Installation Guide for Oracle Self-Service E-Billing

Version 6.0.4, Rev. A
October 2011

ORACLE®

Contents

Chapter 4:   Configuring Oracle WebLogic

Chapter 5:   Configuring IBM WebSphere

Chapter 6: Installing the ETL Module for Oracle Self-Service E-Billing

Chapter 7:   Migrating to Oracle Self-Service E-Billing Version 6.0.4

Chapter 8:   Migrating to Oracle Self-Service E-Billing Version 6.0.4 From Other Products

Index

1 What's New in This Release

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.4, Rev. A

Table 1 lists the changes in this version of the documentation to support this release of the software.

Table 1.   What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.4, Rev. A

Topic	Description
"Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156	Modified topic. Updated to include steps for using InstallAnywhere, repackaging the GNU Lesser General Public License, and uninstalling Oracle Self-Service E-Billing version 6.0.3 after migrating from version 6.0.2.
"Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163	Modified topic. Updated to include steps for installing the Oracle Self-Service E-Billing 6.0.4 software, configuring your application server, and installing ETL.
"Roadmap for Installing Oracle Self-Service E-Billing" on page 17 "Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25 "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.0.4" on page 137 "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119	New topics. These topics list the set of processes and tasks required to install and configure Oracle Self-Service E-Billing.

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.4

Table 2 lists the changes in this version of the documentation to support this release of the software.

Table 2.   What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.4

Topic	Description
"Adding Foreign Language Fonts for Localization" on page 23	New topic. It covers how to copy foreign language fonts to your application server directories if you plan to localize Oracle Self-Service E-Billing.
"Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163	New topic. It covers how to migrate from Oracle Self-Service E-Billing 6.0.3 to 6.0.4.
"Configuring JMS Resources for the Command Center on Oracle WebLogic" on page 61 "Configuring the JMS Resources for the Command Center on IBM WebSphere" on page 98	Modified topics. Updated the steps for configuring JMS resources for changes in how batch requests are stored. Batch requests are no longer stored in the JMS queue; they are stored in the Oracle Self-Service E-Billing database.
Configuring JMS Resources for the Billing and Payment Application Configuring a Foreign JMS Server Configuring the JMS Resources Resuming the JMS Resource Configuration for the Billing and Payment Application	Deleted topics. These topics were removed because batch requests are no longer stored in the JMS queue; they are stored in the Oracle Self-Service E-Billing database.
"Replacing the Batch Report Configuration Command" on page 91	New topic. It covers how to replace the batch report configuration command when configuring the Command Center in IBM WebSphere.
"Process of Deploying Oracle Self-Service E-Billing Applications on IBM WebSphere" on page 111	Modified topic. Updated the steps for deploying the Customer Service Representative (CSR) application for the addition of payment functionality.
"Creating the Oracle Warehouse Builder Repository Owner" on page 123	Modified topic. Updated this topic with an additional step when creating the first workspace for the repository in Oracle Warehouse Builder 11.2.0.1 for Windows.

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.3, Rev. A

Table 3 lists the changes in this version of the documentation to support this release of the software.

Table 3.   What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.3, Rev. A

Topic	Description
"Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 18	New topic. It covers how to use a checksum utility to check the integrity of the installer package for Oracle Self-Service E-Billing. This feature was added for compliance with the Payment Card Industry Data Security Standard.
"Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 19	Modified topic. Updated this topic with the new option to install the migration tools with Oracle Self-Service E-Billing.

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.3

Table 4 lists the changes in this version of the documentation to support this release of the software.

Table 4.   What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.3

Topic	Description
"Configuring Log File Paths for Log4j" on page 21	New topic. It covers configuring the file output paths for Log4j.
"Choosing a Database Encryption Method" on page 29 "Process of Implementing Transparent Data Encryption" on page 30 "Process of Implementing Tablespace Encryption" on page 31 "Specifying the Oracle Wallet Location" on page 32 "Setting the Master Encryption Key" on page 36 "Opening the Encrypted Wallet" on page 37	New topics. These topics cover the new functionality for encrypting sensitive data in the Oracle Self-Service E-Billing database.
"Configuring the Oracle Self-Service E-Billing Database on Oracle Exadata" on page 40	New topic. It covers installing the Oracle Self-Service E-Billing database on Oracle Exadata.
"Enabling Oracle Auditing" on page 42	New topic. It covers implementing Oracle auditing, added for compliance with the Payment Card Industry Data Security Standard.

Table 4.   What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.3

Topic	Description
"Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156 "Removing Sensitive Authentication Data from Oracle Self-Service E-Billing 6.0.2" on page 157 "Migrating the OLTP Database from Oracle Self-Service E-Billing 6.0.2 to a New Character Set (Optional)" on page 157 "Migrating Oracle Self-Service E-Billing Version 6.0.2 OLTP to 6.0.3" on page 159 "Migrating Oracle Self-Service E-Billing Version 6.0.2 OLAP to 6.0.3" on page 161 "Compiling the Schema for the Oracle Self-Service E-Billing 6.0.3 OLTP and OLAP Databases" on page 161 "Migrating Batch Reports from Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 162	New topics. These topics cover the processes and procedures for migrating from Oracle Self-Service E-Billing Version 6.0.2 to 6.0.3.
"Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4" on page 171 "Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing 6.0.3" on page 173 "Repackaging the Flat-File Component" on page 175 "Loading Historical Bill Data Using ETL" on page 175 "Configuring the Media Retrieval Functionality" on page 176 "Customizing the Oracle Self-Service E-Billing User Interface to Render the Statement" on page 180 "Customizing the Oracle Self-Service E-Billing User Interface to Render the Statement" on page 180	New topics. These topics cover the processes and procedures for migrating from Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing 6.0.3.
"Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 182 "Migrating Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3" on page 183	New topics. These topics cover the process and procedures for migrating from Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3.

