HP Project and Portfolio Management Center

Software Version: 9.10

Operational Reporting Administrator's Guide: July 2012 Revision

Document Release Date: July 2012
Software Release Date: September 2010

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Intel®, Intel® Itanium®, Intel® Xeon®, and Pentium® are trademarks of Intel Corporation in the U.S. and other countries.

Java™ is a U.S. trademark of Sun Microsystems, Inc.

Microsoft®, Windows®, Windows® XP, and Windows Vista® are U.S. registered trademarks of Microsoft Corporation.

Oracle® is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

This manual's title page contains the following identifying information:

- Software version number, which indicates the software version
- Document release date, which changes each time the document is updated
- Software release date, which indicates the release date of this version of the software

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

h20230.www2.hp.com/selfsolve/manuals

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Why version number 9.10?

PPM Center is an integrated part of the HP BTO Operations version 9 portfolio. Using this version number aligns PPM Center with other products that are releasing in the same time frame. PPM Center 9.10 builds on PPM Center 8.0x and is an extension of that product version family. Product releases within the HP BTO Operations version 9 portfolio will feature shared technology, common platforms, integrations, solutions, upgrade tools, and professional services offerings.

The following table indicates changes made to this document.

Publication Date	Summary of Changes
July 2012	• Modified the step on how to change the default installation directory of BusinessObjects Enterprise server software: step 1 on page 26. • Added a note regarding the Materialized View `RPT_DIM_RM_RESOURCES` to *Loading PPM Center Data Into the Operational Reporting Database* on page 39 and *Loading PPM Center Data Into the Operational Reporting Database* on page 79. • Added *Oracle Trace Log Control for ETL Performance Troubleshooting* on page 107.

Support

You can visit the HP Software Support Web site at:

hp.com/go/hpsoftwaresupport

HP Software Support Online provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to:

h20229.www2.hp.com/passport-registration.html

Contents

1 Introduction

About this Guide

This document provides information about how to deploy the Operational Reporting solution for HP Project and Portfolio Management Center (PPM Center). It is written for PPM Center administrators, configurators, and DBAs. Readers are assumed to be knowledgeable about PPM Center and SAP BusinessObjects Enterprise.

This chapter provides an overview of the components and structure of the Operational Reporting solution. Later chapters in this document cover the following information:

- provides an overview of the deployment process and detailed instructions for each phase of deployment.
- Chapter 4, *Refreshing Operational Reporting Data,* on page 97 describes how the Operational Reporting database is updated with PPM Center data.

This document is written for users who are moderately knowledgeable about enterprise application development and skilled in enterprise system and database administration. Readers are assumed to be knowledgeable about PPM Center and SAP BusinessObjects Enterprise. This document is for:

- Application developers and configurators
- System and instance administrators
- Database administrators (DBAs)

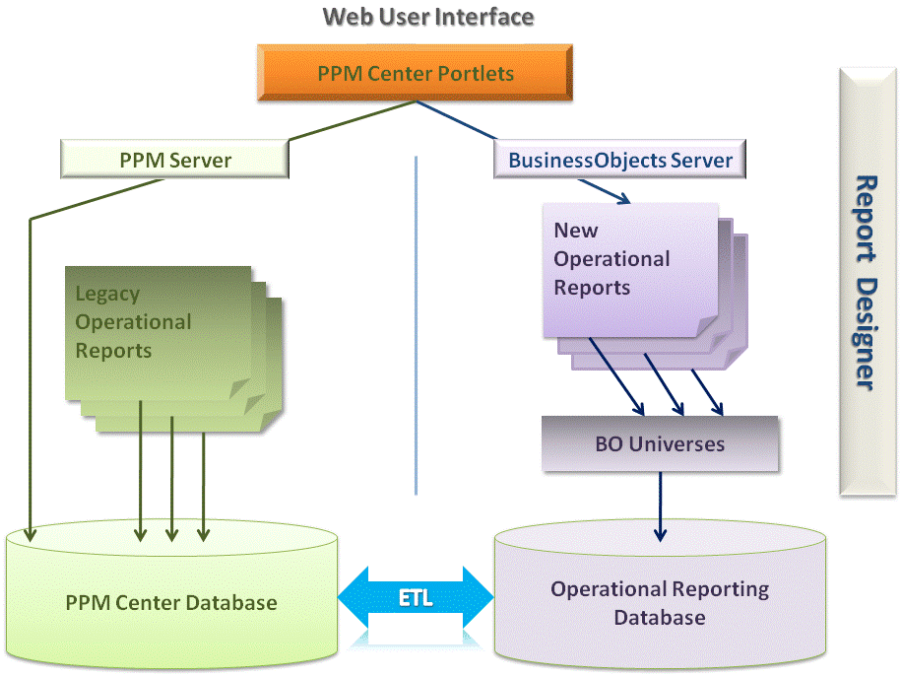
Operational Reporting Components and Architecture

Operational Reporting in PPM Center is based on SAP BusinessObjects Enterprise and Oracle. All reports and universe are created using BusinessObjects Enterprise, and all custom tables, materialized views, and stored procedures are created using Oracle database software.

For information about supported versions of BusinessObjects Enterprise and Oracle software, see the *System Requirements and Compatibility Matrix* document.

Figure 1-1 shows the high-level components of Operational Reporting in PPM Center.

Figure 1-1. Operational Reporting Architecture



Universe Hierarchy

Table 1-1 lists the universes supplied with Operational Reporting for PPM Center. These universes comprise the reporting metalayer that provide ready access to PPM Center data through the classes and objects mapped to the database.

Table 1-1. PPM Center universes for Operational Reporting

PPM Center Universe	PPM Center Module
Kernel Source Universe	N/A
RM Derived Universe	HP Resource Management
TM Derived Universe	HP Time Management
FM Derived Universe	HP Financial Management

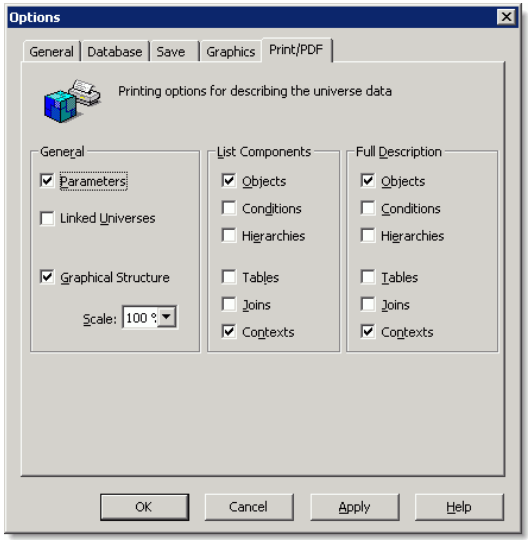
Objects and classes in the RM Derived Universe, TM Derived Universe, and FM Derived Universe are specific to data in the HP Resource Management, HP Time Management, and HP Financial Management modules, respectively. The classes and objects in the Kernel Source Universe are common to all three modules.

Viewing Detailed Information About Universe Structure

The information in later chapters of this guide represents a subset of the universe information available to you in Designer. You can see additional information about the components and structure of a PPM Center universe by saving it as a PDF file in Designer. You can select the components that you want to include in the PDF from the Options dialog box.

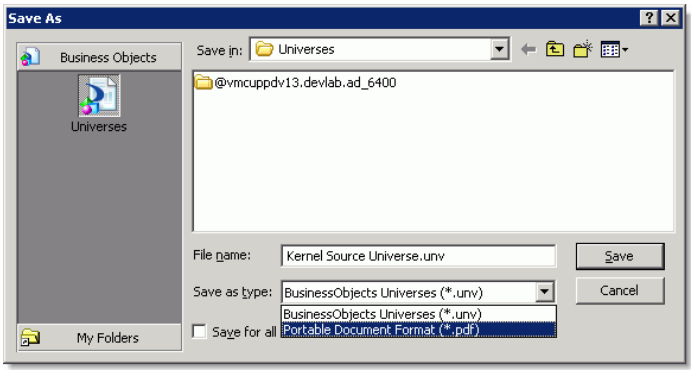
To save universe information as a PDF file:

1. Open the universe of interest in Designer.
2. From the **Tools** menu, select **Options.**



3. On the **Print/PDF** tab in the Options dialog box, select the components that you want to include in the PDF file, and then click **OK.**

4. From the **File** menu, select **Save As.**



5. In the **Save as type** list, select **Portable Document Format (*.pdf).**

Related Documents

This section lists HP documents that contain useful information for Operational Reporting administrators and users.

HP PPM Center Documents

- *System Requirements and Compatibility Matrix*

  Before you start to deploy Operational Reporting, check the *System Requirements and Compatibility Matrix* document to make sure that your operating environment meets *all* of the minimum system requirements for installing SAP BusinessObjects Enterprise (in addition to PPM Center).

- *Operational Reporting User's Guide*

  The *Operational Reporting User's Guide* provides details about how to use BusinessObjects' web desktop tool InfoView to generate operational reports on HP Resource Management, HP Time Management, and HP Financial Management data.

- *Data Model Guide*

  The *Data Model Guide* provides details about the internal structure of the data models for both PPM Center and Operational Reporting.

- *Installation and Administration Guide*

  The *Installation and Administration Guide* provides information about how to deploy, administer, and maintain a PPM Center instance.

- *Deployment Best Practices for PPM Operational Reporting*

  The *Deployment Best Practices for PPM Operational Reporting* document provides guidelines to help you deploy the complete Operational Reporting solution. It includes recommendations for deploying both the reporting schema and the BusinessObjects enterprise server.

Additional documents that you might find useful include:

- *HP Project Management User's Guide*
- *HP Project Management Configuration Guide*
- *HP Financial Management User's Guide*
- *HP Resource Management User's Guide*
- *HP Time Management Configuration Guide*
- *HP Time Management User's Guide*
- *HP-Supplied Entities Guide (includes descriptions of all portlets, request types, and workflows in PPM Center)*

To obtain any of the HP PPM Center documents listed, go to the Software Product Manuals Web site (support.openview.hp.com/selfsolve/manuals). To access this Web site, you must first set up an HP Passport account.

For more detailed information about SAP BusinessObjects Enterprise, see your SAP documentation.

2 Deploying Operational Reporting on Windows Systems

Operational Reporting Solution Deployment

This chapter provides the information you need to implement the Operational Reporting solution for PPM Center on a Windows system. It begins with an overview of the entire process, and then provides detailed instructions for each phase of deployment. For instructions on how to deploy Operational Reporting on a UNIX system, see Chapter 3, *Deploying Operational Reporting on UNIX Systems,* on page 57.

High-Level Deployment Steps

Deploying the Operational Reporting solution for PPM Center involves the following tasks:

1. Install PPM Center version 9.10, and then upgrade to PPM Center 9.10 service pack 2 (SP2).
2. To make sure that your system meets the requirements for BusinessObjects Enterprise installation, check the *Products Availability Report (PAR)* document, which is available on the BusinessObjects support site (support.businessobjects.com/documentation/supported_platforms).
3. (Optional, but strongly recommended for optimal performance) Set up an Oracle database instance specifically for Operational Reporting and set Oracle database parameters. (See *Setting Up a Database for Operational Reporting* on page 20).
4. Check to make sure that the PPM Center database and the Operational Reporting database can communicate over the database link.
5. Create four Oracle tablespaces required for the Operational Reporting schema and database objects. (See *Creating Tablespaces for the Operational Reporting Schema* on page 21.)
6. Download the Oracle 11g database client software and install it on both your BusinessObjects server and client machine.
7. Set the JAVA_HOME variable on the BusinessObjects server. (See *Setting the JAVA_HOME Environment Variable* on page 30.)
8. Install the SAP BusinessObjects Enterprise software and, optionally, the BusinessObjects Enterprise Client Tools software. (See *Installing BusinessObjects Enterprise on a Windows System* on page 26.)

9. Upgrade the BusinessObjects instance with BusinessObjects XI 3.1 Service Pack 2, and, optionally, upgrade the BusinessObjects Enterprise Client Tools software. (See *Installing BusinessObjects Enterprise XI 3.1, Service Pack 2* on page 27.)
10. Run the BusinessObjects Diagnostic Tool to verify successful BusinessObjects Enterprise installation and upgrade. (See *Verifying Successful BusinessObjects Enterprise Installation* on page 30.)
11. Set up the Oracle JDBC driver to establish connections between the BusinessObjects server and the Operational Reporting databases. (See *Configuring the Oracle JDBC Driver* on page 31.)
12. Import the PPM Center reporting universes and preconfigured reports. (See *Importing Universes and Reports* on page 32.)
13. Run the setup script to create the Operational Reporting database schema. (See *Creating the Operational Reporting Database Schema* on page 34.)
14. Run the load script to bring PPM Center data into the Operational Reporting database schema. (See *Running the Load Script* on page 39.)
15. Change the default password for the BusinessObjects Central Management Server (CMS). (See *Changing the BusinessObjects Central Management Server Password* on page 45.)
16. Configure the Operational Reporting database connection. (See *Configuring the Operational Reporting Database Connection* on page 42.) Change the connection parameters for all the universes so that the connection points to the Operational Reporting database schema.
17. To verify successful deployment of Operational Reporting, run the query for an HP-supplied report.
18. (Optional) Configure multilingual support for BusinessObjects Enterprise. (See *Configuring Multilingual Support* on page 47.)

Preparing the Database Schema for Operational Reporting

The following sections provide instructions on how to prepare the Operational Reporting database schema.

Setting Up a Database for Operational Reporting

Requirements and recommendations for setting up the database for Operational Reporting are as follows:

- (Required) Configure the Operational Reporting database to use AL32UTF8 encoding.
- (Required) Set the Oracle `NLS_CHARACTERSET` parameter to `AL32UTF8`.
- HP strongly recommends that you create an Oracle database specifically for Operational Reporting (independent of your Oracle Database instance). Make sure that the PPM Center database and the Operational Reporting database can communicate over the database link.
- HP strongly recommends that you use the Enterprise Edition of Oracle Database for the Operational Reporting database. The advanced compression and partitioning featured in the Enterprise Edition significantly improve performance, especially if you report on a large and growing volume of data.

Configuring Oracle Database Parameters for Operational Reporting

HP recommends that you use Oracle's automatic memory management (AMM) feature. To do this, set the value for either the `memory_max_target` parameter or the `memory_target` parameter, and then let Oracle manage the memory (SGA and the PGA) dynamically. For more information about how to optimize performance, see the *Deployment Best Practices for PPM Operational Reporting* document.

To obtain the *Deployment Best Practices for PPM Operational Reporting* document and other HP PPM Center documents, go to the Software Product Manuals Web site (support.openview.hp.com/selfsolve/manuals). To access this Web site, you must first set up an HP Passport account.

Creating Tablespaces for the Operational Reporting Schema

Before you can create the database schema for Operational Reporting, you must first create tablespaces (two data and two index tablespaces) for the star schema. This section sections provides instructions for performing this task.

To create the empty database schema (with tables to be populated during installation):

1. Set up the required data and index tablespaces for the Operational Reporting database schema.

For information on the minimum size recommended for these tablespaces, see the *System Requirements and Compatibility Matrix*.

