance name to identify the database instance. For example, if you are deploying all three Oracle Fusion Applications in a Fusion Applications instance, then you need to register one Source Instance. In this case, you might create a Source Instance named 'Oracle Fusion Applications'.
Description	(Optional) Additional information about the Source Instance, to assist BI Implementors.
Fusion Applications	Select the **Select** check box for each application that is deployed in the Oracle Fusion Applications database instance. For example, if you have all three Fusion Applications deployed in a Fusion Applications instance, select all three check boxes.
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ View - for information about this option, see Section 4.8.2, "About the Menu Options". ■ Detach - for information about this option, see Section 4.8.3, "About the Icons".

Related Topics

Section 4.8.1, "About the Work Area"

Section 4.8.2, "About the Menu Options"

Section 4.8.3, "About the Icons"

C.13 Import Data dialog

Use this dialog to specify the name and location of the ZIP file that you want to import. For example, if you have previously exported data to a file called 'C:\temp\ExportSetupData_2010-02-01 07:43:15.0.ZIP', use this dialog to specify 'C:\temp\ExportSetupData_2010-02-01 07:43:15.0.ZIP'.

Element	Description
Import File Location	Use this field to specify the name and location of the ZIP file to import. Type in a file location and file name or use the Browse button to locate and select a file using the Choose File dialog.
OK	Use this button to import the specified file. When the import is complete, use the *<Import List>* on the Import Setup Data dialog to monitor the progress.

Related Topics

Section 6.13, "About Exporting and Importing Setup Data for Oracle BI Applications Configuration Manager"

Section 6.13.1, "What Data is Exported?"

C.14 Import Setup Data dialog

Use this dialog to view details of previous imports of setup data, and to import setup data from a ZIP file. For example, you might want to view recent imports, or migrate your Setup Data from a test environment to a production environment.

Element	Description
Import Data	Use this button to display the "Import Data dialog", which enables you to import setup data from a ZIP file located on a local or network drive.
<Import List>	Use this list to view and manage previous imports.
Import Details	Use this pane to view the Setup Objects, Status, Table Name, and Import Start Date of the currently selected import in the *<Import List>* above.

Related Topics

Section 6.13, "About Exporting and Importing Setup Data for Oracle BI Applications Configuration Manager"

Section 6.13.2, "What Data is Imported?"

C.15 Manage BI Applications: BI Applications Offerings tab

Use this tab to enable and disable Oracle BI Applications Offerings. Before you can deploy BI Applications Offerings, you must enable the Offerings that you have purchased. When an Offering is enabled, the Setup Data relating to that Offering is made visible in Oracle BI Applications Configuration Manager.

Note: For information about using icons and menu options, Section 4.8.3, "About the Icons".

Element	Description
BI Applications Offerings	Use this list to view available Offerings. Use the **Enabled** check box to enable or disable an Offering.
Associated BI Application Modules and Functional Areas	This read-only list shows the modules that make up the currently selected Offering in the **BI Applications Offering** list above.
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ View - for information about this menu, see Section 4.8.2, "About the Menu Options".

Related Topics

Section 4.1, "Terminology" (for more information about Offerings).

C.16 Manage BI Applications: Warehouse Languages tab

Use this tab to specify the languages for which data will be loaded into the Business Analytics Warehouse during the ETL process. When you install Oracle Business Intelligence Applications, the American English Language is enabled by default. You must enable the languages that you want to deploy.

You can also specify a Base Language. The Base Language is used if the ETL process cannot located data in the installed language. For example, if the French is the installed language and American English is the Base Language, then if only ninety percent of your source data is in French, the ETL process will extract French data where available and extract American English data for the other ten percent.

Note: For information about using icons and menu options, Section 4.8.3, "About the Icons".