Table 4.   What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.3

Topic	Description
Oracle WebLogic: "Enabling HTTPS on Your Server for the Billing and Payment Application" on page 49 "Enabling HTTPS for the Command Center Server" on page 57 "Enabling HTTPS on Your Server for the Customer Service Representative Application" on page 66 IBM WebSphere: "Enabling HTTPS on Your Server for the Billing and Payment Application" on page 81 "Enabling HTTPS for the Command Center Application Server" on page 95 "Enabling HTTPS on Your Server for the Customer Service Representative Application" on page 107	New topics. These topics cover how to enable HTTPS on your application servers required for compliance with the Payment Card Industry Data Security Standard.
"Modifying the Configuration XMA File for the Billing and Payment Application" on page 87	New topic. This topic covers configuring the configuration properties XMA for the Billing and Payment application.
"Verifying ETL Module System Requirements" on page 119 "Creating the Oracle Warehouse Builder Repository User" on page 125 "Creating File System Locations" on page 129	Modified topics. Updated with steps required for Oracle Database 11g Release 2.
"Installing the ETL Module" on page 131	Modified topic. Updated this topic with steps required for Oracle Database 11g Release 2 and for compliance with the Payment Card Industry Data Security Standard.
"Granting File Location Privileges to MAP_USER for Oracle Database 11g Release 2" on page 134	New topic. It describes how to grant file location privileges for MAP_USER for Oracle Database 11g Release 2.

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.2

Table 5 lists the changes in this version of the documentation to support this release of the software.

Table 5. What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.0.2

Topic	Description
Chapter 2, "Installing Oracle Self-Service E-Billing"	Removed system requirements. This information is available in the *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.
"Creating the Oracle Self-Service E-Billing Database Using Ant" on page 33	Updated the procedures on configuring the Billing and Payment application database for the new build script. Removed the procedures to configure the eStatement and ePayment databases.
"Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 38	Updated the procedures on using the automated ant target to install Oracle Self-Service E-Billing.
Chapter 7, "Migrating to Oracle Self-Service E-Billing Version 6.0.4"	New chapter.
"Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146	Added new procedures for migrating from Oracle Self-Service E-Billing version 6.0.1 to 6.0.2. This includes steps to migrate the payment gateway to PayPal Payflow Pro.
"Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76	Added new procedures to: ■ Set up the Web container filter for the Billing and Payment Application on IBM WebSphere version 6.1.0.3 and higher. ■ Set up log4j output for the Billing and Payment application on IBM WebSphere version 6.1.0.3 and higher.
"Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90	Added a new procedure to set up last participation support in the Command Center.
"Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102	Added new procedures to: ■ Set up the Web container filter for the Customer Service Representative application in IBM WebSphere Version 6.1.0.3 and higher. ■ Set up log4j output for the Customer Service Representative application in IBM WebSphere Version 6.1.0.3 and higher.

2 Installing Oracle Self-Service E-Billing

This chapter covers the tasks you must perform to prepare your platform and install Oracle Self-Service E-Billing. It includes the following topics:

- Roadmap for Installing Oracle Self-Service E-Billing on page 17
- Preparing Your Platform on page 18
- Checking the Integrity of the Oracle Self-Service E-Billing Installer Package on page 18
- Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere on page 19
- Configuring Log File Paths for Log4j on page 21
- Adding Foreign Language Fonts for Localization on page 23
- Uninstalling Oracle Self-Service E-Billing on page 23

Roadmap for Installing Oracle Self-Service E-Billing

This section describes the tasks necessary to install a new implementation of Oracle Self-Service E-Billing, version 6.0.4.

CAUTION: If you are migrating to Oracle Self-Service E-Billing 6.0.4, do not use this roadmap; follow the particular roadmap appropriate for migrating your current product and version. For information about migrating, see "Migrating to Oracle Self-Service E-Billing Version 6.0.4" on page 137 or "Migrating to Oracle Self-Service E-Billing Version 6.0.4 From Other Products" on page 171.

To install a new implementation of Oracle Self-Service E-Billing 6.0.4, perform the following processes and tasks:

1. "Preparing Your Platform" on page 18
2. "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 18
3. "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 19.

   For distributed environments, it is recommended that you install and configure Oracle Self-Service E-Billing in the same top-level directory structure, first on the Oracle Self-Service E-Billing database server, then on the Oracle Self-Service E-Billing application server.

4. "Configuring Log File Paths for Log4j" on page 21
5. "Adding Foreign Language Fonts for Localization" on page 23
6. "Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25
7. Configure your application server:

- ■ **Oracle WebLogic**. Follow "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 45

- ■ **IBM WebSphere.** Follow "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75

8 "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119

Preparing Your Platform

Before installing Oracle Self-Service E-Billing, you must verify that your platform is ready.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

To verify that your platform is ready to install Oracle Self-Service E-Billing

1 Install and test required hardware and software.

For a list of system requirements, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

2 Define required user and group permissions for your Oracle Self-Service E-Billing database and application servers.

For details on configuring your application server, see "Preparing to Configure Oracle WebLogic" on page 46 or "Configuring IBM WebSphere" on page 75.

3 Start and test your Oracle Self-Service E-Billing application server.

For details, see your server documentation.

4 For distributed environments, make sure you have any required Oracle database client software installed on your application server and any other client computers of your Oracle Self-Service E-Billing database server.

Checking the Integrity of the Oracle Self-Service E-Billing Installer Package

After downloading Oracle Self-Service E-Billing from Oracle E-Delivery, you must check the integrity of the installer package using a checksum utility. Run the integrity check before installing Oracle Self-Service E-Billing.