2. Create two tablespaces that include the LOGGING option, as shown in the following examples:

```
CREATE TABLESPACE <PPM_DATA_TS>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
LOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;

CREATE TABLESPACE <PPM_INDEX_TS>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
LOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;
```

3. To improve performance and reduce resource consumption, create two tablespaces that include the NOLOGGING option, as shown in the following examples:

```
CREATE TABLESPACE <PPM_DATA_TS_NL>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
NOLOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;

CREATE TABLESPACE <PPM_INDEX_TS_NL>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
NOLOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;
```

The Operational Reporting database schema is created automatically during deployment.

Deploying BusinessObjects Enterprise

This section contains information about the operating systems and languages supported by the Operational Reporting solution, instructions on how to prepare your system for BusinessObjects Enterprise installation, and the detailed steps to perform the installation.

Operating Systems Support for BusinessObjects Enterprise

BusinessObjects Enterprise XI 3.1 is supported for Windows, Linux, HP-UX IBM AIX, and Sun Solaris operating systems. For information about the specific versions of the operating systems supported, see the *Products Availability Report (PAR)* document, which is available on the BusinessObjects support site (http://support.businessobjects.com/documentation/supported_platforms).

Preparing to Install BusinessObjects Enterprise

This section addresses the tasks to perform before you start to install BusinessObjects Enterprise.

To prepare your system for BusinessObjects Enterprise installation, do the following:

1. Install all necessary service packs and packages for your operating system.
2. Install PPM Center version 9.10, and then upgrade to PPM Center 9.10 service pack 2 (SP2).

   For information about how to install PPM Center 9.10 and service packs, see the *Installation and Administration Guide* for PPM Center 9.10 or the *Release Notes* for the specific service pack.

3. Check to make sure that your system meets the following minimum disk space requirements for BusinessObjects Enterprise installation:
   - 8.0 GB for BusinessObjects Enterprise (BusinessObjects Server and BusinessObjects Client)
   - 3.0 GB for BusinessObjects Enterprise Client
4. The distribution DVD contains the PPM Center Operational Reporting install bundle, the BusinessObjects Enterprise XI 3.1 install bundle, and the BusinessObjects Enterprise XI 3.1 SP2 Upgrade bundle.

   Extract the entire contents of the distribution DVD to a new folder (hereinafter referred to as the `<Op_Reports_Home>` directory) on the machine that is to host BusinessObjects Enterprise.
5. Make sure that an additional 2 GB is available on your `C:\` drive for Windows installer. (Windows installer creates install patches under the `C:\Windows\Installer` folder.)
6. Log on to the system as a user with administrator privileges.
7. Make sure that the `TEMP` environment variable points to a valid folder. This folder will contain temporary files during BusinessObjects Enterprise installation and upgrade.

8. BusinessObjects Enterprise installation and upgrade are memory- and CPU-intensive processes. Shut down all unnecessary processes before you perform the installation (and upgrade).

HP recommends that you have only the Business Object Enterprise installation running.

For more information about the hardware and software requirements for installing and upgrading BusinessObjects Enterprise, see your SAP documentation.

Installing BusinessObjects Enterprise on a Windows System

To install BusinessObjects Enterprise server software on a Windows system:

1. If you want to install the software somewhere other than the default directory (`C:\hp\ppm\reporting\boe31`):

   a. Navigate to the `<Op_Reports_Home>\Deployment\platform\installer` folder and open the `windows.ini` file in a text editor.

   b. Replace the default installation paths for the BusinessObjects Enterprise installation parameters `INSTALLDIR` and `AS_DIR` with your values. For example, set the parameter values as follows:

   ```
   AS_DIR="F:\hp\ppm\reporting\boe31\Tomcat55"
   INSTALLDIR="F:\hp\ppm\reporting\boe31\"
   ```

   c. Save and close the `windows.ini` file.

HP does not recommend changing the default installation directory. If the default installation directory is satisfactory, there is no need to change any parameter values.

2. Navigate to the `<Op_Reports_Home>\Deployment` folder and run the `installReportingServer.bat` file.

   BusinessObjects Reporting Server installation begins. You can monitor the installation process by viewing the `BOInstall.log` file, which is located in the `%TEMP%` folder.

   The BusinessObjects XI 3.1 server is installed in the `C:\hp\ppm\reporting\boe31` directory (hereinafter referred to as `<BO_Home>`). Depending on the resources available to you, installation may take several hours.

3. After you finish installing BusinessObjects XI 3.1, do the following:

   - Install BusinessObjects XI 3.1 Service Pack 2. (See *Installing BusinessObjects Enterprise XI 3.1, Service Pack 2* on page 27.)
   - Perform required post-installation tasks. (See *Post-Installation Tasks* on page 29.)

Installing BusinessObjects Enterprise XI 3.1, Service Pack 2

After you have successful installed BusinessObjects XI 3.1, you must install BusinessObjects XI 3.1 Service Pack 2 (SP2). For information about the requirements for installing BusinessObjects XI 3.1 SP2, see the *Products Availability Report (PAR)* document, which is available on the BusinessObjects support site (http://support.businessobjects.com/documentation/supported_platforms).

To install BusinessObjects XI 3.1 SP2 on Windows:

1. The SP2 upgrade is memory- and CPU-intensive. Before you begin, shut down any processes that are not absolutely required during the upgrade, including the Tomcat Windows service.
2. Navigate to the `<Op_Reports_Home>\Deployment\platform\installer` directory and open the `windows_sp2.ini` file in a text editor.
3. Replace the default values for the parameters listed in the following table based on your BusinessObjects settings.

Parameter	Value
AS_DIR	BusinessObjects installation directory (`<BO_Home>`)
INSTALLDIR	BusinessObjects installation directory (`<BO_Home>`)
NAMESERVER	Name of your local host
SS_INDEX_LOCATION	BusinessObjects installation directory (`<BO_Home>`)
CMSPASSWORD	Password for BusinessObjects Central Management Server (CMS)
NSPORT	Replace the existing value with the BusinessObjects CMS port number

4. Check to make sure that the directory specified by the TEMP environment variable exists. BusinessObjects uses this folder as a temporary log location.

5. Navigate to the <*Op_Reports_Home*>\Deployment directory, and then run the upgradeReportingServer.bat file.

The upgrade takes a few hours to complete. To monitor the progress of the upgrade, check CPU usage, process (setup.exe, msi*.exe), disk usage, and the log file.

6. (Optional) To upgrade BusinessObjects client tools, run the upgradeClientTools.bat file.

7. Check the PPM Center *Release Notes* to see whether additional BusinessObjects Enterprise service packs or fix packs are required for Operational Reporting deployment and perform any additional installations required.

Checking the Deployment Log File After Service Pack 2 Installation

If you install a BusinessObjects Enterprise service pack, the BusinessObjects Web application is automatically re-deployed. After you install BusinessObjects XI 3.1 SP2, do the following:

1. Navigate to the <*BO_Home*>\deployment\workdir directory and check the wdeploy.log file for any errors that may have occurred.

2. If errors occurred during installation, or if you cannot run a report from InfoView because of JavaScript errors, then manually redeploy BusinessObjects Enterprise as follows:

   a. Back up the <*BO_Home*>\deployment\workdir folder.

   b. Delete all contents of the <*BO_Home*>\deployment\workdir folder.

   c. Change to the <*BO_Home*>\deployment directory, and then run the command wdeploy.bat tomcat55 deployall.

3. Check the wdeploy.log file again for errors, and then run a report query from InfoView to test the deployment.

Verifying the Upgrade to BusinessObjects XI 3.1 SP2

After installation, go to the `<BO_Home>\BusinessObjects Enterprise 12.0\Logging` directory and check the `BOE_SP2_Install_0.log` file to make sure that the BusinessObjects XI 3.1 SP2 installation was successful.

Next, complete the tasks described in *Post-Installation Tasks* on page 29.

Post-Installation Tasks

This section addresses the following tasks, which must be performed after you install and update BusinessObjects Enterprise:

- *Verifying Successful BusinessObjects Enterprise Installation*
- *Setting the JAVA_HOME Environment Variable*
- *Configuring the Oracle JDBC Driver*
- *Importing Universes and Reports*
- *Creating the Operational Reporting Database Schema*
- *Changing the BusinessObjects Central Management Server Password*
- *Verify Successful Operational Reporting Deployment*
- (Optional) *Configuring Multilingual Support*

Verifying Successful BusinessObjects Enterprise Installation

After you install BusinessObjects Enterprise, you can use SAP's Deployment Diagnostic Tool to check your installation. The Deployment Diagnostic Tool is installed automatically with BusinessObjects XI Enterprise.

To verify that the BusinessObjects Enterprise installation was successful:

1. Select **Start > Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > Diagnostic Tool.**

The BusinessObjects Enterprise default password is "admin123" (Windows).

2. Make sure that the following diagnostic tests are passed:
   - Log On/Off
   - InfoView
   - Web Intelligence
   - Stop/Start Servers

For detailed information about the diagnostic tests and how to run them, see SAP's *BusinessObjects Enterprise XI 3.1 Deployment Diagnostic Tool User's Guide.*

Setting the JAVA_HOME Environment Variable

You must set JAVA_HOME in the system environment of the user account to be used to start the BusinessObjects server. It is important that the JAVA_HOME environment variable be set for the same shell and for the same user running the installation.

On the BusinessObjects server, set JAVA_HOME to:

```
<BO_Home>\javasdk
```

Make sure that JAVA_HOME value you specify contains no spaces.

Configuring the Oracle JDBC Driver

Operational Reporting deployment relies on the Oracle JDBC driver to establish connections between BusinessObjects server and the Operational Reporting schema.

JDBC configuration is same for both BusinessObjects server and BusinessObjects client tools.

To configure the Oracle JDBC driver on a Windows system:

1. Check to make sure that Oracle client is installed on your BusinessObjects server. If Oracle client is not installed on your BusinessObjects server, then install it.
2. Configure the `tnsnames.ora` file and verify that you can connect to the Operational Reporting database schema from the command line using SQL*Plus.

The `tnsnames.ora` file normally resides in the `<Oracle_Home>\network\admin` directory. For information about how to configure the `tnsnames.ora` file, see the Oracle Technology Network.

3. Navigate to the `<ORACLE_HOME>\jdbc\lib` directory on your BusinessObjects server, and make sure that it contains the `ojdbc5.jar` file.
4. Back up the `jdbc.sbo` file, which is in the `<BusinessObjects_Enterprise_Home>\win32_x86\dataAccess\connectionServer\jdbc` directory.

HP strongly recommends that you back up the `jdbc.sbo` file before you continue to the next step.

5. Open the `jdbc.sbo` file in a text editor, and then locate the following text:

```
<DataBase Active="Yes" Name="Oracle 11">
   <Class JARFile="dbd_jdbc,dbd_oracle"> com.businessobjects
   .connectionserver.java.drivers.jdbc.oracle.OracleDriver
   </Class>
   <JDBCDriver>
```

6. Add the following text under the Oracle 11 `<JDBCDriver>` tag:

```
<ClassPath>
   <Path><Oracle_Home>/jdbc/lib/ojdbc5.jar/ojdbc5.jar</Path>
</ClassPath>
```

7. Save and close the `jdbc.sbo` file.

Importing Universes and Reports

This section provides instructions on how to use the Business Intelligence Archive Resource (BIAR) import tool to import Operational Reporting universes and reports into the BusinessObjects CMS Repository. The BIAR import tool reads the `biar_import.properties` file. It imports all of the universes and reports in the `<Op_Reports_Home>\Universe` and `<Op_Reports_Home>\Reports` directories, respectively.

Requirements for using the BIAR import tool are as follows:

- The JAVA_HOME environment variable must be set (see *Setting the JAVA_HOME Environment Variable* on page 30).
- The `biar_import.properties` file must be configured for your environment.
- The CMS password must be in clear text.

Make sure that you enter the CMS password into the `biar_import.properties` file before you run the BIAR tool, and then remove it from the `biar_import.properties` file after the import is complete.

Configuring the biar_import.properties File

To configure the `biar_import.properties` file:

1. Navigate to the `<Op_Reports_Home>\Deployment\platform\biar` folder on the BusinessObjects Enterprise server.
2. Open the `biar_import.properties` file in a text editor.
3. Replace the default values as shown in the following table.

Default	Description
cms.username=Administrator	BusinessObjects XI Central Management Server (CMS) administrator's username
cms.password=admin123	Password for the CMS administrator **Notes:** The password must be in clear text.
cms.host=localhost	IP address of the BusinessObjects XI CMS machine
cms.port=6400	Port assigned to CMS
bo.home=`\opt\hp\ppm\reporting`	Installation directory for BusinessObjects Enterprise XI

4. Save and close the `biar_import.properties` file.

To import the Operational Reporting universes and reports into the BusinessObjects CMS repository:

1. Navigate to the `<Op_Reports_Home>\Deployment` folder, and then run the `installBIARs.bat` file.
2. Check the `biar_import.log` file (in the `<Op_Reports_Home>\Deployment\platform\biar` folder).
3. Navigate to the `<Op_Reports_Home>\Deployment\platform\biar` folder, open the `biar_import.properties` file in a text editor, and remove the CMS password.

Creating the Operational Reporting Database Schema

To create the Operational Reporting database schema, you run the setup script. To import PPM Center data into the Operational Reporting database, you run the load script. The following sections provide detailed instructions on how to perform each of these tasks.

The setup script does the following:

- Creates the Operational Reporting database schema
- Applies the following grants to the PPM Center database schema for Change Data Capture (CDC) setup:
  - select_catalog_role
  - execute_catalog_role
  - execute on dbms_cdc_publish
  - create job

For Information about Oracle CDC and its use in Operational Reporting, see *Change Data Capture* on page 100.

- Copies several source tables from the PPM Center database schema to the Operational Reporting database schema. The data from these tables are then expanded into dimensions.
- Configures the PPM Center database server for CDC.
- Creates tables, views, and install packages on the Operational Reporting database schema.

Running the Setup Script

To run the setup script:

1. Gather the information listed in the following table.

Information	Description
SYS user name of Reporting DB	SYS user name for the Operational Reporting database Example value: sys
Reporting DB Schema Name	Operational Reporting database schema name Example value: RPT_SCHEMA
Reporting DB data_ tablespace_name	Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS
Reporting DB temp_ tablespace_name	Name of the temp tablespace for the Operational Reporting database Example value: RPT_TEMP_TS
Reporting DB index_tablespace_ name	Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS
Reporting DB TNS Name	Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the `tnsnames.ora` file. Example value: RPT
PPM DB Schema Name	PPM Center database schema name. This value should exist in the Oracle `tnsnames.ora` entry. Example value: PPM_SCHEMA **Important:** The PPM Center database schema name must contain all capital letters. An error occurs if the name contains any lowercase characters.

Information	Description
PPM DB data_ tablespace_name	PPM Center database data tablespace name **Note:** This refers to the existing data tablespace in the *PPM Center database schema*. The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_DATA_TS
PPM DB temp_ tablespace_name	PPM Center database temp tablespace name **Note:** This refers to the existing temp tablespace in the *PPM Center database schema*. The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_TEMP_TS
PPM DB index_ tablespace_name	PPM Center database index tablespace name **Note:** This refers to the existing index tablespace in the *PPM Center database schema*. The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_INDEX_TS
Full tnasame.ora entry to PPM schema	Full `tnasame.ora` entry for the PPM Center database schema • For `HOST`, specify the IP address of the PPM Center database server • For `PORT`, specify the PPM Center database port • FOr `SERVICE_NAME`, specify the SID in `tnsname.ora` file for the PPM Center database Example value: `"(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=16.89.27.63) (PORT=1522)) (CONNECT_DATA= (SERVER=dedicated) (SERVICE_NAME=MDB1106A)))" PPM_DB_LINK`
DB_LINK_NAME to PPM	Name of the link to the PPM Center database This value is generated in the Operational Reporting database schema. Example value: PPM_DB_LINK

Information	Description
SYS user name of PPM DB	SYS user name for the PPM Center database Example value: sys
RPT_DATA_NOLOGGING_TABLESPACE_NAME>	Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: RPT_DATA_TS_NL
RPT_INDEX_NOLOGGING_TABLESPACE_NAME	Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL

2. Stop all PPM Servers (including all nodes in a server cluster).

In the Windows services window, the service name begins with “HP PPM”.