Element	Description
Manage Business Analytics Warehouse Languages	Use this list to view the available languages, and enable the languages that you want to support. Use the **Installed** column to change the default value of 'Disabled' to 'Installed', which enables the selected language. The other table columns display read-only data.
Set Base Language	Use this option to specify the Base Language, which is then marked with a blue dot. You can only specify one Base Language. If the current Base Language record is selected, then this icon is grayed out. To select a different Base Language, select a different language, then click the **Set Base Language** icon.
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ View - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ Query By Example - for information about this option, see Section 4.8.3, "About the Icons". ■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

C.17 Manage Data Load Parameters dialog

Use this dialog to verify that Data Load Parameter values have been set correctly, and edit values where necessary. For example, you might want to change a Global Currency Code value from USD to EUR.

The **Data Load Parameters** pane (or master table) displays parameters and values, and the **Group Specific Parameter Values for:** pane displays associated Fact Groups or Dimension Groups (if there are any) and their values. You can specify different values for associated Fact Groups and Dimension Groups, except for Non-Overridable Application Specific parameters (for more information, see Section 6.5.1, "Key points about working with Data Load Parameters").

Tip: Use the Restore Pane arrow in the bottom right-hand corner of the dialog to display parameter details in the Context Pane (for more information, see Section 4.8.1, "About the Work Area").

Note: For information about the icons used on this dialog, see Section C.17.1, "Icons on the Data Load Parameters dialog".

Element	Description
Data Load Parameters	Use this list to navigate and edit Data Load Parameters.
	Options:
	■ <**Edit icon**> Use this option to display the "Edit Parameter Value dialog (for Data Load Parameters)", which enables you to update the currently selected Global parameter.
	Note: The Edit icon is grayed out for Application Specific parameters. You edit Application Specific parameters values in the **Group Specific Parameter Values for** pane.
	If you click Edit for an parameter that is associated with specific Fact Groups and Dimension Groups, then a Warning dialog displays a list of groups that will be affected if you specify a value.
	You can also edit a Global parameter value by clicking on the value in the **Global Parameter Value** column.
	Columns:
	■ **Instance** displays the instance to which the parameter is applicable (for example, 'Fusion Applications' for a source instance, and 'BI Applications Data-Warehouse' for a target instance).
	■ **Parameter Name** is the human-readable parameter name. For a description of icons used in this column, see Section C.17.1, "Icons on the Data Load Parameters dialog".
	■ **Parameter Code** is the underlying parameter identifier.
	■ **Global Parameter Value** displays one of the following:
	For Global parameters, the parameter value is displayed. If the parameter is associated with Fact Groups or Dimension Groups (as displayed in the **Group Specific Parameter Values for** pane), then the Global value can be overridden at the Fact Groups or Dimension Group level.
	For Application Specific parameters, the text 'Group Specific' is displayed. Use the **Group Specific Parameter Values for** pane to edit the values for associated Fact Groups and Dimension Groups.
	■ **Description** contains additional information about the parameter.
	■ **Last Updated By** displays the login details used to last update the value.