The purpose of the integrity check is to validate that the full package has been delivered and that no data was corrupted during the download of the Oracle Self-Service E-Billing installation files. The checksum utility verifies that the checksum value of the installer package matches the value stored in the checksum.md5 file included with the download. If the values are not the same, either the download was interrupted or data was corrupted. If this is the case, start the download again.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

To check the integrity of the Oracle Self-Service E-Billing installer package

1 Verify that you have a checksum tool, or download a free checksum tool from the Web if necessary:

■ **UNIX.** The md5sum utility is installed by default. You can also download the file from the following GNU Web site:

http://ftp.gnu.org/gnu/coreutils/

■ **Windows.** Download one of a variety of free md5 utilities available for verifying the checksum, such as the following tool, and extract the content of the ZIP file:

http://www.winmd5.com/winmd5free.zip

2 Verify that the binary installer file for Oracle Self-Service E-Billing, eBilling.bin or WinMD5.exe, and the checksum.md5 file are both in the same directory.

3 Run the appropriate command for your operating system:

■ **UNIX.** You can perform the value check automatically or manually. To run the automatic check, use the automatic check option, -c. The result is either valid (OK) or failed, for example:

```
$ md5sum -c /temp/checksum.md5

eBilling.bin: OK
```

To perform the value check manually, run the following command, then manually compare the value generated with the value in the checksum.md5 file:

```
$ md5sum eBilling.bin
```

■ **Windows.** Run the WinMD5.exe command, click Browse, then select the downloaded eBilling.exe file. Copy the checksum value from the downloaded checksum.md5 file, then paste it in the text input area. Click Verify.

Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere

Oracle distributes Oracle Self-Service E-Billing as an InstallAnywhere package. You can change the default installation directory when prompted during the installation procedure.

This document refers to the directory where you install Oracle Self-Service E-Billing as the *EDX_HOME* directory which is, by default:

■ **UNIX.** /opt/Oracle/eBilling

■ **Windows.** Oracle\eBilling

CAUTION: On AIX, the default *EDX_HOME* is: usr/eBilling. Also for AIX, *usr* replaces *opt* wherever used in file paths in this document.

Scripts for creating and configuring the Oracle Self-Service E-Billing database are located in the following directories:

■ **UNIX.** *EDX_HOME/*db

■ **Windows.** *EDX_HOME*db

Web applications you must deploy to your Oracle Self-Service E-Billing application server are located in the following directories:

■ **UNIX.** *EDX_HOME/*J2EEApps

■ **Windows.** *EDX_HOME*J2EEApps

This task is a step in

To install the Oracle Self-Service E-Billing software using InstallAnywhere

1 Start InstallAnywhere in UI mode:

UNIX:

a Log in using the user and group name of the Oracle Self-Service E-Billing application server owner, such as edxadmin:edxadmin.

b Make sure DISPLAY is set, then type the following command:

`./eBilling.bin`

To start InstallAnywhere in Console Mode, type:

`./eBilling.bin –i console`

c Follow the on-screen instructions.

Windows:

❑ Double-click the eBilling.exe file, and follow the on-screen instructions.

2 On the Introduction screen, read the Oracle Self-Service E-Billing introductory information. Click Next to continue.

3 On the License Agreement screen, read the licensing agreement carefully, select the terms acceptance, then click Next.

4 On the Enter Serial Number screen, enter your product serial number, then click Next.

5 (UNIX Only) On the Owner of Web Application Server screen, enter the name of the application server owner (if you have installed other Oracle Self-Service products, use the same owner at this screen that you used for those product installations). Then click Next.

6 (UNIX Only) On the Group of Web Application Server screen, enter the name of the group for the application server. If you have installed other Oracle Self-Service products, use the same group at this screen that you used for those product installations. Then click Next.

7 On the Choose Install Folder screen, accept the default installation folder or click Choose and enter the directory where you want to install the Oracle Self-Service E-Billing files and directories. The directory where you install Oracle Self-Service E-Billing is referred to in this document as *EDX_HOME*. Click Next to continue.

8 On the Choose Product Features screen, choose one of the following:

■ If you are installing Oracle Self-Service E-Billing for the first time, select Option 1, Oracle E-Billing, and click Next.

■ If you are migrating to version 6.0.4, select Option 2, Oracle E-Billing and Migration Tools, and click Next.

For details about valid migration paths, see "Migrating to Oracle Self-Service E-Billing Version 6.0.4" on page 137.

9 (Windows Only) On the Choose Shortcut Folder, choose the Program Group, then click Next.

10 On the Pre-Installation Summary screen, verify that the information is correct, then click Install. To correct any entries, click Previous.

The installer copies the Oracle Self-Service E-Billing software components to the designated installation folder. A status bar on the bottom of the screen shows each component being installed.

If the installation is successful, a congratulatory message appears with the directory that contains the Oracle Self-Service E-Billing components. Click Next.

11 Click Done to exit the installer.

If the installation fails, determine the cause of the problem, and run InstallAnywhere again to reinstall Oracle Self-Service E-Billing.

CAUTION: The installation and configuration examples in this guide use default Oracle Self-Service E-Billing paths, privileges, and permissions. If you choose not to accept the default values, make sure your values are consistent on all servers across your installation.

Configuring Log File Paths for Log4j

By default, Oracle Self-Service E-Billing writes log files to the following directories on the application server:

■ Oracle WebLogic: $WL_HOME/user_projects/domains/domain_name

■ IBM WebSphere: $WS_HOME/profiles/profile_name

It is recommended that you specify the log output paths to the $*EDX_HOME*/logs directory, where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing, as follows:

■ **UNIX.** /opt/Oracle/eBilling/logs

■ **Windows.** D:\Oracle\eBilling\logs

Oracle Self-Service E-Billing consists of three applications, each of which generates appender (-log) files. To change log output paths, you update the XML log configuration files associated with each application.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

To change the output path of log files

■ In each configuration file, located in *EDX_HOME*/config (where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing), edit the output path name in the File parameter for each of the appender log files.