3. Log on to the BusinessObjects server machine, navigate to the `<Op_Report_Home>\DB\install\sample` directory, and open the `sample_setup_all.bat` file in a text editor.
4. Replace each of the variables in the setup script with the corresponding values you prepared for step 1, and then save and close the file.
5. Run the `sample_setup_all.bat` script.
6. During the script run, provide the following information when prompted:
   - PPM Center database server SYS user password
   - PPM Center database server schema password
   - Operational Reporting database server SYS user password
   - Operational Reporting database server schema password
7. The script run creates a log file in the `<Op_Report_Home>\DB\install\log` directory. Check the log file to make sure that no errors occurred.
8. Log on to the BusinessObjects server machine, navigate to the `<PPM_CP1>\DB` directory, and run `resync_ppm.bat`.

9. Provide the following information when prompted:

Prompt	Description
Reporting DB Schema Name	Operational Reporting database schema name Example value: RPT_SCHEMA
Reporting DB TNS Name	Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the `tnsnames.ora` file. Example value: RPT
PPM DB Schema Name	PPM Center database schema name. This value should exist in the Oracle `tnsnames.ora` entry. Example value: PPM_SCHEMA

10. Review the `resync_ppm_<Date_Time>.log` report file (located in the `<Op_Reports_Home>\log` directory).

11. Restart the PPM Servers.

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

Loading PPM Center Data Into the Operational Reporting Database

After you have created the Operational Reporting database schema (*Creating the Operational Reporting Database Schema* on page 34), you can import your PPM Center data into the Operational Reporting database. This section provides information about how to run the load script that brings PPM Center data into the Operational Reporting database schema.

The definition of Materialized View `RPT_DIM_RM_RESOURCES` in Operational Reporting 9.10 (GA) causes big performance problems when loading data. To avoid this problem in Operational Reporting Content Pack 1.2 (CP1.2), between setup and loading data, replace it with the new definition in Operational Reporting CP1.2: `CP1.2\DB\updated_scripts\rpt_dim_rm_resources.sql`.

Running the Load Script

The load script performs a full ETL (extract, transform, and load) to load PPM Center data into the Operational Reporting database schema. The ETL process involves the following:

- Load common dimension tables
- Populate fact tables for the RM Derived Universe, TM Derived Universe, and the FM Derived Universe.
- Post-load configuration (index creation, schema compilation, and so on)

To run the load script, do the following:

1. Gather the information listed in the following table.

Parameter	Description
Reporting DB Schema Name	Operational Reporting database schema name Example value: RPT_SCHEMA
Reporting DB TNS Name	Identifies the Oracle instance running the Operational Reporting database schema. the TNS name is configured in the `tnsnames.ora` file. Example value: RPT

Parameter	Description
Reporting DB index_ tablespace_name	Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS
DB_LINK_NAME to PPM	Name of the link to the PPM Center database. This link is created automatically during the setup_all script run. Example value: PPM_DB_LINK
ETL start date (mm-dd-yyyy)	Start date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Example value: 01-01-2010
ETL end date (mm-dd-yyyy)	End date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. **Note:** The ETL end date you specify is converted based on the fiscal year. For details, see the *Installation and Administration Guide*. Example value: 01-01-2011
Reporting DB data_ tablespace_name	Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS
RPT SYS Username	SYS user name for the Operational Reporting database Example value: sys
Request dimension ETL start date (mm-dd-yyyy)	Start date (in mm-dd-yyyy format) for the PPM Center request data to load into the Operational Reporting database schema. Example value: 01-01-2010 **Note:** If your PPM Center database contains data for old, but active requests, you can include that data without importing all data from that time period.

Parameter	Description
RPT_DATA_NOLOGGING_ TABLESPACE_NAME	Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: PPM_DATA_TS_NL
RPT_INDEX_ NOLOGGING_ TABLESPACE_NAME	Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: PPM_INDEX_TS_NL

2. Log on to the BusinessObjects server machine, navigate to the `<Op_Report_Home>\DB\install\sample` directory, and open the `sample_load_data.bat` file in a text editor.
3. Replace each of the variables in the load script with the corresponding values you prepared for step 1, and then save and close the file.
4. Navigate to the `<Op_Report_Home>\DB\install\sample` directory, and run the `sample_load_data.bat` script.
5. During the load script run, provide Operational Reporting database schema password and the Operational Reporting SYS user password, as prompted.
6. The script creates a `load_data.log` file in the `<Op_Report_Home>\DB\install\log` directory. Check the log file to make sure that no errors occurred.

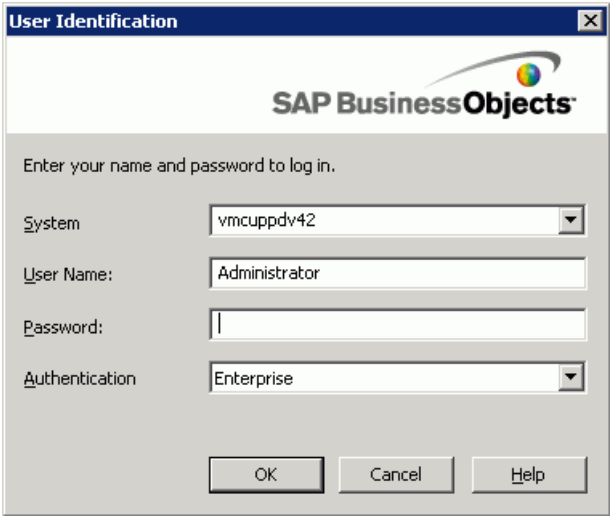
Configuring the Operational Reporting Database Connection

After you import the universes and reports, you must configure the connection to the Operational Reporting database. Before you can configure this connection, make sure that you have completed the following:

- Installed BusinessObjects Enterprise, including SP2 (*Installing BusinessObjects Enterprise on a Windows System* on page 26)
- Configured Oracle 11 JDBC driver (*Configuring the Oracle JDBC Driver* on page 31)
- Imported the universes and reports (*Importing Universes and Reports* on page 32)
- Run the setup script (*Running the Setup Script* on page 35) and load script (*Running the Load Script* on page 39) to setup up the Operational Reporting schema.

To configure the Operational Reporting database connection:

1. To open the Designer logon screen, click **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > Designer.**



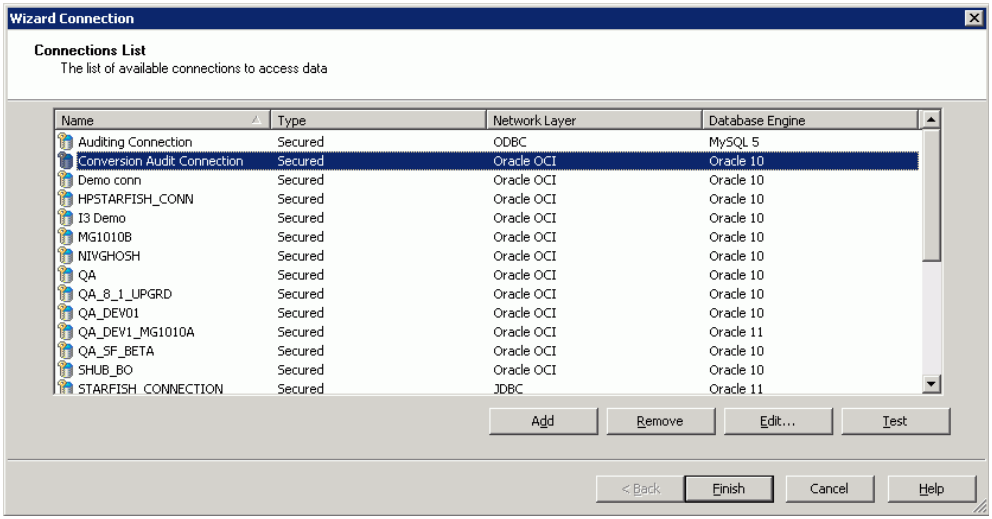
2. In the **User Name** box, type **`Administrator.`**
3. In the **Password** box, type **`admin123.`**

4. If the Welcome to Quick Design screen opens, click **Cancel.**

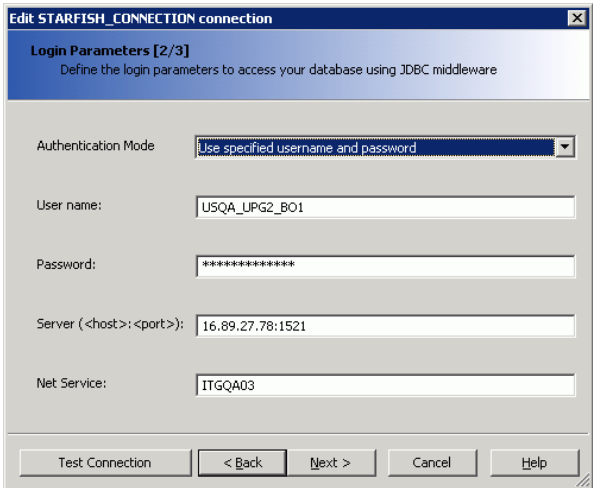
   The Designer starts up.

5. From the **Tools** menu, select **Connections.**

6. In the **Connections** list, select **STARFISH_CONNECTION.**



7. Click **Edit.**



8. Provide the information listed in the following table.

Field	Value
Authentication Mode	Keep the default value (Use specified username and password)
User name	BusinessObjects schema name
Password	BusinessObjects schema password
Server (<host>:<port>)	BusinessObjects database host name and port number (separated by a colon)
Net Service	BusinessObjects database service name

9. Click **Test Connection.**
10. After you see the message “The server is responding,” click **OK.**
11. Finish the process and close the Edit connection window.

Changing the BusinessObjects Central Management Server Password

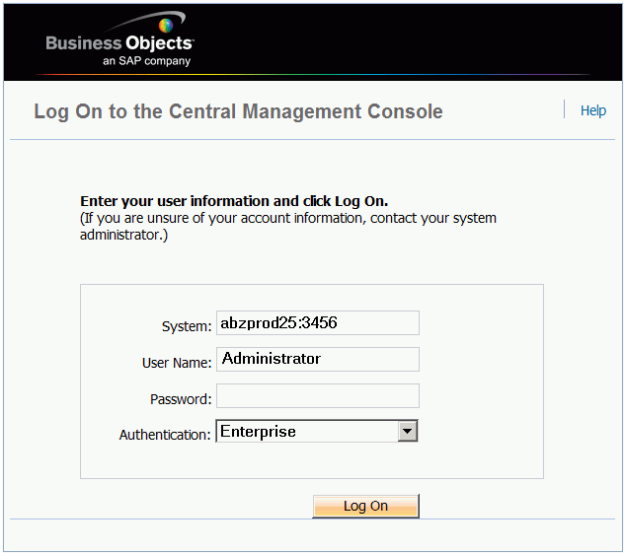
During installation, a default password is used to configure CMS server and deploy HP-provided universes and reports. To prevent unauthorized access after installation, change the default password.

To change the password, do the following:

1. Open a Web browser window and enter the URL for the BusinessObjects Enterprise Central Management Console logon page.

   The default URL is as follows:

   `http://<BusinessObjects_Server_Name>:8080/CmcApp`



2. In the Central Management Console Log On window, log on using the following credentials:
   - In the **User Name** box, type **`Administrator.`**
   - In the **Password** box, type **`admin123.`**
3. Go to the Users management area of the CMC.
4. Click the link for the Administrator account.

5. In the **Enterprise Password Settings** section, type a new password, and then confirm the new password.
6. If the **User must change password at next logon** check box is selected, clear it.
7. Click **Update.**

Verify Successful Operational Reporting Deployment

To verify successful deployment of the Operational Reporting solution, log onto InfoView and generate one of the HP-supplied operational reports. For descriptions of these reports and instructions on how to run them, see the *Operational Reporting User's Guide.*

Configuring Multilingual Support

Although reporting interface elements (control labels, headings, and so on) are displayed only in English, you can configure your BusinessObjects instance to enable users to view operational report contents in a non-English definition language.

The definition language is the language in which a PPM Center entity is defined.The definition language is used as the *fallback* language for PPM Center entities if no translations for those entities are available in PPM Center. For more information, see the *Multilingual User Interface Guide.*

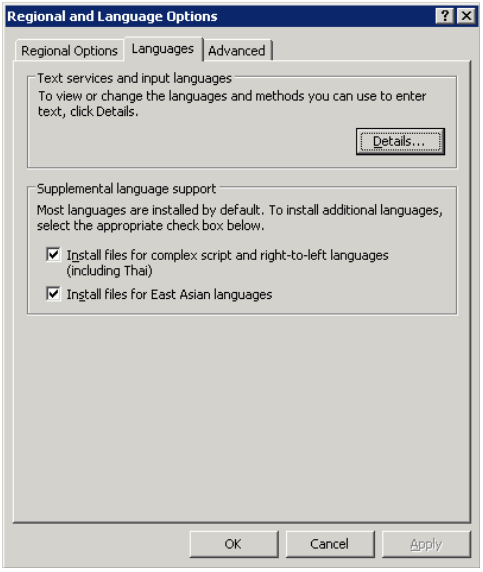
Configuring Multilingual Operational Reporting on a Windows System

This section provides information about how to enable multilingual Operational Reporting on a Windows system. The steps described in the following procedure are for a Windows 2003 system. Depending on your Windows operating system, your steps may differ from those described here.

To enable the display of operational report results on a non-English PPM Center instance:

1. Install the Arial Unicode font on the BusinessObjects server machine.
2. If operational reports are to be accessed from a client installed on a different machine, you must also install the Arial Unicode font on that machine.
3. Open the Control Panel on the BusinessObjects server machine, and then double-click **Regional and Language Options.**

4. Click the **Languages** tab.

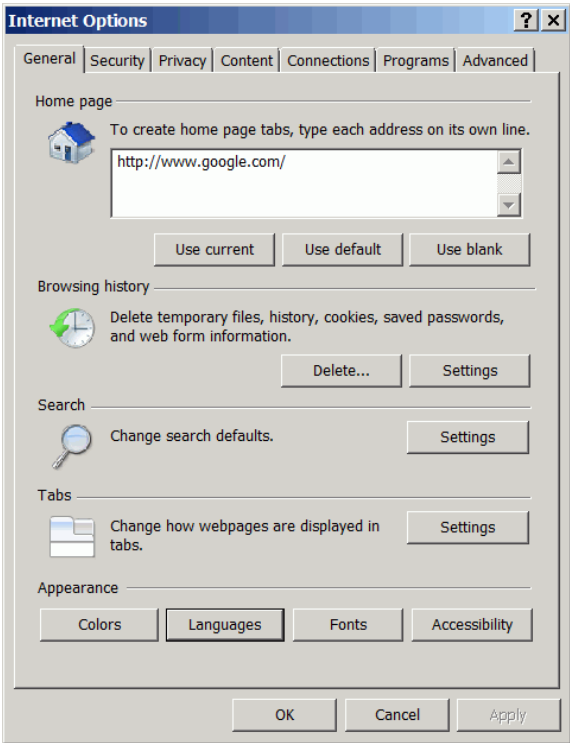


5. In the **Supplemental language support** section, select the check boxes for supplemental language groups to add, and then click **OK.**
6. Restart the BusinessObjects server machine.
7. To update the Oracle NLS_LANG environment variable:
   a. Click **Start > Run,** and then run regedit.
   b. In the Registry Editor, navigate to `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE`, and then set the NLS_LANG variable to `AMERICAN_AMERICA.AL32UTF8`.