Element	Description
Group Specific Parameter Values for:	Use this list to view and edit parameter values for associated Fact Groups and Dimension Groups. The parameter values specified in this pane are specific to a Fact Group or Dimension Group. For example, you might specify the Time Grain as MONTH for the Order Fulfilment Fact Group, and the Time Grain as WEEK for the Purchase Cycle Fact Group. **Note**: Global Parameters that are not associated with any Fact Groups or Dimension Groups do not show any records in this table. **Options**: - <**Edit icon**> Use this option to display the "Edit Parameter Value dialog (for Data Load Parameters)", which enables you to update the parameter value for the currently selected Fact Group or Dimension Group. If you edit a Non-Overridable Application Specific Parameter, then a Warning dialog is displayed showing a list of Dimension Groups and Fact Groups that will be affected if you change the value. - **Edit All** Use this option to display the "Edit Parameter Value dialog (for Data Load Parameters)", which enables you to update the parameter value for all associated Fact Groups and Dimension Groups in the **Group Specific Parameter Values for** list. For Global parameters, using this option does not override the Global parameter value (that is, as displayed in the **Data Load Parameters** pane). **Columns**: - **Group** displays the Fact Group or Dimension Group to which the parameter is applicable (for example, Refresh Period applies to the Fact Group named Inventory Transactions). - **Parameter Name** is the human-readable parameter name, which is taken from the parent Data Load Parameter name. For a description of icons used in this column, see Section C.17.1, "Icons on the Data Load Parameters dialog". - **Parameter Value** is the parameter value for that Fact Group or Dimension Group. To edit the value, click the value. If a value has not been specified, click <Edit Value> to specify the value. If you edit a Non-Overridable Application Specific Parameter, a Warning dialog is displayed showing a list of Dimension Groups and Fact Groups that will be affected if you change the value. - **Last Updated By** displays the login details used to last update the value.
Search (Only displayed when this dialog is invoked directly from Oracle BI Applications Configuration Manager. Search is not available when this dialog is invoked from FSM.)	For User Assistance on the Search pane, see Section C.26, "Search pane".

Element	Description
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ View - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ Edit - for information about this option, see Section 4.8.3, "About the Icons". ■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

Related Topics

Section 4.8.1, "About the Work Area"

Section 6.5.1, "Key points about working with Data Load Parameters"

Section 6.5.4, "About Editing Application Specific Parameter Values"

Section 6.5.5, "How to Edit a Data Load Parameter value"

C.17.1 Icons on the Data Load Parameters dialog

The Manage Data Load Parameter dialog uses the following icons:

Table C–1   Icons on the Data Load Parameters dialog

Icon or Menu Option	Description
(ab)	**Application Specific Parameter or Non-Overridable Parameter** If displayed in the **Data Load Parameters** pane (or top list), then this icon indicates an Application Specific parameter. If displayed in the **Group Specific Parameter Values For** pane (or lower list), then this icon indicates a Non-Overridable Application Specific parameter, which means that each associated Fact Group or Dimension Group must have the same value. In other words, if you change the value for one Fact Group or Dimension Group, then all Fact Groups and Dimension Groups are updated with that value (if you first confirm at a Warning dialog). For more information, see Section 6.5.1, "Key points about working with Data Load Parameters".
(icon)	**Dimension Group** This icon indicates that a Group Specific Parameter applies to a Dimension Group. For more information, see Section 6.5.1, "Key points about working with Data Load Parameters".
(icon)	**Fact Group** This icon indicates that a Group Specific Parameter applies to a Fact Group. For more information, see Section 6.5.1, "Key points about working with Data Load Parameters".

Table C–1   (Cont.) Icons on the Data Load Parameters dialog

Icon or Menu Option	Description
	Global Parameter This icon indicates that the parameter is global (that is, it applies to all (or a majority of) ETL tasks). For more information Global Data Load Parameters, see Section 6.5.1, "Key points about working with Data Load Parameters".
	Read Only Parameter The Lock icon indicates that you can cannot edit the parameter value. For example, the value might be read-only, or the value might be set in a different part of the application. For example, Base Language, which you set using the "Manage BI Applications: Warehouse Languages tab" in Oracle BI Applications Configuration Manager (this option is not available in FSM).
	Overridable Parameter This icon indicates an overridable Application Specific parameter, which means that each associated Fact Group or Dimension Group can have a different value if required. For more information, see Section 6.5.1, "Key points about working with Data Load Parameters".
	Set Before A Full Load The Alert icon indicates that you must set the value of a Data Load Parameter before you perform a Full Data Load. If you change this value after you have performed a Full Data Load, then you must perform a new Full Data Load.
	Parameter Category This icon denotes a grouping of related parameters, for example, the Configure Time Dimension category is a group of parameters that is related to calendars. To expand a Parameter Category, click Expand: To collapse a Parameter Category, click Collapse:

C.18 Manage Domains and Mappings: Domain Mappings tab

Use this tab to verify that data is mapped correctly from the Source Instance to Oracle Business Analytics Warehouse, and edit the domain member mappings if required. For example, in a HR application you might want to check that the EMPLOYEE_SEX code in the Source Instance maps to W_EMPLOYEE_SEX_MF in the target system. Or, you might want to change the Range Start and Range End values for a category (or Range) mapping.