Oracle Self-Service E-Billing Application	Log4j Configuration File	Log Files
Billing and Payment	log4j.xml	■ ebilling-log ■ apache-log ■ root-log ■ reporting-log ■ hierarchy-log ■ umf-log
Command Center	log4j_cc.xml	■ FILE_ESTATEMENT ■ FILE_SCHEDULER ■ FILE_Thirdparty
Customer Service Representative	log4j_csr.xml	■ cba-log ■ csr-log ■ root-log ■ reporting-log ■ hierarchy-log ■ umf-log

For example, the log4j.xml configuration file specifies the default output path for the eBilling-log file as eBilling.log (under the application server's domain_name or profile_name directory):

```
<appender name="ebilling-log" class="org.apache.log4j.RollingFileAppender">

<param name="File" value="eBilling.log"/>

...

</appender>
```

To change the output path for the eBilling-log file, set the File value to /opt/Oracle/eBilling/logs/eBilling.log, as shown:

```
<appender name="ebilling-log" class="org.apache.log4j.RollingFileAppender">

<param name="File" value="/opt/Oracle/eBilling/logs/eBilling.log"/>

...

</appender>
```

Adding Foreign Language Fonts for Localization

If you plan to localize Oracle Self-Service E-Billing, you must copy the foreign language fonts to your application server directories.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

To copy foreign language fonts to your application server

■ Copy all of the TTF files found in *EDX_HOME*/`config/fonts` to the following folders:

■ Oracle WebLogic:

`$WL_HOME/jdk150_11/jre/lib/fonts/`

`$WL_HOME/jrockit_150_11/jre/lib/fonts/`

■ IBM WebSphere:

`$WS_HOME/AppServer/java/jre/lib/fonts/`

Uninstalling Oracle Self-Service E-Billing

You can uninstall and remove Oracle Self-Service E-Billing components and deployed J2EE applications using the Oracle Self-Service E-Billing Uninstaller.

Uninstall Oracle Self-Service E-Billing from the database server first, then uninstall it from the application server.

Note that the Uninstaller does not delete any directories that contain files modified since installation. Instead, it lists these items, which you must then remove manually.

Oracle Self-Service E-Billing does not provide a log for uninstalling; steps to capture the uninstall debug log are included in the following procedure.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

To uninstall Oracle Self-Service E-Billing

1 Stop your application server.

2 Stop your database instance and your database server.

3 Navigate to the Uninstall folder of your Oracle Self-Service E-Billing home directory.

4 On UNIX, direct debug output to either the console or to a file.

■ To direct debug output to the console, enter the following commands:

`export LAX_DEBUG=true`

`- or - setenv LAX_DEBUG true - or - LAX_DEBUG=true set LAX_DEBUG`

■ To direct debug output to a file, enter the following commands:

setenv LAX_DEBUG /export/home/temp/uninstall.log

set LAX_DEBUG=/export/home/temp/uninstall.log

5 Start the Oracle Self-Service E-Billing Uninstaller.

■ **UNIX.** Run the Uninstall_eBilling command, located in the *EDX_HOME*/Uninstall directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

■ **Windows.** Click the Uninstall eBilling.exe file icon; immediately press and hold the Ctrl key to direct the debug log output to the console.

6 When the Uninstall screen appears, click Uninstall.

Oracle Self-Service E-Billing lists the components as it removes them from your computer.

7 When the Uninstaller is finished, a screen appears listing any items that could not be removed.

8 Change the directory to your Oracle Self-Service E-Billing home directory, and manually remove any remaining files and directories as necessary.

9 Click Done to close the Uninstaller.

10 Repeat this procedure on your application server and any other installations.

3 Configuring the Oracle Database

This chapter describes how to configure the Oracle database for Oracle Self-Service E-Billing. It includes the following topics:

■ Roadmap for Configuring the Oracle Self-Service E-Billing Database on page 25

■ Preparing to Configure the Oracle Self-Service E-Billing Database on page 26

■ Configuring Oracle Services on page 26

■ Choosing a Database Encryption Method on page 29

■ Process of Implementing Transparent Data Encryption on page 30

■ Process of Implementing Tablespace Encryption on page 31

■ Specifying the Oracle Wallet Location on page 32

■ Creating the Oracle Self-Service E-Billing Database Using Ant on page 33

■ Setting the Master Encryption Key on page 36

■ Opening the Encrypted Wallet on page 37

■ Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target on page 38

■ Configuring the Oracle Self-Service E-Billing Database on Oracle Exadata on page 40

■ Enabling Oracle Auditing on page 42

Roadmap for Configuring the Oracle Self-Service E-Billing Database

To configure the Oracle Self-Service E-Billing database, perform the following processes and tasks:

1 Preparing to Configure the Oracle Self-Service E-Billing Database on page 26

2 Configuring Oracle Services on page 26

3 Choosing a Database Encryption Method on page 29

4 Process of Implementing Transparent Data Encryption on page 30

5 Process of Implementing Tablespace Encryption on page 31

6 Perform one of following methods of using Ant to create the Oracle Self-Service E-Billing database:

■ Creating the Oracle Self-Service E-Billing Database Using Ant on page 33

■ Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target on page 38

Related Topics
"Roadmap for Installing Oracle Self-Service E-Billing" on page 17

Preparing to Configure the Oracle Self-Service E-Billing Database

Before running the Oracle Self-Service E-Billing database installation Ant script, you must complete the following steps.