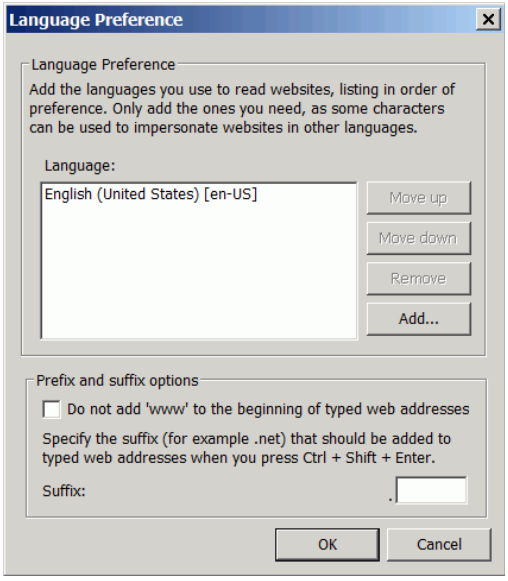
      If you cannot find the NLS_LANG variable in `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE`, add it to the registry manually.

   c. Navigate to `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE/HOME0`, and then set the NLS_LANG variable to `AMERICAN_AMERICA.AL32UTF8`.

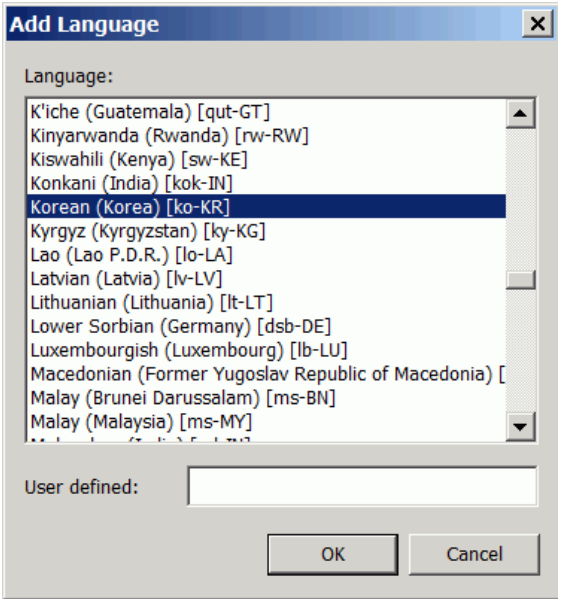
8. On each client machine, do the following:
    a. Open an Internet Explorer browser window.
    b. From the **Tools** menu, select **Internet Options.**



c. In the **Appearance** section, click **Languages.**



d. Click **Add.**



e. In the **Language** box, select one or more languages to add, and then click **OK.**

9. Set Unicode management to UTF-8 encoding, as follows:

   a. Navigate to the `C:\Program Files\Business Objects\BusinessObjects Enterprise 12.0\win32_x86\dataAccess\connectionServer\oracle` directory and open the `oracle.sbo` file in a text editor and locate the `<Defaults>` section.

   b. Replace the lines in the `<Defaults>` section with the following:

```
Parameter Name="Family">Oracle</Parameter>
<Parameter Name="SQL External File">oracle</Parameter>
<Parameter Name="SQL Parameter File">oracle</Parameter>
<Parameter Name="Description File">oracle</Parameter>
<Parameter Name="Strategies File">oracle</Parameter>
<Parameter Name="Driver Level">31</Parameter>
<Parameter Name="Array Fetch Available">True</Parameter>
<Parameter Name="Array Fetch Size">250</Parameter>
<Parameter Name="Array Bind Available">True</Parameter>
<Parameter Name="Array Bind Size">32767</Parameter>
<Parameter Name="Query TimeOut Available">False
   </Parameter>
<Parameter Name="Binary Slice Size">32000</Parameter>
<Parameter Name="CharSet Table">oracle</Parameter>
<Parameter Name="Unicode">UTF8</Parameter>
```

   c. Save and close the `oracle.sbo` file.

   d. On the Oracle server, navigate to the `C:\Program Files\Business Objects\BusinessObjects Enterprise 12.0\win32_x86\dataAccess\connectionServer` directory, open the `cs.cfg` file in a text editor.

   e. Locate the `<DriverDefaults>` section and set the Unicode parameter as follows:

```
<Parameter Name="Unicode">UTF8</Parameter>
```

   f. Save and close the `cs.cfg` file.

10. Modify the `defaultconfig.xml` file to support the Arial Unicode MS font as follows:

    a. Navigate to the `C:\Program Files\Business Objects\Tomcat55\Webapps\AnalyticalReporting\webiapplet\AppletConfig` directory and open the `defaultconfig.xml` file in edit mode.

b. Locate `<CUSTOM_GUI_FONTS VALUE=""/>`, and change it to the following:

```
<CUSTOM_GUI_FONTS VALUE="Arial Unicode MS"/>
```

c. Save and close the `defaultconfig.xml` file.

11. Modify the `fontalias.xml` file to support the Arial Unicode MS font as follows:

a. Navigate to the `C:\Program Files\Business Objects\Business Objects Enterprise 12.0\win32_x86\fonts` directory and open the `fontalias.xml` file in edit mode.

b. Add the following just above the `<FONT NAME="default">` section:

```
<FONT NAME="Arial Unicode">
   <FONTFAMILY PLATFORM="ttf" NAME="'Arial Unicode MS'">
   <FONTATTRIBUTE BOLD="false" ITALIC="false"
   LOGICAL="'Arial Unicode MS'" PHYSICAL="ARIALUNI.ttf"/>
   </FONTFAMILY>
   <FONTFAMILY PLATFORM="win" NAME="'Arial Unicode MS'"/>
   <FONTFAMILY PLATFORM="java" NAME="'Arial
   Unicode MS'"/>
   <FONTFAMILY PLATFORM="html" NAME="'Arial
   Unicode MS'"/>
</FONT>
```

12. Navigate to the `C:\Program Files\Business Objects\Business Objects Enterprise 12.0\win32_x86\scripts` directory, open the `i18n.xml` file in edit mode, and then add the following to the `<font_aliasing> <TTF>` section:

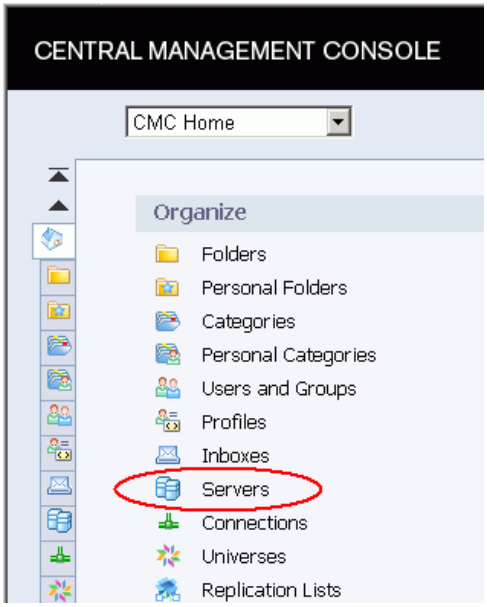
```
</font>
<font name="Arial Unicode MS">
   <os type="all">
      <Attributs style="0" filename="arialuni.ttf"
      encoding=""aliaspsname="Arial Unicode MS"/>
   </os>
</font>
```

13. Select **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > Central Configuration Manager.**

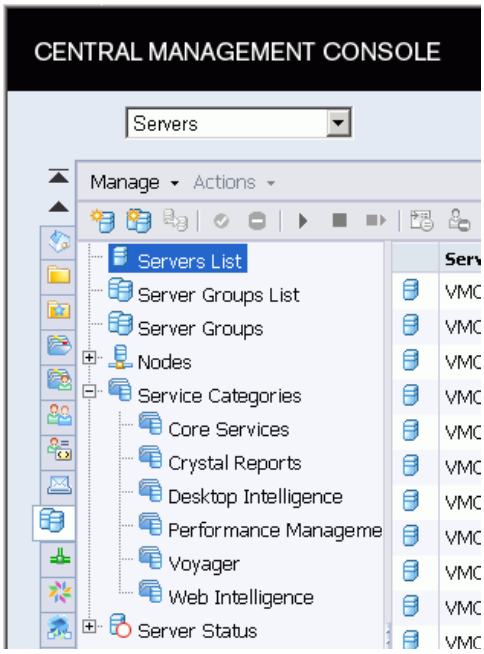
14. Right-click **Apache Tomcat 5.5.2.0,** and then select **Restart** from the shortcut menu.

15. To make sure that all WebI Processing Servers are running:

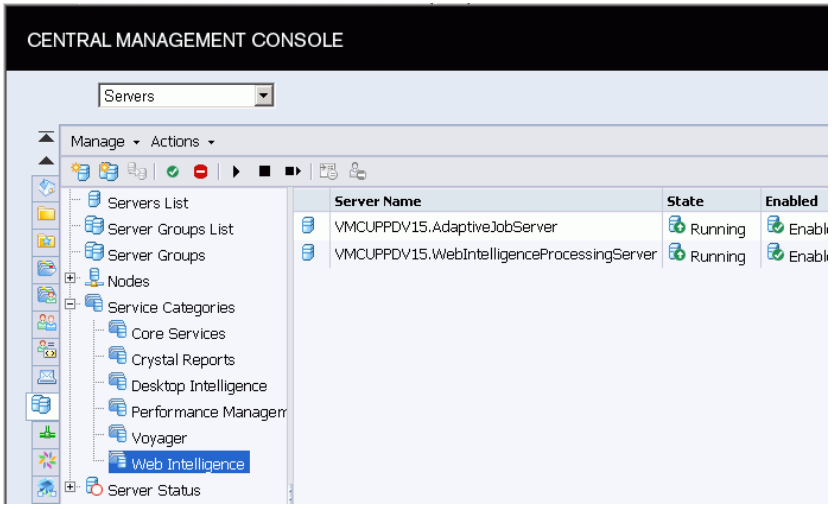
    a. Select **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > BusinessObjects Enterprise Central Management Console.**



b. In the **Organize** column, click **Servers.**



16. In the left pane, expand **Service Categories,** and then click **Web Intelligence.**



17. Check the **State** column to make sure that your Web Intelligence processing servers are running.

18. In the Registry Editor, do the following:

    a. Expand the **HKEY_LOCAL_MACHINE** folder.

    b. Expand the **SOFTWARE** folder.

    c. Expand the **ORACLE** folder.

    d. Open the **KEY_OraClient10g_home1** folder.

    e. Change the **NLS_LANG** value from `AMERICAN_AMERICA.WE8MSWIN1252` to `AMERICAN_AMERICA.AL32UTF8`.

Save the universe, and then export it to the repository.

3 Deploying Operational Reporting on UNIX Systems

Operational Reporting Solution Deployment

This chapter provides the information you need to implement the Operational Reporting solution for PPM Center on a UNIX system. It begins with an overview of the entire process, and then provides detailed instructions for each phase of deployment. For instructions on how to deploy Operational Reporting on a Windows system, see Chapter 2, *Deploying Operational Reporting on Windows Systems,* on page 17.

High-Level Deployment Steps

Deploying the Operational Reporting solution for PPM Center involves the following tasks:

1. Install PPM Center version 9.10, and then upgrade to PPM Center 9.10 service pack 2 (SP2).
2. To make sure that your system meets the requirements for BusinessObjects Enterprise installation, check the *Products Availability Report (PAR)* document, which is available on the BusinessObjects support site (http://support.businessobjects.com/documentation/supported_platforms).
3. (Optional, but strongly recommended for optimal performance) Set up an Oracle database instance specifically for Operational Reporting and set Oracle database parameters. (See *Setting Up a Database for Operational Reporting* on page 60).
4. Check to make sure that the PPM Center database and the Operational Reporting database can communicate over the database link.
5. Create four Oracle tablespaces required to create the Operational Reporting schema and database objects. (See *Creating Tablespaces for the Operational Reporting Schema* on page 61.)
6. Download the Oracle 11g database client software and install it on both your BusinessObjects server and client machine.
7. Set the JAVA_HOME variable on the BusinessObjects server. (See *Setting the JAVA_HOME Environment Variable* on page 71.)
8. Install the SAP BusinessObjects Enterprise software and, optionally, the BusinessObjects Enterprise Client Tools software.
9. Upgrade the BusinessObjects instance with BusinessObjects XI 3.1 Service Pack 2, and, optionally, upgrade the BusinessObjects Enterprise Client Tools software. (See *Installing BusinessObjects Enterprise XI 3.1, Service Pack 2* on page 68.)

10. Verify successful BusinessObjects Enterprise installation and upgrade. (See *Verifying Successful BusinessObjects Enterprise Installation* on page 70.)

11. Set up the Oracle JDBC driver to establish connections between the BusinessObjects server and the Operational Reporting databases. (See *Configuring the Oracle JDBC Driver* on page 71.)

12. Import the PPM Center reporting universes and preconfigured reports. (See *Importing Universes and Reports* on page 72.)

13. Run the setup script to create the Operational Reporting database schema. (See *Creating the Operational Reporting Database Schema* on page 74.)

14. Run the load script to bring PPM Center data into the Operational Reporting database schema. (See *Running the Load Script* on page 79.)

15. Remove the default password for the BusinessObjects Central Management Server (CMS). (See *Removing the BusinessObjects Central Management Server Password* on page 86.)

16. Configure the Operational Reporting database connection. (See *Configuring the Operational Reporting Database Connection* on page 82.) Change the connection parameters for all the universes so that the connection points to the Operational Reporting database schema.

17. To verify successful deployment of Operational Reporting, run the query for an HP-supplied report.

18. (Optional) Configure multilingual support for BusinessObjects Enterprise. (See *Configuring Multilingual Support* on page 87.)

Preparing the Database Schema for Operational Reporting

The following sections provide instructions on how to prepare the Operational Reporting database schema.

Setting Up a Database for Operational Reporting

Requirements and recommendations for setting up the database for Operational Reporting are as follows:

- (Required) Configure the Operational Reporting database to use UTF-8 encoding.
- (Required) Set the Oracle `NLS_CHARACTERSET` parameter to `AL32UTF8`.
- HP strongly recommends that you create an Oracle database specifically for Operational Reporting (independent of your Oracle Database instance). Make sure that the PPM Center database and the Operational Reporting database can communicate over the database link.
- HP strongly recommends that you use the Enterprise Edition of Oracle Database for the Operational Reporting database. The advanced compression and partitioning featured in the Enterprise Edition significantly improve performance, especially if you report on a large and growing volume of data.

Configuring Oracle Database Parameters for Operational Reporting

HP recommends that you use Oracle's automatic memory management (AMM) feature. To do this, set the value for either the `memory_max_target` parameter or the `memory_target` parameter, and then let Oracle manage the memory (SGA and the PGA) dynamically. For more information about how to optimize performance, see the *Deployment Best Practices for PPM Operational Reporting* document.