Note: When you first display this tab, the Domain mappings list is empty. Use the Search pane to specify the BI Application Module that you want to look at.

For more information about Domain Mappings, see Section 6.4.1, "About Domain Mappings and Domain Member Mappings".

Element	Description
Domain Mappings	Use this list to navigate and view the Domain Mappings. If this list is empty, use the Search pane to specify the functional area (that is BI Application Modules, Dimensions or Fact Groups) that you want to configure.
	The domains in the **Source Domain** list at the left hand side map to the adjacent domains in the **Target Domain** list at the right hand side.
	When you select a domain in the **Domain Mappings** list, domain members are displayed in the **Domain Member Mappings** list below.
Domain Member Mappings	Use this list to view and edit the Domain Member Mappings for the domain map that is currently selected in the Domain Mappings list.
	▪ Edit Domain Member Mappings (Edit icon) - Use this option to display the "Edit Domain Member Mappings dialog", which enables you to edit the member mappings.
	Note: If the Edit Domain Member Mappings icon (Edit icon) is grayed out, then the Domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").
	For more information about how to create ranges and categories, see Section 6.4.5, "About Setting Up Domain Member Mappings".
Search (Only displayed when this dialog is invoked directly from Oracle BI Applications Configuration Manager. Search is not available when this dialog is invoked from FSM.)	For User Assistance on the Search pane, see Section C.26, "Search pane".
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following:
	▪ Actions - for information about this menu, see Section 4.8.2, "About the Menu Options".
	▪ View - for information about this menu, see Section 4.8.2, "About the Menu Options".
	▪ Query By Example - for information about this option, see Section 4.8.3, "About the Icons".
	▪ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

Related Topics

Section 4.8.1, "About the Work Area"

Section 6.4.1, "About Domain Mappings and Domain Member Mappings"

Section 6.4.5, "About Setting Up Domain Member Mappings"

Section 6.4.6, "How to modify a Warehouse Domain Hierarchy"

Section 6.4.7, "Why are some domains non-extensible?"

C.19 Manage Domains and Mappings: Source Domains tab

Use this read-only tab to view the domains and domain members in the Source Instance. For example, you might want to verify that the CURRENCY domain has the correct domain members (e.g. EUR for Euro, USD for US Dollar).

Note: This tab is only display in Oracle BI Applications Configuration Manager. It is not displayed in FSM.

For more information about Domains, see Section 6.4, "About Working With Domains and Domain Mappings".

Element	Description
Source Domains	Use this read-only list to navigate the Source Domains.
Domain Members	Use this list to view the members for the currently selected domain.
Search (Only displayed when this dialog is invoked directly from Oracle BI Applications Configuration Manager. Search is not available when this dialog is invoked from FSM.)	For User Assistance on the Search pane, see Section C.26, "Search pane".
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ Actions - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ View - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ Query By Example - for information about this option, see Section 4.8.3, "About the Icons". ■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

Related Topics

Section 4.8.1, "About the Work Area"

Section 6.4.2, "About Source Domains"

Section 6.4.5, "About Setting Up Domain Member Mappings"

Section 6.4.6, "How to modify a Warehouse Domain Hierarchy"

Section 6.4.7, "Why are some domains non-extensible?"

C.20 Manage Domains and Mappings: Warehouse Domains tab

Use this tab to verify that the domains and domain members in the target system have been set correctly, and also to edit the name and description for domain members. For example, you might want to change the Name and Description for the Application transaction type for AP Transaction Type (W_AP_XACT_TYPE).