This task is a step in "Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25.

To prepare to install the Oracle Self-Service E-Billing database

1 Make sure you have met the minimum system requirements.

For details on system requirements, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

2 Confirm that you have sufficient space on your Oracle Self-Service E-Billing database server. For updated disk space requirements, see 1358365.1 (Article ID) on My Oracle Support.

CAUTION: Insufficient disk space can cause the Oracle Self-Service E-Billing database configuration to fail.

3 Upgrade your Oracle database server software as necessary. For distributed environments, make sure you have any required database client software installed on your application server and any other client computers of your Oracle Self-Service E-Billing database server.

4 Plan passwords with your system administrator, and have syspwd available.

Using Database Clustering

Note that your application server handles database clustering, not Oracle Self-Service E-Billing. For help with clustered installations, contact your Oracle sales representative to request assistance from Oracle's Professional Services.

Configuring Oracle Services

You must edit the following Oracle configuration files that control access to the Oracle Self-Service E-Billing production database:

■ **listener.ora.** Includes a list of service names and address of all listeners on a computer, the instance names of the databases for which they listen, and listener control parameters. The address for a server in the listener.ora file requires the SID (SID_NAME) of a database server in the tnsnames.ora file.

You modify the listener.ora file on the database servers.

■ **tnsnames.ora**. Includes a list of service names of network databases that are mapped to connect descriptors. Clients and distributed database servers use this file to identify potential server destinations. The address of a given database server in the tnsnames.ora file matches the address of a listener for that server in the listener.ora file.

You modify the tnsnames.ora file on the database clients.

By default, these files install in the network administration directory of your Oracle Self-Service E-Billing database server $ORACLE_HOME/network/admin *(%ORACLE_HOME%\network\admin on Windows)*.

Consult with your onsite DBA to configure database connectivity, to make sure you comply with client standards for the enterprise.

For help with database connectivity, create a service request (SR) on My Oracle Support. Alternatively, you can phone Oracle Global Customer Support directly to create a service request or get a status update on your current SR. Support phone numbers are listed on My Oracle Support.

This task is a step in .

To configure Oracle services

1 Change the directory to the network administration directory of your Oracle Self-Service E-Billing database server, for example:

■ **UNIX.** $ORACLE_HOME/network/admin

■ **Windows.** %ORACLE_HOME%\network\admin

2 Open the listener.ora file, and edit the SID_LIST_LISTENER section to reflect your Oracle SID and Oracle Self-Service E-Billing database home directory, for example:

SID_LIST_LISTENER=

(SID_LIST=

(SID_DESC =

 (SID_NAME = OLTP)

 (ORACLE_HOME = /opt/oracle/product/11.2.0.1)

)

(SID_DESC =

 (SID_NAME = OLAP)

 (ORACLE_HOME = /opt/oracle/product/11.2.0.1)

))

3 Save and close the listener.ora file.

4 Change directory to the network administration directory of your Oracle Self-Service E-Billing database client, for example, %ORACLE_HOME%\network\admin.

5 Open the tnsnames.ora file, and edit the database service that identifies your protocol, host, and port.

The following text is an example of a tnsnames.ora file that uses the service names OLTP and OLAP, installed on the database server localhost. Your service names might be different:

```
OLTP =
 (DESCRIPTION =
 (ADDRESS_LIST =
 (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
 )
 (CONNECT_DATA =
 (SERVICE_NAME = OLTP)
 )
 )
OLAP =
 (DESCRIPTION =
 (ADDRESS_LIST =
 (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
 )
 (CONNECT_DATA =
 (SERVICE_NAME = OLAP)
 )
 )
```

6 Save and close the tnsnames.ora file.

7 Repeat Step 5 for the tnsnames.ora file on your application server.

This file installs with your database client software. Single computer environments can skip this step.

8 Stop and restart the Oracle listener with the following listener control commands.

```
> lsnrctl stop
> lsnrctl start
```

9 After the Oracle listener is restarted, to see the service summary for the Oracle Self-Service E-Billing instance, run:

```
> lsnrctl status
```

Services Summary…

OLAP has 1 service handler(s)

OLTP has 1 service handler(s)

This service handler must match the name you entered for the Oracle SID during Oracle Self-Service E-Billing database configuration, in this example, OLTP and OLAP.

Choosing a Database Encryption Method

Oracle Self-Service E-Billing supports Transparent Data Encryption and Tablespace Encryption shipped with the Oracle database. Encryption is an optional feature. If you decide to use encryption, you must choose your security strategy.

Transparent Data Encryption protection ships with Oracle Database 10*g* and Oracle Database 11*g*; Tablespace Encryption is available in Oracle Database 11*g*.

Both Transparent Data Encryption and Tablespace Encryption provide secure storage and management of encryption keys in Oracle Wallet.

Transparent Data Encryption enables encryption of sensitive data in database columns.

Tablespace Encryption is an alternative to Transparent Data Encryption column encryption by enabling encryption of an entire tablespace. You can use Transparent Data Encryption if you only have small amount of data to encrypt, or use Tablespace Encryption to encrypt an entire tablespace, eliminating the need to identify which columns contain sensitive data.

NOTE: It is very important to know how to administer the Oracle Self-Service E-Billing database before choosing and implementing any database encryption strategy. See *Oracle Database Advanced Security Administrator's Guide* on Oracle Technology Network for details.

This task is a step in “Roadmap for Configuring the Oracle Self-Service E-Billing Database” on page 25.

To choose a database encryption method
■ Identify which data is sensitive, and decide which encryption solution is suitable for your organization:

   ■ For Transparent Data Encryption, identify which columns contain sensitive data.