To obtain the *Deployment Best Practices for PPM Operational Reporting* document and other HP PPM Center documents, go to the Software Product Manuals Web site (support.openview.hp.com/selfsolve/manuals). To access this Web site, you must first set up an HP Passport account.

Creating Tablespaces for the Operational Reporting Schema

Before you can create the database schema for Operational Reporting, you must first create tablespaces (two data and two index tablespaces) for the star schema. This section sections provides instructions for performing this task.

To create the empty database schema (with tables to be populated during installation):

1. Set up the required data and index tablespaces for the Operational Reporting database schema.

For information on the minimum size recommended for these tablespaces, see the *System Requirements and Compatibility Matrix*.

2. Create two tablespaces that include the LOGGING option, as shown in the following examples:

```
CREATE TABLESPACE <PPM_Data>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
LOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;

CREATE TABLESPACE <PPM_Index>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
LOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;
```

3. To improve performance and reduce resource consumption, create two tablespaces that include the NOLOGGING option, as shown in the following examples:

```
CREATE TABLESPACE <PPM_Data_nologging>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
NOLOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;

CREATE TABLESPACE <PPM_Index_nologging>
datafile <'/u0/oracle/oradata/G1010/ppm_data01.dbf'>
size <Size>m
NOLOGGING
DEFAULT COMPRESS
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE
BLOCKSIZE 32K
SEGMENT SPACE MANAGEMENT AUTO
FLASHBACK ON;
```

The Operational Reporting database schema is created automatically during deployment.

Deploying BusinessObjects Enterprise XI 3.1

This section contains information about the operating systems and languages supported by the Operational Reporting solution, instructions on how to prepare your system for BusinessObjects Enterprise installation, and the detailed steps to perform the installation.

Operating Systems Support for BusinessObjects Enterprise

BusinessObjects Enterprise XI 3.1 is supported for Windows, Linux, HP-UX IBM AIX, and Sun Solaris operating systems. For information about the specific versions of the operating systems supported, see the *Products Availability Report (PAR)* document, which is available on the BusinessObjects support site (http://support.businessobjects.com/documentation/supported_platforms).

Preparing to Install BusinessObjects Enterprise XI 3.1 on UNIX

This section addresses the tasks to perform before you start to install BusinessObjects Enterprise.

To prepare your system for BusinessObjects Enterprise installation, do the following:

1. Install all necessary service packs and packages for your operating system.
2. Install PPM Center version 9.10, and then upgrade to PPM Center 9.10 service pack 2 (SP2).

   For information about how to install PPM Center 9.10 and service packs, see the *Installation and Administration Guide* for PPM Center 9.10 or the *Release Notes* for the specific service pack.

3. Check to make sure that your system meets the following minimum disk space requirements for BusinessObjects Enterprise installation:
   - 8.0 GB for BusinessObjects Enterprise (BusinessObjects Server and BusinessObjects Client)
   - 3.0 GB for BusinessObjects Enterprise Client
4. The distribution DVD contains the PPM Center Operational Reporting install bundle, the BusinessObjects Enterprise XI 3.1 install bundle, and the BusinessObjects Enterprise XI 3.1 SP2 Upgrade bundle.

   Extract the entire contents of the distribution DVD to a new folder (hereinafter referred to as the `<Op_Reports_Home>` directory) on the machine that is to host BusinessObjects Enterprise.
5. Set the `LC_ALL` environment variable as follows:

   ```
   export LC_ALL=en_US.utf8
   ```

   To verify that the variable is set correctly, run:

   ```
   env |grep LC_ALL
   ```

6. Create a non-root user account for the directory in which you plan to install the BusinessObjects Enterprise software, as follows:

   `# groupadd <Your_Group_Name>` (`boe` in this case)

   `# useradd -d <User_Home_Path>` (`/home/boe` in this case) `-g <Your_Group_Name>` (`boe` in this case) `<Your_User_Name>` (`boe` in this case)

   ```
   # chown R boe:boe /home/boe
   ```

   ```
   # passwd boe
   ```

7. Update the `/etc/passwd` file so that it points to the directory in which you plan to install BusinessObjects Enterprise.

8. If you plan to install on a Linux system, make sure that:

   - You have write permission for the directory in which you plan to install BusinessObjects Enterprise.
   - The full path name for the directory contains no spaces, and no commas, or other non-alphanumeric characters, except for hyphens (-) or underscores (_). For example, “server1_1” is a valid name, but “server 1,1” is not a valid name.

9. BusinessObjects Enterprise installation and upgrade are memory- and CPU-intensive processes. Shut down all unnecessary processes before you perform the installation (and upgrade).

HP recommends that you have only the Business Object Enterprise installation running.

For more information about the hardware and software requirements for installing and upgrading BusinessObjects Enterprise, see your SAP documentation.

Installing BusinessObjects Enterprise

This section provides instructions on how to install BusinessObjects Enterprise server software.

The BusinessObjects Enterprise IX 3.1 software installation bundle that HP provides includes the Service Pack 2 upgrade. If you already have an instance of BusinessObjects Enterprise IX 3.1 installed, and you plan to use that for Operational Reporting of PPM Center data, you must upgrade to BusinessObjects Enterprise 3.1 Service Pack 2. (See *Installing BusinessObjects Enterprise XI 3.1, Service Pack 2* on page 68.)

Installing BusinessObjects Enterprise on a UNIX System

To install the BusinessObjects Enterprise server software:

1. Navigate to the `<Op_Reports_Home>/Deployment/platform/installer` folder, open the `installer.properties` file in a text editor, and then set the BusinessObjects Enterprise installation parameters, as shown in the following table.

Parameter	Value
boe.unix.install.dir	Installation directory for BusinessObjects Enterprise on UNIX
boe.unix.sianodename	BusinessObjects server intelligence agent node name
boe.unix.cmsnameserver	Host name of the BusinessObjects server
boe.unix.localnameserver	Host name of the BusinessObjects server
boe.unix.upgrade.log	Example: `/opt/boe/Deployment/platform/installer/upgrade.out`
boe.unix.username	Non-root user name (see step 6 on page 65)
boe.unix.response.file	Path to the `unix.ini` file (`<Op_Reports_Home>/Deployment/platform/installer/unix.ini`)

Parameter	Value
boe.unix.cd.dir	BusinessObjects Enterprise installation directory path (<*Op_Reports_Home*>/Deployment/platform/boe31)
boe.unix.dbhostname	Host name or IP address of the machine to host BusinessObjects Enterprise

2. Save and close the `installer.properties` file.
3. Navigate to the <*Op_Reports_Home*>/Deployment directory and run `installReportingServer.sh` file.

   You can monitor the BusinessObjects Reporting Server installation process by viewing the `BOInstall.log` file, which is located in the `var/temp` directory.

On SUSE Linux systems, the log file is located in the `/temp` directory.

The BusinessObjects server is installed in the `C:/hp/ppm/reporting/boe31` directory (hereinafter referred to as <*BO_Home*>). Depending on the resources available to you, installation may take several hours.

4. After you finish installing BusinessObjects Enterprise, do the following:
   - Install BusinessObjects XI 3.1 Service Pack 2. (See *Installing BusinessObjects Enterprise XI 3.1, Service Pack 2* on page 68.)
   - Perform required post-installation tasks. (See *Post-Installation Tasks* on page 70.)

Installing BusinessObjects Enterprise XI 3.1, Service Pack 2

After you have successful installed BusinessObjects XI 3.1, you must install BusinessObjects XI 3.1 Service Pack 2 (SP2). For information about the requirements for installing BusinessObjects XI 3.1 SP2, see the *Products Availability Report (PAR)* document, which you can get from the BusinessObjects support site (http://support.businessobjects.com/documentation/supported_platforms).

To install BusinessObjects XI 3.1 SP2 on UNIX:

1. Before you begin, shut down any processes that are not absolutely required during the upgrade.
2. Navigate to the `<Op_Reports_Home>/Deployment` directory, and then run the `upgradeReportingServer.sh` file.

The upgrade takes a few hours to complete. To monitor the progress of the upgrade, check CPU usage, process (`setup.exe`, `msi*.exe`), disk usage, and the log file.

3. Check the PPM Center *Release Notes* to see whether additional BusinessObjects Enterprise service packs or fix packs are required for Operational Reporting deployment and perform any additional installations required.

Checking the Deployment Log File After BusinessObjects XI 3.1 SP2 Installation

If you install a BusinessObjects Enterprise service pack, the BusinessObjects Web application is automatically re-deployed.

After you install BusinessObjects XI 3.1 SP2, do the following:

1. Navigate to the `<BO_Home>/deployment/workdir/` directory and check the `wdeploy.log` file for any errors that may have occurred.
2. If errors occurred during installation, or if you cannot run a report from InfoView because of JavaScript errors, then manually redeploy BusinessObjects Enterprise as follows:
   a. Back up the `<BO_Home>/deployment/workdir` folder.
   b. Delete all contents of the `<BO_Home>/deployment/workdir` folder.
   c. Change to the `<BO_Home>/deployment` directory, and then run the command `wdeploy.bat tomcat55 deployall`.
3. Check the `wdeploy.log` file again for errors, and then run a report query from InfoView to test the deployment.

Verifying the Upgrade to BusinessObjects XI 3.1 SP2

After installation, go to the `<BO_Home>/setup/logs` directory and check the `BOE_SP2_Install_0.log` file to make sure that the BusinessObjects XI 3.1 SP2 installation was successful.

Next, complete the tasks described in *Post-Installation Tasks* on page 70.

Post-Installation Tasks

This section addresses the following tasks, which must be performed after you install and update BusinessObjects Enterprise:

- *Verifying Successful BusinessObjects Enterprise Installation*
- *Setting the JAVA_HOME Environment Variable*
- *Configuring the Oracle JDBC Driver*
- *Importing Universes and Reports*
- *Creating the Operational Reporting Database Schema*
- *Removing the BusinessObjects Central Management Server Password*
- *Verify Successful Operational Reporting Deployment*
- (Optional) *Configuring Multilingual Support*

Verifying Successful BusinessObjects Enterprise Installation

After you install BusinessObjects Enterprise, check your installation.

To verify that the BusinessObjects Enterprise installation was successful:

- What's the best way to do this?

Setting the JAVA_HOME Environment Variable

Operational Reporting requires that you set the JAVA_HOME variable in the system environment of the user account to be used to start the BusinessObjects server. It is important that the JAVA_HOME environment variable be set for the same shell and user who runs the installation.

On the BusinessObjects server, set JAVA_HOME to:

`<BO_Home>/jdk`

Make sure that the value you specify contains no spaces.

Configuring the Oracle JDBC Driver

Operational Reporting deployment relies on the Oracle JDBC driver to establish connections between BusinessObjects server and the Operational Reporting schema. This section provides instructions for setting up the Oracle JDBC driver on UNIX or Linux systems.

JDBC configuration is same for both BusinessObjects server and BusinessObjects client tools.

Setting up the Oracle JDBC Driver on Unix or Linux

To set up the Oracle JDBC driver on a Linux or UNIX system:

1. Check to make sure that Oracle client is installed on your BusinessObjects server. If Oracle client is not installed, then install it.
2. Configure the `tnsnames.ora` file and verify that you can connect to the Operational Reporting database schema from the command line using SQL*Plus.

The *tnsnames.ora* file normally resides in the `<Oracle_Home>/network/admin` directory.

For information about how to configure the `tnsnames.ora` file, see the Oracle Technology Network.

3. Navigate to the <*ORACLE_HOME*>/jdbc/lib directory on your BusinessObjects server, and make sure that it contains the ojdbc5.jar file.
4. Navigate to the <*Op_Report_Home*>/boe31/bobje/enterprise120/linux_x86/dataAccess/RDBMS/connectionServer/jdbc directory.
5. Back up the jdbc.sbo file.

HP strongly recommends that you backup the jdbc.sbo file before you continue to the next step.

6. Open the jdbc.sbo file in a text editor, and then add the following under the <JDBCDriver> tag:

```
<ClassPath>
   <Path><Oracle_Home>/jdbc/lib/ojdbc5.jar/ojdbc5.jar</Path>
</ClassPath>
```

7. Save and close the jdbc.sbo file.

Importing Universes and Reports

This section provides instructions on how to use the Business Intelligence Archive Resource (BIAR) import tool to import Operational Reporting universes and reports into the BusinessObjects CMS Repository. The BIAR import tool reads the biar_import.properties file. The tool imports all of the universes and reports in the <*Op_Reports_Home*>/Universe and <*Op_Reports_Home*>/Reports directories, respectively.

Requirements for using the BIAR import tool are as follows:

- The JAVA_HOME environment variable must be set (see *Setting the JAVA_HOME Environment Variable* on page 71).
- The biar_import.properties file must be configured for your environment.

Configuring the biar_import.properties File

To configure the `biar_import.properties` file:

1. Navigate to the `<Op_Reports_Home>/Deployment/platform/biar` folder on the BusinessObjects Enterprise server.
2. Open the `biar_import.properties` file in a text editor.
3. Replace the default values as shown in the following table.

Default	Description
cms.username=Administrator	BusinessObjects XI Central Management Server (CMS) administrator's username
cms.password=admin123	Password for the Central Management Server (CMS) administrator. **Note:** You must remove the CMS password from the properties file before you run `upgradeBIARs`.
cms.host=localhost	IP address of the BusinessObjects XI Central Management Server machine.
cms.port=6400	Port assigned to Central Management Server.
bo.home=`/opt/hp/ppm/reporting`	Installation directory for BusinessObjects Enterprise XI.

4. Save and close the `biar_import.properties` file.

To import the Operational Reporting universes and reports into the BusinessObjects CMS repository:

- Navigate to the `<Op_Reports_Home>/Deployment` folder, and then run the `installBIARs.sh` file.

Check the `biar_import.log` file (in the `<Op_Reports_Home>/Deployment/platform/biar` folder).

Creating the Operational Reporting Database Schema

To create the Operational Reporting database schema, you run the setup script. To import PPM Center data into the Operational Reporting database, you run the load script. The following sections provide detailed instructions on how to perform each of these tasks.

This section provides information about how to run the setup script that creates the Operational Reporting database schema.

Running the Setup Script

The setup script does the following:

- Creates the Operational Reporting database schema
- Applies the following grants to the PPM Center database schema for Change Data Capture (CDC) setup:
  - select_catalog_role
  - execute_catalog_role
  - execute on dbms_cdc_publish
  - create job

For Information about Oracle CDC and its use in Operational Reporting, see *Change Data Capture* on page 100.

- Copies several source tables from the PPM Center database schema to the Operational Reporting database schema. The data from these tables are then expanded into dimensions.
- Configures the PPM Center database server for CDC.
- Creates tables, views, and install packages on the Operational Reporting database schema.