Note: This tab is only display in Oracle BI Applications Configuration Manager. It is not displayed in FSM.

For more information about Domains, see Section 6.4, "About Working With Domains and Domain Mappings".

Element	Description
Warehouse Domains	Use this list to navigate the Warehouse domains.
Warehouse Members	Use this list to view and edit the members for the currently selected domain.
Add Warehouse Domain Member	Use this option to display the "Add Warehouse Domain Member/Add Target Domain Member dialog", which enables you to create a new warehouse member. The "Add Warehouse Domain Member/Add Target Domain Member dialog" prompts you to specify a Code, Name, and Description (optional) for the new warehouse member. For example, if you have a Performance Percentile domain in a HR application with the ranges: 0 - 100 as Performance Range 1, and 101 - 200 as Performance Range 2, then you might want to create a category called Performance Range 3, to which you can map a new range. Or, you might want to create a new target domain called Payment Method in a Financial application. If this option is grayed out, then the domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").
Search (Only displayed when this dialog is invoked directly from Oracle BI Applications Configuration Manager. Search is not available when this dialog is invoked from FSM.)	For User Assistance on the Search pane, see Section C.26, "Search pane".
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ Actions - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ View - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ Query By Example - for information about this option, see Section 4.8.3, "About the Icons". ■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

Related Topics

Section 4.8.1, "About the Work Area"

Section 6.4.3, "About Warehouse Domains"

Section 6.4.5, "About Setting Up Domain Member Mappings"

Section 6.4.6, "How to modify a Warehouse Domain Hierarchy"

Section 6.4.7, "Why are some domains non-extensible?"

C.21 Manage Domains and Mappings: Warehouse Domain Hierarchies tab

Use this tab to check that the domain hierarchies have been set up correctly in Oracle Business Analytics Warehouse. You can also change how domain values are mapped from the source system to Oracle Business Analytics Warehouse.

For more information about Warehouse Domain Hierarchies, see Section 6.12, "Viewing Domain Hierarchies".

Note: Domain Hierarchies are displayed in inverted format, that is in the following format:

<Child 1>\

    <Child *n*>\

        <Parent>.

For more information, see Section 6.12, "Viewing Domain Hierarchies".

Element	Description
Domain Mappings	Use this list to navigate the domain hierarchies in Oracle Business Analytics Warehouse. For example, in a Financials target warehouse, the Group Account domain W_GL_GROUP_ACCOUNT is the parent domain to Financials Statement Item W_FIN_STMT.
	See **Note** above for more information about the tree format.
Domain Member Mappings	Use this list to view and edit the members for the domain that is currently selected in the Domain Mappings list above.
	Use the **Source Members** drop down list to change the type of domains displayed. For example, select 'Unmapped' to locate source domains that are not mapped to target domains. Select 'Mapped' to display only source domains that are mapped to target domains. The default value 'All' displays both mapped and unmapped domains.
	Edit Domain Member Mappings (Edit icon) - Use this option to display the "Edit Domain Member Mappings dialog", which enables you to edit the values in this list. If a domain has been defined as extensible, then you can also create ranges and categories (or Ranges).
	Note: If the Edit Domain Member Mappings icon is grayed out, then the Domain is non-extensible (for more information, see Section 6.4.7, "Why are some domains non-extensible?").
	For more information about how to create ranges and Ranges, see Section 6.4.5, "About Setting Up Domain Member Mappings".