      For details on implementing Transparent Data Encryption, see “Process of Implementing Transparent Data Encryption” on page 30.

   ■ For Tablespace Encryption, identify which tablespaces contain sensitive data.

      For details on implementing Tablespace Encryption, see “Process of Implementing Tablespace Encryption” on page 31. Consider the potential for a reduction in performance of about 9% when database-level encryption is enabled.

      NOTE: There are some restrictions when using Transparent Data Encryption. See *Oracle Database Advanced Security Administrator's Guide* on Oracle Technology Network for details.

Process of Implementing Transparent Data Encryption

Transparent Data Encryption encrypts the columns listed in two CSV files during the creation of the Oracle Self-Service E-Billing database.

By default, columns in the Oracle Self-Service E-Billing OLAP and OLTP databases already known to contain sensitive data are listed in the CSV files:

■ **tde_olap_columns.csv.** Columns in the OLAP instance that contain sensitive data; 0 columns included by default.

■ **tde_oltp_columns.csv.** Columns in the OLTP instance that contain sensitive data; 95 columns included by default.

You can identify additional columns that contain sensitive data and add them to these files.

To implement Transparent Data Encryption, perform the following tasks:

1 In the following CSV files, specify the additional data columns you want to encrypt, if any. These files can be found in the *EDX_HOME*/db/ebilling/oracle/encrypt directory (or the *EDX_HOME*\db\ebilling\oracle\encrypt directory on Windows), where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing:

■ tde_olap_columns.csv

■ tde_oltp_columns.csv

Add the additional columns using the following format:

table_name1, column_name1

table_name1, column_name2

…

table_name2, column_name1

table_name2, column_name2

…

table_nameN, column_name1

table_nameN, column_name2

…

2 Specify the Oracle Wallet Location using the steps in "Specifying the Oracle Wallet Location" on page 32.

3 Follow Step 1 through Step 9 on page 35 in "Creating the Oracle Self-Service E-Billing Database Using Ant" to create the OLAP database instance. In the same procedure, follow Step 12 through Step 15 on page 35 to create the OLTP database instance.

4 Follow the steps in "Setting the Master Encryption Key" on page 36 to set the master encryption key.

5 Follow the steps in *"Opening the Encrypted Wallet" on page 37* to open the encrypted wallet.

6 Run the Ant script again. Select Option 1, Install the OLAP database. Select Option 4, Create OLAP tablespaces. Then select Option 5, Create an OLAP schema, and Option 6, Create OLAP database objects. Return to the Main Menu, and select Option 2, Install the OLTP database. Select Option 4, Install Application Database - Create tablespace and user, and Option 5, Create the OLTP database objects.

7 Complete Step 18 on page 36 to encrypt the data.

8 Complete Step 19 on page 36 through Step 23 on page 36. It is critical to follow these steps to load the seed data and perform the ETL installation process.

This process is a step in *"Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25*.

Process of Implementing Tablespace Encryption

Tablespace Encryption encrypts the tablespaces identified in two SQL files during the creation of the Oracle Self-Service E-Billing database.

By default, Tablespace Encryption is disabled in Oracle Self-Service E-Billing. You must identify the tablespaces you want to encrypt and enable them in two SQL files as described in the following steps to implement Tablespace Encryption.

To set up Tablespace Encryption, perform the following tasks:

1 For each tablespace you want to encrypt, change the encrypted field value from FALSE to TRUE in the corresponding insert statements in the following two files. In the paths, *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing:

■ OLTP database:

  ❏ **UNIX.** *EDX_HOME*/db/eStatement/oracle/estatblspace_list.sql

  ❏ **Windows.** *EDX_HOME*\db\eStatement\oracle\estatblspace_list.sql

■ OLAP database:

  ❏ **UNIX.** *EDX_HOME*/db/ebilling/oracle/olap/setupdatabaseolap.sql

  ❏ **Windows.** *EDX_HOME*\db\ebilling\oracle\olap\setupdatabaseolap.sql

In the following example of an insert statement, EDX_REPORT_IDX is the tablespace name. In the last two field values, FALSE and 3DES168, FALSE means this tablespace will not be encrypted by default, and 3DES168 is the encryption algorithm used for the Tablespace Encryption. Change the encrypted field value from FALSE to TRUE to enable encryption for this tablespace.

```
:insert into "edocs_tablespace" (name, location, tablespace_size, auto_extend,
auto_allocate, uniform_size, encrypted, encrypt_algorithm)

value ('EDX_REPORT_IDX', '$L_DB_EDX_INDX_TB_FILE_LOC/edx_report_idx_01.dbf',
'28M', 'TRUE', 'FALSE', '1m', 'FALSE', '3DES168');
```

2 Specify the Oracle Wallet Location using the steps in "Specifying the Oracle Wallet Location" on page 32.

3 Follow Step 1 through Step 9 on page 35 in "Creating the Oracle Self-Service E-Billing Database Using Ant" to create the OLAP database instance. In the same procedure, follow Step 12 through Step 15 on page 35 to create the OLTP database instance.

4 Follow the steps in "Setting the Master Encryption Key" on page 36 to set the master encryption key.

5 Follow the steps in "Opening the Encrypted Wallet" on page 37 to open the encrypted wallet.

6 Run the Ant script again. Select Option 1, Install the OLAP database. Select Option 4, Create OLAP tablespaces. Then select Option 5, Create an OLAP schema, and Option 6, Create OLAP database objects. Return to the Main Menu, and select Option 2, Install the OLTP database. Select Option 4, Install Application Database - Create tablespace and user and Option 5, Create the OLTP database objects.