To run the setup script:

1. Gather the information listed in the following table.

Information	Description
SYS user name of Reporting DB	SYS user name for the Operational Reporting database Example value: sys
Reporting DB Schema Name	Operational Reporting database schema name Example value: RPT_SCHEMA
Reporting DB data_ tablespace_name	Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS
Reporting DB temp_tablespace_ name	Name of the temp tablespace for the Operational Reporting database Example value: RPT_TEMP_TS
Reporting DB index_tablespace_ name	Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS
Reporting DB TNS Name	Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the `tnsnames.ora` file. Example value: RPT
PPM DB Schema Name	PPM Center database schema name. This value should exist in the Oracle `tnsnames.ora` entry. Example value: PPM_SCHEMA
PPM DB data_ tablespace_name	PPM Center database data tablespace name **Note:** This refers to the existing data tablespace in the *PPM Center database schema*. The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_DATA_TS

Information	Description
PPM DB temp_ tablespace_name	PPM Center database temp tablespace name **Note:** This refers to the existing temp tablespace in the *PPM Center database schema*. The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_TEMP_TS
PPM DB index_ tablespace_name	PPM Center database index tablespace name **Note:** This refers to the existing index tablespace in the *PPM Center database schema*. The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_INDEX_TS
Full tnasame.ora entry to PPM schema	Full `tnasame.ora` entry for the PPM Center database schema • For `HOST`, specify the IP address of the PPM Center database server • For `PORT`, specify the PPM Center database port • FOr `SERVICE_NAME`, specify the SID in `tnsname.ora` file for the PPM Center database Example value: `"(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=16.89.27.63) (PORT=1522)) (CONNECT_DATA= (SERVER=dedicated) (SERVICE_NAME=MDB1106A)))" PPM_DB_LINK`
DB_LINK_NAME to PPM	Name of the link to the PPM Center database This value is generated in the Operational Reporting database schema. Example value: PPM_DB_LINK

Information	Description
SYS user name of PPM DB	SYS user name for the PPM Center database Example value: sys
RPT_DATA_NOLOGGING_TABLESPACE_NAME>	Separate tablespace that requires no redo log for the Operational Reporting database to store data.Example value: PPM_DATA_TS_NL
RPT_INDEX_NOLOGGING_TABLESPACE_NAME	Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL

2. Stop all PPM Servers (including all nodes in a server cluster), as follows:

   If the `REMOTE_ADMIN_REQUIRE_AUTH` parameter is set to `true`, users running `kStop.sh` to shut down the PPM Server must supply a valid PPM Center user name and password. If the parameter is set to `false`, any user with access to the `kStop.sh` script can shut down the server. For information about the `REMOTE_ADMIN_REQUIRE_AUTH` parameter, see the *Installation and Administration Guide.*

   a. Navigate to the `<PPM_Home>/bin` directory.

   b. Run the `kStop.sh` script as follows.

      ```
      sh ./kStop.sh -now -user <User_Name>
      ```

      Make sure that you type a valid user name for a user who has Administrator privileges.

3. Log on to the BusinessObjects server machine, navigate to the `<Op_Report_Home>/DB/install/sample` directory, and open the `sample_setup_all.sh` file in a text editor.

   Make sure that you make the file as an executable.
   For example: `chmod +x sample_setup_all.sh`

4. Replace each of the variables in the setup script with the corresponding values you prepared for step 1, and then save and close the file.

5. Run the `sample_setup_all.sh` script.

6. During the script run, provide the following passwords when prompted:
   - PPM Center database server SYS user password
   - PPM Center database server schema password
   - Operational Reporting database server SYS user password
   - Operational Reporting database server schema password
7. The script run creates a log file in the `<Op_Report_Home>/DB/install/log` directory. Check the log file to make sure that no errors occurred.
8. Log on to the BusinessObjects server machine, navigate to the `<PPM_CP1>/DB` directory, and run `resync_ppm.sh`.
9. Provide the following information when prompted:

Prompt	Description
Reporting DB Schema Name	Operational Reporting database schema name Example value: RPT_SCHEMA
Reporting DB TNS Name	Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the `tnsnames.ora` file. Example value: RPT
PPM DB Schema Name	PPM Center database schema name. This value should exist in the Oracle `tnsnames.ora` entry. Example value: PPM_SCHEMA

10. Review the `resync_ppm_<Date_Time>.log` report file (located in the `<Op_Reports_Home>/log` directory).
11. Restart the PPM Servers.

If your PPM Center instance includes multiple nodes in a cluster configuration, you must start these nodes one at a time. Make sure that you wait until each node is fully started before you start the next node.

Loading PPM Center Data Into the Operational Reporting Database

After you create the Operational Reporting database schema (*Creating the Operational Reporting Database Schema* on page 74), you can import your PPM Center data into the Operational Reporting database. This section provides information about how to run the load script that brings PPM Center data into the Operational Reporting database schema.

The definition of Materialized View `RPT_DIM_RM_RESOURCES` in Operational Reporting 9.10 (GA) causes big performance problems when loading data. To avoid this problem in Operational Reporting Content Pack 1.2 (CP1.2), between setup and loading data, replace it with the new definition in Operational Reporting CP1.2: `CP1.2\DB\updated_scripts\rpt_dim_rm_resources.sql`.

Running the Load Script

The load script performs a full ETL (extract, transform, and load) to load PPM Center data into the Operational Reporting database schema. The ETL process involves the following:

- Load common dimension tables
- Populate fact tables for the RM Derived Universe, TM Derived Universe, and the FM Derived Universe.
- Post-load configuration (index creation, schema compilation, and so on)

To run the load script, do the following:

1. Gather the information listed in the following table.

Parameter	Description
Reporting DB Schema Name	Operational Reporting database schema name Example value: RPT_SCHEMA
Reporting DB TNS Name	Identifies the Oracle instance running the Operational Reporting database schema. the TNS name is configured in the `tnsnames.ora` file. Example value: RPT

Parameter	Description
Reporting DB index_ tablespace_name	Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS
DB_LINK_NAME to PPM	Name of the link to the PPM Center database. This link is created automatically during the setup_all script run. Example value: PPM_DB_LINK
ETL start date (mm-dd-yyyy)	Start date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Example value: 01/01/2009
ETL end date (mm-dd-yyyy)	End date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Example value: 01/01/2011 **Note:** The ETL end date you specify is converted based on the fiscal year. For details, see the *Installation and Administration Guide*.
Reporting DB data_ tablespace_name	Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS
RPT SYS Username	SYS user name for the Operational Reporting database Example value: sys
Request dimension ETL start date (mm-dd-yyyy)	Start date (in mm-dd-yyyy format) for the PPM Center request data to load into the Operational Reporting database schema. Example value: 01/01/2009 **Note:** If your PPM Center database contains data for old, but active requests, you can include that data without importing all data from that time period.

Parameter	Description
RPT_DATA_NOLOGGING_TABLESPACE_NAME	Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: PPM_DATA_TS_NL
RPT_INDEX_NOLOGGING_TABLESPACE_NAME	Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: PPM_INDEX_TS_NL

2. Log on to the BusinessObjects server machine, navigate to the `<Op_Report_Home>/DB/install/sample` directory, and open the `sample_load_data.sh` file in a text editor.

Make sure that you make the file executable.
For example: `chmod +x sample_load_data.sh`

3. Replace each of the variables in the load script with the corresponding values you prepared for step 1, and then save and close the file.
4. Navigate to the `<Op_Report_Home>/DB/install/sample` directory, and run `sample_load_data.sh` script.
5. During the load script run, provide Operational Reporting database schema password and the Operational Reporting SYS user password, as prompted.
6. The script creates a `load_data.log` file in the `<Op_Report_Home>/DB/install/log` directory. Check the log file to make sure that no errors occurred.

Configuring the Operational Reporting Database Connection

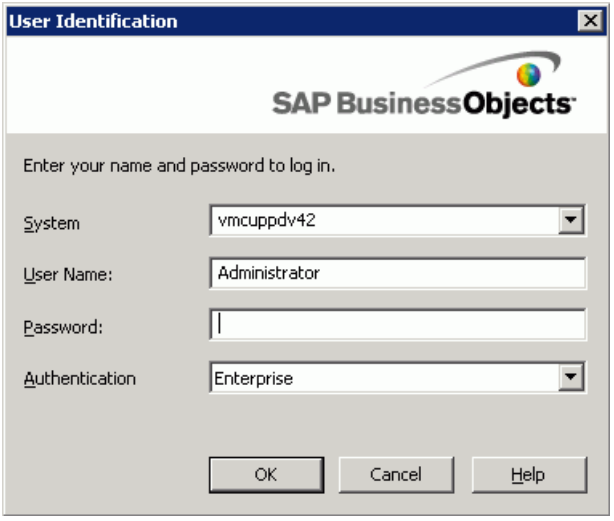
After you import the universes and reports, you must configure the connection to the Operational Reporting database. Before you can configure this connection, make sure that you have completed the following:

- Installed BusinessObjects Enterprise, including SP2 (*Installing BusinessObjects Enterprise* on page 66)
- Configured Oracle 11 JDBC driver (*Configuring the Oracle JDBC Driver* on page 71)
- Imported the universes and reports (*Importing Universes and Reports* on page 72)
- Run the setup script (*Running the Setup Script* on page 74) and load script (*Running the Load Script* on page 79) to setup up the Operational Reporting schema.

To configure the Operational Reporting database connection:

1. To open the Designer logon screen, click **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > Designer.**

If BusinessObjects Enterprise is installed on a Linux, HPUX, AIX, or Solaris system, then you must open BusinessObjects Designer from the Windows client.

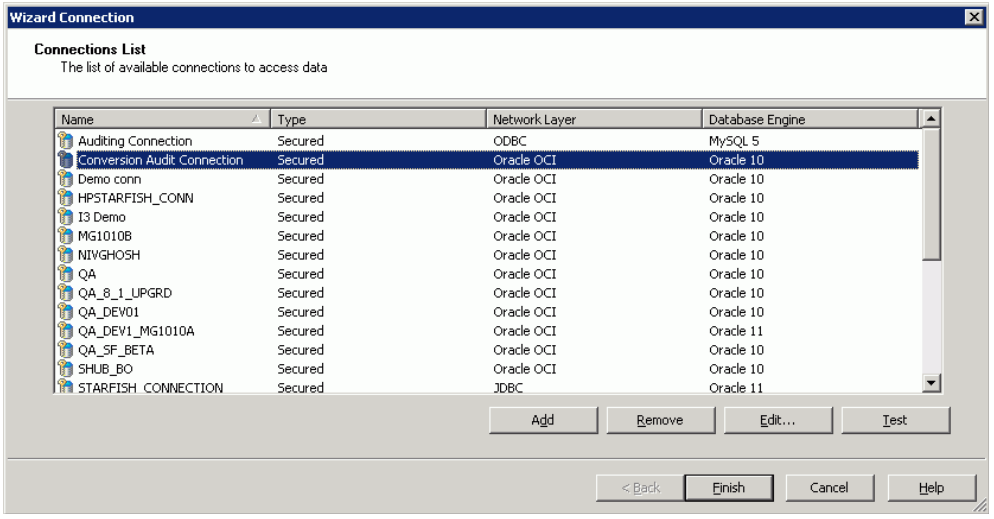


2. In the **User Name** box, type **`Administrator.`**
3. In the **Password** box, type **`admin123.`**
4. If the Welcome to Quick Design screen opens, click **Cancel.**

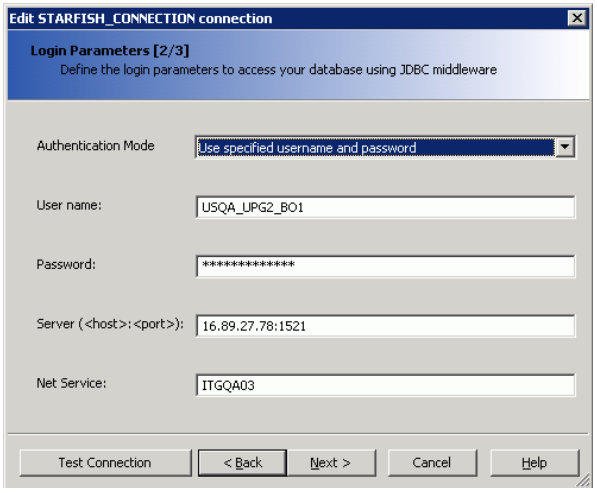
   The Designer starts up.

5. From the **Tools** menu, select **Connections.**

6. In the **Connections** list, select **STARFISH_CONNECTION.**



7. Click **Edit.**



8. Provide the information listed in the following table.

Field	Value
Authentication Mode	Keep the default value (Use specified username and password)
User name	BusinessObjects schema name
Password	BusinessObjects schema password
Server (<host>:<port>)	BusinessObjects database host name and port number (separated by a colon)
Net Service	BusinessObjects database service name

9. Click **Test Connection.**
10. After you see the message “The server is responding,” click **OK.**
11. Finish the process and close the Edit connection window.

Removing the BusinessObjects Central Management Server Password

During installation, a default password is used to configure CMS server and deploy HP-provided universes and reports. On a UNIX system, you must remove the default password.

To remove the password, do the following:

1. Open a Web browser window and enter the URL for the BusinessObjects Enterprise Central Management Console logon page.

   The default URL is as follows:

   `http://<BusinessObjects_Server_Name>:8080/CmcApp`

2. In the Central Management Console Log On window, log on using the following credentials:
   - In the **User Name** box, type **`Administrator.`**
   - Leave the **Password** box empty.
3. Go to the Users management area of the CMC.
4. Click the link for the Administrator account.
5. In the **Enterprise Password Settings** section, delete the default password.

6. If the **User must change password at next logon** check box is selected, clear it.
7. Click **Update.**

Verify Successful Operational Reporting Deployment

To verify successful deployment of the Operational Reporting solution, log onto InfoView and generate one of the HP-supplied operational reports. For descriptions of these reports and instructions on how to run them, see the *Operational Reporting User's Guide.*

Configuring Multilingual Support

Although reporting interface elements (control labels, headings, and so on) are displayed only in English, you can configure your BusinessObjects instance to enable users to view operational report contents in a non-English definition language.

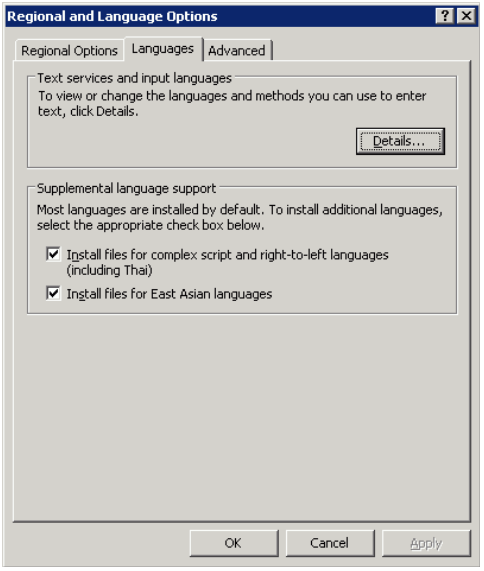
The definition language is the language in which a PPM Center entity is defined.The definition language is used as the *fallback* language for PPM Center entities if no translations for those entities are available in PPM Center. For more information, see the *Multilingual User Interface Guide.*

Configuring Multilingual Operational Reporting on a Windows System

This section provides information about how to enable multilingual Operational Reporting on a Windows system. The steps described in the following procedure are for a Windows 2003 system. Depending on your Windows operating system, your steps may differ from those described here.

To enable the display of operational report results on a non-English PPM Center instance:

1. Install the Arial Unicode font on the BusinessObjects server machine.
2. If operational reports are to be accessed from a client installed on a different machine, you must also install the Arial Unicode font on that machine.
3. Open the Control Panel on the BusinessObjects server machine, and then double-click **Regional and Language Options.**
4. Click the **Languages** tab.



5. In the **Supplemental language support** section, select the check boxes for supplemental language groups to add, and then click **OK.**
6. Restart the BusinessObjects server machine.