Element	Description
View menu	Use the View menu options as follows: Expand - use this option to expand the currently selected node. Expand All Below - use this option to expand all levels below the currently selected node. Expand All Above - use this option to expand all levels above the currently selected node. Expand All - use this option to expand all nodes at all levels. Collapse All - use this option to collapse all nodes at all levels. Show as Top (child nodes only) - use this option to display the currently selected tree branch and hide the other tree branches. Go Up - display the next level of parent nodes. Go to Top - use this option to display the top level of parent nodes.
Search (Only displayed when this dialog is invoked directly from Oracle BI Applications Configuration Manager. Search is not available when this dialog is invoked from FSM.)	For User Assistance on the Search pane, see Section C.26, "Search pane".
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following: ■ Actions - for information about this menu, see Section 4.8.2, "About the Menu Options". ■ Query By Example - for information about this option, see Section 4.8.3, "About the Icons". ■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

Related Topics

Section 4.8.1, "About the Work Area"

Section 6.4.4, "About Warehouse Domain Hierarchies"

Section 6.4.5, "About Setting Up Domain Member Mappings"

Section 6.4.6, "How to modify a Warehouse Domain Hierarchy"

Section 6.4.7, "Why are some domains non-extensible?"

C.22 Manage Preferred Currencies dialog

Use this dialog to view and edit the preferred currency names that are used on BI dashboards in the Currency dropdown on the My Account dialog\Preferences tab for a user logged into Oracle Business Intelligence. For example, if your organization prefers to use the term 'Local Currency' instead of 'Ledger Currency', you might use this dialog to change the default value 'Ledger Currency' to 'Local Currency'.

Note: For information about using icons and menu options, Section 4.8.3, "About the Icons".

Element	Description
Preferred Currencies	Use this pane to view and edit the preferred currency names.
	To change a preferred currency name, click the name in the Preferred Currency Name column (or select that row and click the Edit icon) to display the "Edit Preferred Currency Name dialog", and use the **Preferred Currency Name** field to change the name.
Associated BI Application Modules	Use this pane to view the modules that use the currency category that is selected in the **Preferred Currencies** list.
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following:
	■ View - for information about this menu, see Section 4.8.2, "About the Menu Options".
	■ Query By Example - for information about this option, see Section 4.8.3, "About the Icons".
	■ Detach - for information about this option, see Section 4.8.2, "About the Menu Options".

C.23 Manage Reporting Parameters: Global/Application Specific tab

Use this tab to verify that Reporting Parameters have been set correctly, and edit values where necessary.

For more information about Reporting Parameters, see Section 6.6, "About Working With Reporting Parameters".

Tip: Use the Restore Pane arrow in the bottom right-hand corner of the dialog to display parameter details in the Context Pane (for more information, see Section 4.8.1, "About the Work Area").

Element	Description
Global/Application Specific Reporting Parameters	This list displays information about each reporting parameter (for example, name, code, value).
	On the Global tab, the list defaults to alphabetical order on Parameter Name.
	On the Application specific tab, the list is sub-grouped on Application, and the **Module Name** column shows the name of the application to which the parameter applies.
Parameter Details	This read-only tab displays the Name and full Parameter Description for the selected parameter.
Edit	Use the Edit icon to display the "Edit Parameter Value dialog (for Reporting Parameters)", which enables you to edit the value of the currently selected parameter.
	Alternatively, click on the parameter value in the **Parameter Value** column to display the "Edit Parameter Value dialog (for Reporting Parameters)".
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following:
	■ View - for information about this option, see Section 4.8.2, "About the Menu Options".
	■ Edit - for information about this option, see Section 4.8.3, "About the Icons".
	■ Detach - for information about this option, see Section 4.8.3, "About the Icons".

Related Topics

Section 4.8.1, "About the Work Area"

Section 6.5, "About Working With Data Load Parameters"

C.24 Overview Page for Oracle BI Applications Configuration Manager

The Overview page is the main page in Oracle BI Applications Configuration Manager. For information about getting started with Oracle BI Applications Configuration Manager, see Section 4.3, "Getting Started With Oracle BI Applications Configuration Manager".