7 Complete Step 19 on page 36 through Step 23 on page 36, in "Creating the Oracle Self-Service E-Billing Database Using Ant" on page 33. It is critical to follow these steps to load the seed data and to perform the ETL installation process. For information about installing ETL, see Chapter 6, "Installing the ETL Module for Oracle Self-Service E-Billing."

This process is a step in "Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25.

Specifying the Oracle Wallet Location

To use database encryption, you must specify an Oracle Wallet Location in the sqlnet.ora configuration file before creating the database instance.

After creating the database instance, the security administrator creates a wallet and sets a master key. The external security module stores encryption keys in the Oracle Wallet specified in the sqlnet.ora configuration file.

This task is a step in "Process of Implementing Transparent Data Encryption" on page 30 and in "Process of Implementing Tablespace Encryption" on page 31.

Follow these steps to specify an Oracle Wallet location in the sqlnet.ora configuration file.

To specify the Oracle Wallet location

1 Change to the network administration directory of your Oracle Self-Service E-Billing database server, for example:

■ **UNIX.** $ORACLE_HOME/network/admin

■ **Windows.** %ORACLE_HOME%\network\admin

2 Open the sqlnet.ora file, and use the ENCRYPTION_WALLET_LOCATION parameter to specify the Oracle Wallet Location.

The following example shows a sqlnet.ora file that uses the /export/home/oracle11/wallet directory as the Oracle Wallet Location:

```
ENCRYPTION_WALLET_LOCATION=

    (SOURCE=

            (METHOD=FILE)

            (METHOD_DATA=

                (DIRECTORY=/export/home/oracle11/wallet)

                        )

                    )
```

CAUTION: The wallet location directory must have an absolute path, end with right parentheses, and be an existing directory. Verify that there are no invisible characters at the end of the directory path; this can cause Oracle not to recognize the directory.

3 Save and close the sqlnet.ora file.

4 Specify the same Oracle Wallet location directory in the ebilling_olap.properties file, found in the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows)

Find the ENCRYPTION_WALLET_LOCN parameter in the ebilling_olap.properties file, and set the same value as you set in the sqlnet.ora file:

```
ENCRYPTION_WALLET_LOCN=/export/home/oracle11/wallet
```

5 Save and close the ebilling_olap.properties file.

Creating the Oracle Self-Service E-Billing Database Using Ant

Oracle Self-Service E-Billing provides an ant script for creating the Oracle Self-Service E-Billing database.

You must configure the ebilling_olap.properties and ebilling_oltp.properties files with the same OLAP and OLTP database SID, user name, password and tnsnames to provide a link between the databases. The ebilling_olap.properties, ebilling_oltp.properties, and ebilling_etl.properties files contain configuration parameters specific to each installation and are used by the Ant script that installs the Oracle Self-Service E-Billing database. You update these files before running the E-Billing Ant script.

This task is a step in .

To create the Oracle Self-Service E-Billing database using Ant

1 Specify the following values in the ebilling_olap.properties file for the current installation. This file can be found in the *EDX_HOME*/db/ebilling/oracle directory or the *EDX_HOME*\db\ebilling\oracle directory on Windows, where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing:

■   ORACLE_HOME and ORACLE_ADMIN locations

■   OLAP and OLTP database SID, user name, and password

■   SYSDBA password for OLAP

■   TNSNAMES for OLAP and OLTP database

■   Database file locations for OLAP

■   Redo file locations for OLAP

■   Trace file location for OLAP

2   Specify the following values in the ebilling_oltp.properties file, found in the *EDX_HOME*/db/
ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows) for
the current installation:

■   ORACLE_HOME and ORACLE_BASE locations

■   OLTP and OLAP database SID, user name, and password

■   SYSDBA password for OLTP

■   TNSNAMES for OLAP and OLTP database

■   Database file locations for OLTP

■   Redo file locations for OLTP

■   Trace file location for OLTP

3   Go to the directory of the Oracle Database installation files in your software installation:
EDX_HOME/db/ebilling/oracle (or the *EDX_HOME*\db\ebilling\oracle directory on Windows).

4   If you have not configured the Apache Ant environment, do so now:

■   UNIX:

export ANT_HOME=/opt/apache-ant-1.6.5

export PATH=$ANT_HOME/bin:$PATH

Oracle WebLogic users, run the following command, where *JDK150_11* is your JDK version:

export JAVA_HOME= $WEBLOGIC_HOME/*JDK150_11*

IBM WebSphere users, run the following command:

export JAVA_HOME=$WS_HOME/java

Also, for all application servers on UNIX, run the following command:

export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$PATH

■   Windows:

set ANT_HOME=C:\apache-ant-1.6.5

set PATH=%PATH%;%ANT_HOME%\bin

```
set JAVA_HOME= %WEBLOGIC_HOME%\JDK150_11
```

where *JDK150_11* is your JDK version.

5 Type ant to run the build script. By default, Ant picks up the build.xml file in the current directory.

6 Select Option 1, Install the OLAP database.

If you want to create a schema only on an existing database instance, run Options 4-6 on this menu only. If you want to create a database instance as well, run Options 1-6 or Option 7.

7 Select Option 1, Create an Oracle instance.

This step creates a new database instance for Oracle Self-Service E-Billing reporting, defines a data dictionary, and defines all the system database objects only. No user input is required although several progress messages appear.

NOTE: This step can take anywhere from 20 minutes to 2 hours to complete, depending on the speed of your platform. Ant returns to the current menu when finished.

CAUTION: Review all log files for possible errors.

8 (Optional when creating a new instance and schema) Select Option 2, Shut down the database.

Ant returns to the current menu when finished.

9 Select Option 3, Start up the database.

Ant returns to the current menu when finished.