7. To update the Oracle NLS_LANG environment variable:

   a. Start the Registry Editor, navigate to `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE`, and then set the NLS_LANG variable to `AMERICAN_AMERICA.AL32UTF8`.

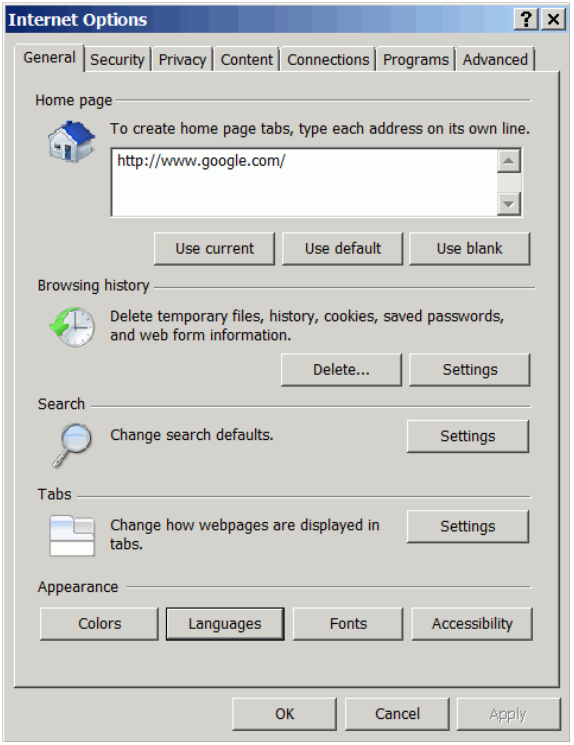
   If you cannot find the NLS_LANG variable in `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE`, add it to the registry manually.

   b. Navigate to `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE/HOME0`, and then set the NLS_LANG variable to `AMERICAN_AMERICA.AL32UTF8`.

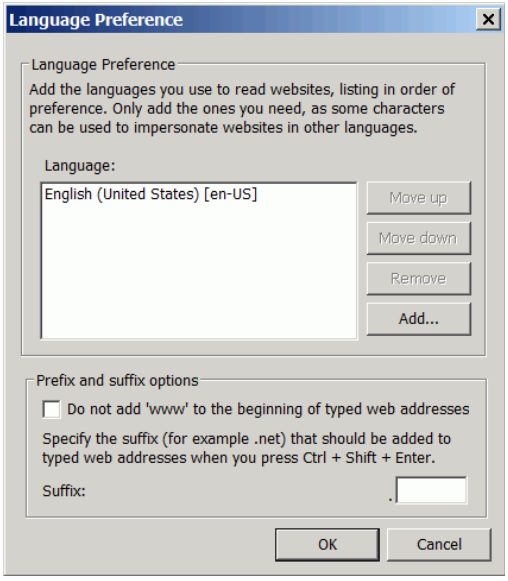
8. On each client machine, do the following:

   a. Open an Internet Explorer browser window.

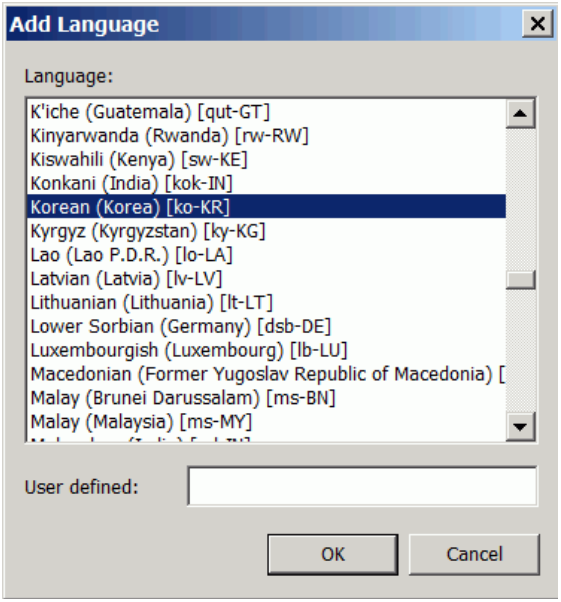
   b. From the **Tools** menu, select **Internet Options.**



c. In the **Appearance** section, click **Languages.**



d. Click **Add.**



e. In the **Language** box, select one or more languages to add, and then click **OK.**

You can use the **Ctrl** key to select multiple languages.

9. Set Unicode management to UTF-8 encoding, as follows:

   a. Navigate to the `C:\Program Files\Business Objects\BusinessObjects Enterprise 12.0\win32_x86\dataAccess\connectionServer\oracle` directory and open the `oracle.sbo` file in a text editor and locate the `<Defaults>` section.

   b. Replace the lines in the `<Defaults>` section with the following:

```
Parameter Name="Family">Oracle</Parameter>
<Parameter Name="SQL External File">oracle</Parameter>
<Parameter Name="SQL Parameter File">oracle</Parameter>
<Parameter Name="Description File">oracle</Parameter>
<Parameter Name="Strategies File">oracle</Parameter>
<Parameter Name="Driver Level">31</Parameter>
<Parameter Name="Array Fetch Available">True</Parameter>
<Parameter Name="Array Fetch Size">250</Parameter>
<Parameter Name="Array Bind Available">True</Parameter>
<Parameter Name="Array Bind Size">32767</Parameter>
<Parameter Name="Query TimeOut Available">False
   </Parameter>
<Parameter Name="Binary Slice Size">32000</Parameter>
<Parameter Name="CharSet Table">oracle</Parameter>
<Parameter Name="Unicode">UTF8</Parameter>
```

   c. Save and close the `oracle.sbo` file.

   d. On the Oracle server, navigate to the `C:\Program Files\Business Objects\BusinessObjects Enterprise 12.0\win32_x86\dataAccess\connectionServer` directory, open the `cs.cfg` file in a text editor.

   e. Locate the `<DriverDefaults>` section and set the Unicode parameter as follows:

```
<Parameter Name="Unicode">UTF8</Parameter>
```

   f. Save and close the `cs.cfg` file.

10. Modify the `defaultconfig.xml` file to support the Arial Unicode MS font as follows:

    a. Navigate to the `C:\Program Files\Business Objects\Tomcat55\Webapps\AnalyticalReporting\webiapplet\AppletConfig` directory and open the `defaultconfig.xml` file in edit mode.

    b. Locate `<CUSTOM_GUI_FONTS VALUE=""/>`, and change it to the following:

    ```
    <CUSTOM_GUI_FONTS VALUE="Arial Unicode MS"/>
    ```

    c. Save and close the `defaultconfig.xml` file.

11. Modify the `fontalias.xml` file to support the Arial Unicode MS font as follows:

    a. Navigate to the `C:\Program Files\Business Objects\Business Objects Enterprise 12.0\win32_x86\fonts` directory and open the `fontalias.xml` file in edit mode.

    b. Add the following just above the `<FONT NAME="default">` section:

    ```
    <FONT NAME="Arial Unicode">
       <FONTFAMILY PLATFORM="ttf" NAME="'Arial Unicode MS'">
       <FONTATTRIBUTE BOLD="false" ITALIC="false"
       LOGICAL="'Arial Unicode MS'" PHYSICAL="ARIALUNI.ttf"/>
       </FONTFAMILY>
       <FONTFAMILY PLATFORM="win" NAME="'Arial Unicode MS'"/>
       <FONTFAMILY PLATFORM="java" NAME="'Arial
       Unicode MS'"/>
       <FONTFAMILY PLATFORM="html" NAME="'Arial
       Unicode MS'"/>
    </FONT>
    ```

12. Navigate to the `C:\Program Files\Business Objects\Business Objects Enterprise 12.0\win32_x86\scripts` directory, open the `i18n.xml` file in edit mode, and then add the following to the `<font_aliasing> <TTF>` section:

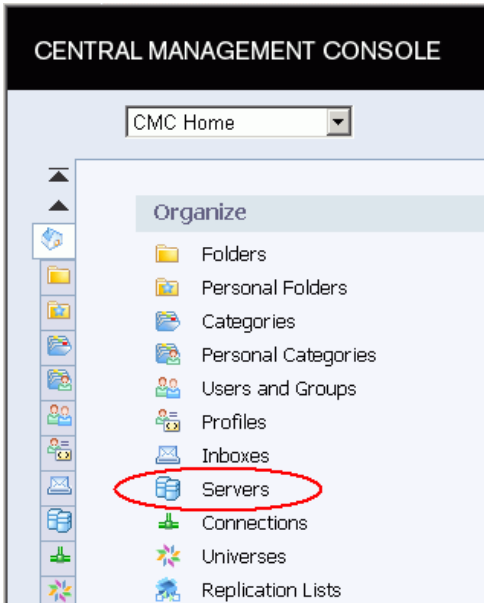
    ```
    </font>
    <font name="Arial Unicode MS">
       <os type="all">
          <Attributs style="0" filename="arialuni.ttf"
          encoding=""aliaspsname="Arial Unicode MS"/>
       </os>
    </font>
    ```

13. Select **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > Central Configuration Manager.**

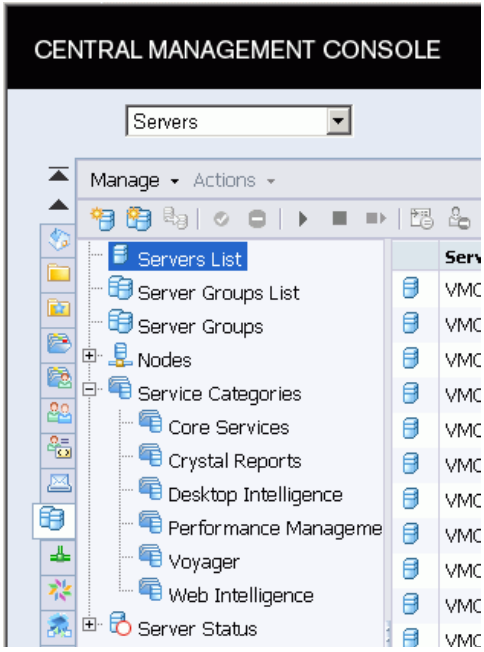
14. Right-click **Apache Tomcat 5.5.2.0,** and then select **Restart** from the shortcut menu.

15. To make sure that all WebI Processing Servers are running:

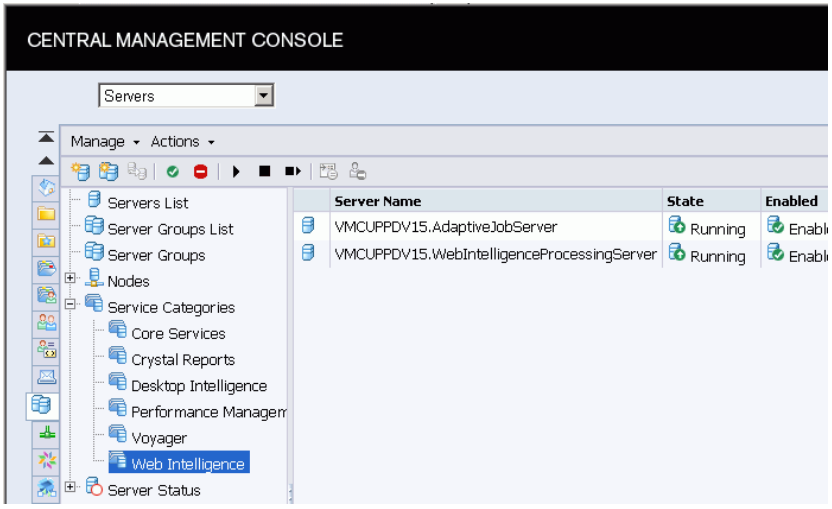
    a. Select **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > BusinessObjects Enterprise Central Management Console.**



b. In the **Organize** column, click **Servers.**



16. In the left pane, expand **Service Categories,** and then click **Web Intelligence.**



17. Check the **State** column to make sure that your Web Intelligence processing servers are running.

18. In the Registry Editor, do the following:

    a. Expand the **HKEY_LOCAL_MACHINE** folder.

    b. Expand the **SOFTWARE** folder.

    c. Expand the **ORACLE** folder.

    d. Open the **KEY_OraClient10g_home1** folder.

    e. Change the **NLS_LANG** value from `AMERICAN_AMERICA.WE8MSWIN1252` to `AMERICAN_AMERICA.AL32UTF8`.

Save the universe, and then export it to the repository.

4 Refreshing Operational Reporting Data

Synchronizing Data in the Operational Reporting and PPM Center Data Schemas

This chapter provides information about how data in the PPM Center database schema and the Operational Reporting database schema are synchronized.

Running Incremental ETL Jobs

The load script that you run during Operational Reporting deployment performs a full ETL to load all PPM Center data into the Operational Reporting database schema. Incremental ETL jobs are scheduled to run automatically every 24 hours thereafter. These incremental ETL jobs cover the day-to-day updates for the PPM Center data tables.

The incremental ETL job that runs automatically every 24 hours is named PPM_ETL_BATCH_JOB. You can use an Oracle commands to reschedule the or change the frequency of the PPM_ETL_BATCH_JOB run. (For information on how to reschedule the PPM_ETL_BATCH_JOB, see the Oracle Database Online Documentation.) This section contains instructions on how to perform incremental ETL jobs manually.

Checking ETL Job Progress

To check the job progress, you can query the `RPT_EVENT_LOG_DETAIL` table, as follows:

```
SELECT   *
FROM   rpt_event_log_detail
ORDER BY   event_time
```

To view the status of an incremental ETL job, you can query the job control tables (RPT_ETL_JOB and RPT_EVENT_LOG_DETAIL tables).

Running Incremental ETL Jobs Manually

To manually run an incremental ETL batch job immediately:

1. Navigate to the `<Op_Report_Home>/DB/install/sample` directory.
2. Open the `sample_onetime_batch.bat` file in a text editor, and then change the script parameters, as shown in the following table.

Parameter	Value
<Reporting DB Schema Name>	Operational Reporting database schema name
<Reporting DB TNS Name>	Operational Reporting database TNS name
<ETL BATCH JOB NAME>	Any job name Example: TM_ETL_DAILY

3. Run the `sample_onetime_batch.bat` (or `sample_onetime_batch.sh`) file.

   Example:

   ```
   call sample_onetime_batch.bat REPORTING SCHEMA ORASID TEST_
   ETL_JOB
   ```

4. When you are prompted, type the password for the Operational Reporting database schema.
5. To check the job progress, you can query the `RPT_EVENT_LOG_DETAIL` table, as follows:

   ```
   SELECT   *
   FROM   rpt_event_log_detail
   ORDER BY   event_time
   ```

To view the status of an incremental ETL job, you can query the job control tables (RPT_ETL_JOB and RPT_EVENT_LOG_DETAIL tables).

If an incremental ETL job fails, it is rerun automatically when the Oracle scheduler starts the job for the next scheduled run, or when you run the job manually, whichever occurs first.

Change Data Capture

Incremental ETL relies on Oracle Change Data Capture, or *CDC,* which is provided as a database server component with your Oracle Database software. CDC identifies and captures data that has been added to, updated, or removed from Oracle relational tables, and makes the change data available for ETL jobs.

Purging Data

CDC uses the DBMS_SCHEDULER package (which runs under the account of the publisher who created the first change table) to create a purge job named `cdc$_default_purge_job`. This purge job calls the `DBMS_CDC_PUBLISH.PURGE` procedure to remove data that subscribers no longer use from the change tables. By default, `cdc$_default_purge_job` runs every 24 hours. You can reschedule the purge job using `DBMS_SCHEDULER.SET_ATTRIBUTE` and setting the `repeat_interval` attribute.