Use the **Tasks** bar at the left hand side as follows:

Table C–2   Options on the Tasks bar in Oracle BI Applications Configuration Manager

Tasks bar categories	Use to
Overview	Use the **Overview** link to display or return to the Overview page.
System Setups	Set up the Oracle BI Applications Configuration Manager environment. For example, specify a Source Instance and Oracle Business Analytics Warehouse.
Functional Configurations	Functionally configure the installed Offerings. Select the **Perform Functional Configurations** link to start FSM. For example, you might start FSM to perform the Tasks relating to Oracle Financial Analytics.
Setup Data Maintenance and Administration	Monitor and make changes to Oracle BI Applications Configuration Manager Setup Data. For example, check for un-mapped domain values, or add domain values to Oracle Business Analytics Warehouse.
Setup Data Export and Import	Migrate Setup Data to a separate environment or backup Set Data. For example, migrate Setup Data from a test environment to a production environment.

Use the **Overview** panes as follows:

Table C–3   Options on the Overview panes in Oracle BI Applications Configuration Manager

Overview Pane	Use these options to
Getting Started	Use these options to display roadmap information.
BI Applications At A Glance	This pane displays System Setup information about the installed applications.
Alerts	This pane displays Setup Data alerts reported by Oracle BI Applications Configuration Manager.
	For example, use the **Data Load Parameters with No Values** pane to view parameters with no value specified. Use the **Domain maps with Unmapped Source Members** to view target domains with one or more unmapped source members.
	The Alerts pane only displays alerts for 'enabled' Offerings.

For general information, see:

Chapter 4, "Overview of Functional Configuration in Oracle BI Applications"

Section 4.2.1, "What is Oracle BI Applications Configuration Manager?"

Section 4.7, "About the Main Task Areas in Oracle BI Applications Configuration Manager"

For roadmap information, see:

For a Functional Configuration Roadmap, see Section 5.2, "Roadmap for Functional Configuration".

For a Setup Data Maintenance and Administration Roadmap, see Section 6.2, "Roadmap for Setup Data Maintenance and Administration".

Related Topics

For information about getting started with Oracle BI Applications Configuration Manager:

Section 4.3, "Getting Started With Oracle BI Applications Configuration Manager"

Section 4.8, "About the Oracle BI Applications Configuration Manager Work Area"

Section 4.9, "About Accessibility Features In Oracle BI Applications Configuration Manager"

For information about using Oracle BI Applications Configuration Manager:

Section 5, "Performing Functional Configuration"

Section 6, "Administering and Maintaining Functional Configuration Data"

C.25 Register Source dialog

Use this dialog to specify a Source Instance instance for Oracle BI Applications.

Oracle Business Intelligence Applications Version 11.1.1.6.0 supports Oracle Fusion Applications installed in a single database instance. In a new Oracle Business Intelligence Applications installation, you need to specify and enable at least one Source Instance.

For example, if you are deploying Oracle Fusion Applications in one Fusion Applications instance, you might specify a Source Instance called 'Oracle Fusion Applications', specify a unique data source number, and select all three Fusion Applications in the **Fusion Applications** list.

Element	Description
Source Instance Name	Specify a Source Instance name to identify the Fusion Applications source system.
	For example, if you are deploying all three Fusion Applications in one Fusion Applications instance, then you need to register one Source Instance for all three Offerings. In this case, you might create a Source Instance named 'Oracle Fusion Applications'.
Description	(Optional) Additional information, to assist BI Implementors.
Data Source Number	A unique ID for the Data Source that identifies data in the Business Analytics Warehouse. The Data Source Number entered here must match the DSN specified in the Physical connection in DAC for the Source of type BI Server.
	The default Data Source Number for Fusion Applications source is 200.

Element	Description
Select Fusion Applications	Select the **Select** check box for each application that is deployed in the Oracle Fusion Applications database instance.
	For example, if you have all three Fusion Applications deployed in one Fusion Applications instance, select all three check boxes.
Common Options	For information about using common options in Oracle BI Applications Configuration Manager, refer to the following:
	■ View - for information about this option, see Section 4.8.2, "About the Menu Options".
	■ Detach - for information about this option, see Section 4.8.3, "About the Icons".

Related Topics

Section 4.8.1, "About the Work Area"

Section 4.8.2, "About the Menu Options"

Section 4.8.3, "About the Icons"

C.26 Search pane

The Search pane is used to locate setup data for specific functional areas. For example, you might want to locate Domains and Mappings for the Financial Analytics application only. Or, you might want to locate Data Load Parameters with codes that include 'GLOBAL'.

Use the Search fields to specify search values, then click Search to locate matching Setup Data.

Note: The Manage Data Load Parameters dialog displays all values by default. Use the Search pane to restrict the list to parameters that match specified search values. The Manage domains and Mappings tabs display no values by default. Use the Search pane to locate parameters that match specified search values.

Note: If you use the **Dimension or Fact Group** field to specify a Dimension or Fact Group, then you will only return domains and mappings that are explicitly associated with the specified dimension or fact group. That is, if the fact table within the Fact Group or Dimension Group contains the Mapping. If you cannot locate a domain or mapping by specifying a Dimension Group or Fact Group, then use the 'All' option to display all domains and mappings. For example, if you use the **Dimension or Fact Group** field to select 'Purchase Orders', then no domains or mappings will be returned by the search. If you select 'All' in the **Dimension or Fact Group** list, then you will return domains and mappings related to Purchase Orders.

Element	Description
Source Instance	Use this list to specify a Fusion Applications instance, if you have more than one instance deployed.
BI Application	Use this list to locate Setup Data for specific applications. For example, if you want to configure Oracle Financial Analytics, you might select Financial Analytics.
	To display Domains and Mappings for all functional applications, leave this field empty.

Element	Description
Dimension or Fact Group	(Specific to Domains and Mappings) Use this list to locate Setup Data for specific functional areas within the BI Application specified in the **BI Application** field. For example, if you want to configure Oracle Financial Analytics, you might want to locate Setup Data GL Balance only.
Functional Area	(Specific to Manage Data Load Parameters) Use this list to locate Data Load Parameters for specific functional areas within the BI Application specified in the **BI Application** field. For example, if you want to configure Oracle Financial Analytics, you might want to locate setup data for GL Balance only.
Parameter	(Specific to Manage Data Load Parameters) Use this field to locate parameters by Name, Code, or Category. Specify all or part of a search term in the adjacent text field. For example, to locate all parameters with 'GLOBAL' in the parameter code, you might select Code from the **Parameter** list and type 'GLOBAL' in the text field.
Fact Group	(Specific to Manage Data Load Parameters) Use this list to locate parameters for a specific Fact Group within the BI Application specified in the **BI Application** field. For example, if you want to configure General Ledger, you might want to locate parameters for GL Balance only.
Show Global Parameters	(Specific to Manage Data Load Parameters) Select this check box to locate parameters that apply to all applications. If you clear this check box, you display only Data Load Parameters that apply to the application specified in the **BI Application** field.
Dimension Group	(Specific to Manage Data Load Parameters) Use this list to locate parameters for a specific Dimension Group within the BI Application specified in the **BI Application** field. For example, if you want to configure General Ledger\GL Balance, you might want to locate Setup Data for Ledger Dimension only.
Search	Use this button to start the search, using the search values specified in the search fields.
Reset	Use this button to clear the Search values. To start a new search, select new search values then click Search.

Note: This Search pane is used on the following dialogs:

- "Manage Data Load Parameters dialog"

- "Manage Domains and Mappings: Domain Mappings tab"

- "Manage Domains and Mappings: Source Domains tab"

- "Manage Domains and Mappings: Warehouse Domains tab"

- "Manage Domains and Mappings: Warehouse Domain Hierarchies tab"

C.27 Getting Started With System Setup

To get started with System Setup, see Chapter 3 in *Oracle Fusion Middleware Installation and Configuration Guide for Oracle Business Intelligence Applications*.

Index

T

U

V

W