Running the `cdc$_default_purge_job` regularly ensures that the tables do not grow without limit. If you have a large volume of data and need to schedule frequent incremental ETL jobs, you can schedule the `cdc$_default_purge_job` to run more frequently than the default of every 24 hours.

The call to the `DBMS_CDC_PUBLISH.PURGE` procedure evaluates all active subscription windows to determine which change data are still needed. It does not purge any data that could be referenced by one or more subscribers with active subscription windows.

For information about the DBMS_SCHEDULER package, see the *Oracle Database PL/SQL Packages and Types Reference*.

You can also purge the data manually. For detailed information, see the *Oracle Database Data Warehousing Guide* online.

PPM Center Data Transfer During ETL

The following sections describe how PPM Center data are transferred during the ETL process.

Common Dimension Data Transfer

The COMMON ETL job transfers all PPM Center data (that fall within the interval defined by the specified start and end dates) and that are shared by HP Time Management, HP Resource Management, and HP Financial Management into the Operational Reporting database schema.

HP Time Management Data Transfer

A full ETL for HP Time Management (TM) transfers all time-sheet data (that fall within the interval defined by the specified start and end dates) from the HP Time Management tables into the Operational Reporting database schema. A subsequent incremental TM ETL job loads HP Time Management data that have changed since the last TM ETL job run.

HP Resource Management Data Transfer

A full ETL for HP Resource Management (RM) transfers all resource capacity, demand, and actual effort data (that fall within the interval defined by the specified start and end dates) from the HP Resource Management tables into the Operational Reporting database schema. A subsequent incremental RM ETL job loads HP Resource Management data that have changed since the last RM ETL job run.

HP Financial Management Data Transfer

The FM incremental ETL job transfers all HP Financial Management data (that fall within the interval defined by the specified start and end dates) from the HP Financial Management tables into the Operational Reporting database schema. A subsequent incremental FM ETL job loads HP Financial Management data changed since the last FM ETL job run.

Date Range for Transferred Data

The date range for the data moved to the Operational Reporting database during the ETL process is determined by the values you specify for the ETL_START_DATE and ETL_END_DATE parameters. The start date of the fiscal year is determined the year you specify for the ETL_START_DATE value. *Table 4-1* shows how this influences the actual start date for the ETL.

Table 4-1. Effect of PPM Center fiscal year on the calculated ETL start date

Specified ETL_START_DATE	Start Date for the PPM Center Fiscal Year	Calculated (Actual) ETL Start Date
01/15/2008	January 1	01/01/2008
	November 1	11/01/2007

The end date of the fiscal year is determined the year you specify for the ETL_END_DATE value. *Table 4-1* shows how this influences the actual end date for the ETL.

Table 4-2. Effect of PPM Center fiscal year on the calculated ETL end date

Specified ETL_END_DATE	End Date for the PPM Center Fiscal Year	Calculated (Actual) ETL End Date
11/30/2012	December 31	12/31/2012
	October 31	10/31/2013

Date Range for Transferred HP Time Management Data

For HP Time Management data, the value set for the `ETL_START_DATE` parameter determines which time sheets' data are brought into the Operational Reporting database. If a time sheet has an end date that is on or later than the `ETL_START_DATE`, then that time sheet is used to generate data in the Operational Reporting schema.

The `ETL_END_DATE` parameter value is not used. Except for cancelled time sheets, all time sheets with end dates that fall after the ETL start date are brought over.

Date Range for Transferred HP Resource Management Data

Calculated ETL start and end dates affect HP Resource Management data transfer in the following ways:

- Fiscal period definitions are brought over for fiscal periods whose start dates and end dates fall within the time period specified by the calculated start and end dates.
- Resource demand data are brought over for all staffing profiles whose demand falls within the time period defined by the calculated start and end dates.
- Resource capacity data are brought over for all of the resources for the time period between the calculated start and end dates, provided that the resource's end date is later than the calculated end date, and the resource's start date falls within the time period specified by the calculated start and end dates.
- Resource actual effort data are brought over for all the time sheets (excluding cancelled time sheets) with ending dates later than the calculated start date.

If you have long-running projects, keep in mind that requests created before the ETL start date you specify are not brought over, and so the actual effort data for these requests are not available for reporting.

Extending the Time Range of Resource Capacity Data

The Resource Capacity data for resources that do not have an end date are generated based on the ETL start and end dates during the initial load. You can use the extend data script (`sample_extend_data.bat` or `sample_extend_data.sh`) to extend this time interval so that you can compare resource capacity and demand over time. Suppose, for example, that the last full ETL populated the Operational Reporting database with data through 2011. You can use the extend data script to include data for additional years, for example, through the calendar year two years in the future.

The start date for the data loaded using the extend data script is the day after the end year boundary. The end year boundary is based on the end date that you specify and the fiscal calendar's year end. (See *Table 4-1* on page 102 and *Table 4-2* on page 102.)

The extend data script runs the full ETL for capacity and demand for the extended time period and performs the incremental ETL for COMMON, RM, FM, and TM universes. If an incremental ETL job started by the extend script fails, you must run the incremental ETLs again. There is no need to run the extend data script again.

Recommendations for Running the Extend Data Script

To minimize the performance impact of running the extend data script, consider the following:

- Specify a data extension of just one year at a time instead of specifying multiple years.
- When you run `sample_extend_data.bat`, the script first drops all of the bitmap indexes in the HP Resource Management fact tables, and then recreates the indexes after loading the data. HP recommends that you *not* run reports during the extend data script run.

To run the extend data script, do the following:

1. Gather the information listed in the following table.

Variable in the Extend Data Script	Description
<Reporting DB Schema Name>	Operational Reporting database schema name
<Reporting DB TNS Name>	Operational Reporting database TNS name
<Reporting DB index_tablespace_ name>	Name of the index tablespace for the Operational Reporting database
<ETL end date (mm-dd-yyyy)>	End date for the PPM Center data to extract, transform, and load into the Operational Reporting database schema.

2. Log on to the BusinessObjects server machine.
3. Do one of the following:
   - On a Windows system, navigate to the `<Op_Report_Home>/DB/install/sample` directory and open the `sample_extend_data.bat` file in a text editor.
   - On a UNIX system, navigate to the `<Op_Report_Home>\DB\install\sample` directory and open the `sample_extend_data.sh` file in a text editor.
4. Replace each of the variables in the extend data script with the corresponding values you prepared for step 1, and then save and close the file.
5. Depending on your operating system, do one of the following:
   - On a Windows system, run `sample_extend_data.bat`.
   - On a UNIX system, run `sample_extend_data.sh`.
6. During the extend data script run, provide the Operational Reporting database schema password when prompted.
7. The script creates the `extend_data.log` file in the `<Op_Report_Home>/DB/install/log` directory. Log data are also captured in the RPT_EVENT tables. Review the log files and data.

A Troubleshooting

Oracle Trace Log Control for ETL Performance Troubleshooting

This section applies to Operational Reporting for PPM Center Content Pack 1.2 or later.

This functionality is used to analyze ETL performance on Oracle side.

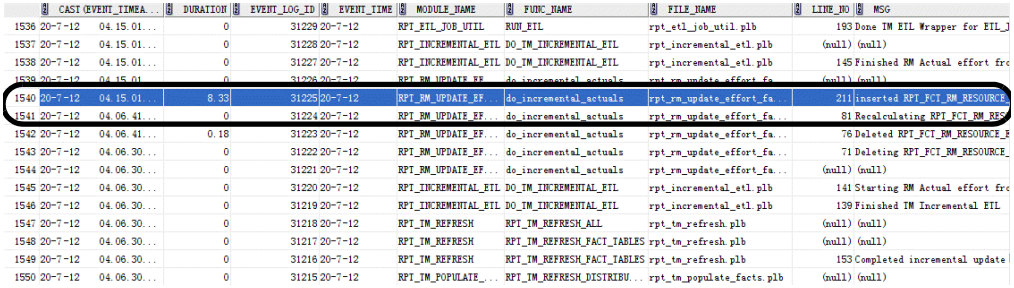
To troubleshoot the ETL performance,

1. Find the SQLs that cost long time from the `rpt_event_log_detail` table.
   a. Run the following SQL:

```
select cast(event_time as timestamp),
round((event_time - lead(event_time,1) over
(order by event_log_id desc))*24*60 ,2) duration, t1.*
from rpt_event_log_detail t1 order by event_log_id desc;
```

   From the DURATION column of the returned results, you can find the SQLs that cost long time.

For example, as shown in the screenshot below, Row 1540 costs 8.33 minutes.



	CAST (EVENT_TIMEA...	DURATION	EVENT_LOG_ID	EVENT_TIME	MODULE_NAME	FUNC_NAME	FILE_NAME	LINE_NO	MSG
1536	20-7-12 04.15.01...	0	31229	20-7-12	RPT_ETL_JOB_UTIL	RUN_ETL	rpt_etl_job_util.plb	193	Done IM ETL Wrapper for ETL_J
1537	20-7-12 04.15.01...	0	31228	20-7-12	RPT_INCREMENTAL_ETL	DO_IM_INCREMENTAL_ETL	rpt_incremental_etl.plb	(null)	(null)
1538	20-7-12 04.15.01...	0	31227	20-7-12	RPT_INCREMENTAL_ETL	DO_IM_INCREMENTAL_ETL	rpt_incremental_etl.plb	145	Finished RM Actual effort fr
1539	20-7-12 04.15.01...	0	31226	20-7-12	RPT_RM_UPDATE_EF...	do_incremental_actuals	rpt_rm_update_effort_fa...	(null)	(null)
1540	20-7-12 04.15.01...	8.33	31225	20-7-12	RPT_RM_UPDATE_EF...	do_incremental_actuals	rpt_rm_update_effort_fa...	211	inserted RPT_FCT_RM_RESOURCE_
1541	20-7-12 04.06.41...	0	31224	20-7-12	RPT_RM_UPDATE_EF...	do_incremental_actuals	rpt_rm_update_effort_fa...	81	Recalculating RPT_FCT_RM_RES
1542	20-7-12 04.06.41...	0.18	31223	20-7-12	RPT_RM_UPDATE_EF...	do_incremental_actuals	rpt_rm_update_effort_fa...	76	Deleted RPT_FCT_RM_RESOURCE_E
1543	20-7-12 04.06.30...	0	31222	20-7-12	RPT_RM_UPDATE_EF...	do_incremental_actuals	rpt_rm_update_effort_fa...	71	Deleting RPT_FCT_RM_RESOURCE_
1544	20-7-12 04.06.30...	0	31221	20-7-12	RPT_RM_UPDATE_EF...	do_incremental_actuals	rpt_rm_update_effort_fa...	(null)	(null)
1545	20-7-12 04.06.30...	0	31220	20-7-12	RPT_INCREMENTAL_ETL	DO_IM_INCREMENTAL_ETL	rpt_incremental_etl.plb	141	Starting RM Actual effort fr
1546	20-7-12 04.06.30...	0	31219	20-7-12	RPT_INCREMENTAL_ETL	DO_IM_INCREMENTAL_ETL	rpt_incremental_etl.plb	139	Finished IM Incremental ETL
1547	20-7-12 04.06.30...	0	31218	20-7-12	RPT_IM_REFRESH	RPT_IM_REFRESH_ALL	rpt_tm_refresh.plb	(null)	(null)
1548	20-7-12 04.06.30...	0	31217	20-7-12	RPT_IM_REFRESH	RPT_IM_REFRESH_FACT_TABLES	rpt_tm_refresh.plb	(null)	(null)
1549	20-7-12 04.06.30...	0	31216	20-7-12	RPT_IM_REFRESH	RPT_IM_REFRESH_FACT_TABLES	rpt_tm_refresh.plb	153	Completed incremental update
1550	20-7-12 04.06.30...	0	31215	20-7-12	RPT_IM_POPULATE_...	RPT_IM_REFRESH_DISTRIBU...	rpt_tm_populate_facts.plb	(null)	(null)

b. Note down the file name, line number, and function name for the pervious rows (start record).

   In this example, you need to note down the information for Row 1541. The file name is `rpt_rm_update_effort_fact.plb`, the line number is `81`, and the function name is `do_incremental_actuals`.

2. Enable trace log for the SQLs you noted down.

   a. Set the parameter `TRACE_LOG_FLAG` in the table `RPT_PARAMS` to `true` by running the following SQL:

      ```
      update RPT_PARAMS set PARAMETER_VALUE='TRUE' where
      PARAMETER_NAME='TRACE_LOG_FLAG';
      ```

   b. Enable trace log for the SQLs you noted down by adding the file name, line number, and function name of the rows to the `RPT_TRACE_DETAILS` table.

      If you add file name and set line number to `-1`, the SQL trace log stays open for the entire package body file (in this example, the `rpt_rm_update_effort_fact.plb` file) .

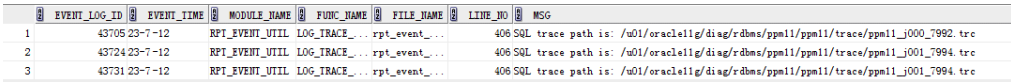
      In this example, you need to set the information as follows:

	FILE_NAME	LINE_NO	FUNC_NAME
1	rpt_rm_update_effort_fact.plb	81	do_incremental_actuals

      The specified SQLs will be traced during the next ETL running process.

3. After the next ETL is completed, find the trace log file path in the table `rpt_event_log_details` by running the following SQL:

   ```
   select * from rpt_event_log_detail where MSG like 'SQL trace
   path is%'
   ```



	EVENT_LOG_ID	EVENT_TIME	MODULE_NAME	FUNC_NAME	FILE_NAME	LINE_NO	MSG
1	43705	23-7-12	RPT_EVENT_UTIL	LOG_TRACE_...	rpt_event_...	406	SQL trace path is: /u01/oracle11g/diag/rdbms/ppm11/ppm11/trace/ppm11_j000_7992.trc
2	43724	23-7-12	RPT_EVENT_UTIL	LOG_TRACE_...	rpt_event_...	406	SQL trace path is: /u01/oracle11g/diag/rdbms/ppm11/ppm11/trace/ppm11_j001_7994.trc
3	43731	23-7-12	RPT_EVENT_UTIL	LOG_TRACE_...	rpt_event_...	406	SQL trace path is: /u01/oracle11g/diag/rdbms/ppm11/ppm11/trace/ppm11_j001_7994.trc

   The SQL trace log file path can be found from the MSG column of the returned results.

4. Log on to the computer where Oracle is installed.

5. Generate a formatted version of the SQL trace log file by running the following command:

   ```
   cd <trace_file_path>
   tkprof  <trace_file_path> <new_log_filename>
   ```

   where *<trace_file_path>* is the SQL trace log file path you get in step 3; *<new_log_filename>* is the file name you specify for the target log file to be generated.

6. Open the new log file and find the SQLs that cost long time for analysis.

7. To close the SQL trace log, empty the table `RPT_TRACE_DETAILS` and disable the parameter `TRACE_LOG_FLAG` by running the following SQL:

   ```
   truncate table RPT_TRACE_DETAILS;
   update RPT_PARAMS set PARAMETER_VALUE='FALSE' where
   PARAMETER_NAME='TRACE_LOG_FLAG';
   ```

Index

E

F

I

J

L

M

O

P

R

S

T

U

W