10 Select Option 4, Create OLAP tablespaces. Then select Option 5, Create an OLAP schema, and Option 6, Create OLAP database objects.

11 Select Option 8, Return to the Main Menu.

12 From the Oracle Self-Service E-Billing Database Installation Menu, select Option 2, Install the OLTP database.

If you want to create a schema on an existing database instance, run Options 4 and 5 on this menu only. If you want to create a database instance as well, run Options 1-5 or Option 6.

13 Select Option 1, Create an Oracle instance.

This step creates a new database instance, defines a data dictionary and stored procedure for the new database, and modifies the stored procedures to contain the absolute paths you defined in the ebilling_oltp.properties file. No user input is required. Several progress messages appear.

NOTE: This step can take anywhere from 20 minutes to 2 hours to complete, depending on the speed of your platform.

CAUTION: Review all log files for possible errors.

14 (Optional when creating a new instance and schema) Select Option 2, Shutdown Database.

Ant returns to the current menu when complete.

15 Select Option 3, Start up Database.

Ant returns to the current menu when complete.

16 Select Option 4, Install Application Database - Create tablespace and user, and Option 5, Create the OLTP database objects.

17 Select Option 7, Return to the Main Menu.

18 (Optional) Select Option 3, Encrypt Sensitive Data. Choose this option if you intend to use Transparent Data Encryption and have followed "Process of Implementing Transparent Data Encryption" on page 30, including specifying an Oracle Wallet location, setting the master encryption key, and opening the encrypted wallet.

   a (Optional) Select Option 1 to run an encryption precheck.

   A precheck reviews the columns listed in the CSV files and reports how many columns can be encrypted and how many cannot. A precheck generates two log files: precheck_olap.log and precheck_oltp.log. Review these two log files for detailed information. These files are found in the $EDX_HOME/db/ebilling/oracle/encrypt directory (or the %EDX_HOME%\db\ebilling\oracle\encrypt directory on Windows).

   b Select Option 4, Encrypt OLAP sensitive data. Review the encrypt_olap.log file, found in the $EDX_HOME/ db/ebilling/oracle/encrypt (or in the %EDX_HOME%\db\ebilling\oracle\encrypt directory on Windows) for detailed information.

   c Select Option 5, Encrypt OLTP sensitive data. Review the encrypt_oltp.log file, found in the $EDX_HOME/db/ ebilling/oracle/encrypt directory (or %EDX_HOME%\db\ebilling\oracle\encrypt directory on Windows) for detailed information.

   d Select Q, Quit.

19 Select Option 4, Load the seed and sample data. Select Option 1, Load the seed data.

This step creates the initial data in some of the seed tables and is required for Oracle Self-Service E-Billing to run properly.

20 (Optional) Select Option 2, Load the sample data.

21 (Optional) If you want a backup, select Option 3, Export the database data.

22 Select Option 4, Return to the Main Menu.

CAUTION: Review all log files for possible errors.

23 Follow the tasks to install the ETL described in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119.

Setting the Master Encryption Key

When implementing database encryption, you must set the master encryption key after creating the database instance using ant.

This task is a step in "Process of Implementing Transparent Data Encryption" on page 30 and in "Process of Implementing Tablespace Encryption" on page 31.

Follow these instructions to set the master encryption key.

To set master encryption key

1 Change to the directory of the Oracle Self-Service E-Billing database installation files, where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing:

   ■ **UNIX.** *EDX_HOME*/db/ebilling/oracle

   ■ **Windows.** *EDX_HOME*\db\ebilling\oracle

2 Type Ant to run the build script. By default, Ant picks up the build.xml file in the current directory.

3 Select Option 3, Encrypt sensitive data.

4 Select Option 2, Set master encryption key. Enter your password twice. The valid password must have a minimum length of eight characters and contain alphabetic characters combined with numbers or special characters.

5 The following message appears:

   Do you want to open wallet automatically when database restart? (Y/N) (Y, y, [N], n)

   Enter Y if you want to keep the wallet open when the database restarts.

   NOTE: Setting the master encryption key is one-time activity. If you enter N for this step, you will have to open the wallet each time you restart the database.

Opening the Encrypted Wallet

The external security module stores the master encryption key in an Oracle wallet. The database must load the master encryption key into memory before it can encrypt or decrypt columns. Opening the wallet allows the database to access the master encryption key.

NOTE: Once the wallet is open, it remains open until you shut down the database instance. When you restart the instance, you must open the wallet again if it is set to manual open mode.

This task is a step in "Process of Implementing Transparent Data Encryption" on page 30 and in "Process of Implementing Tablespace Encryption" on page 31.

Follow these steps to open the encrypted wallet.

To open the encrypted wallet

1 Change to the directory of the Oracle Self-Service E-Billing database installation files, where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing:

   ■ **UNIX.** *EDX_HOME*/db/ebilling/oracle

   ■ **Windows.** *EDX_HOME*\db\ebilling\oracle

2 Type Ant to run the build script.

   By default, Ant picks up the build.xml file in the current directory.

3 Select Option 3, Encrypt sensitive data.

4 Select option 3, Open Oracle Wallet.

NOTE: If you forcibly shutdown an encrypted database (a shutdown abort), the following error message might occur when restarting the database without opening the wallet. This error can occur because during recovery background processes might require access to encrypted redo and undo logs.

ORA-28365: wallet is not open

ORA-28365: wallet is not open

ORA-00600: internal error code, arguments: [kcrp_init_1], [], [], [], [], [],

[], []

The Wallet must be open before opening the database:

SQL> startup mount;

SQL> alter system set wallet open identified by "password";

SQL>alter database open;

Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target

Instead of manually performing each of the Oracle Self-Service E-Billing database setup steps, you can use the automated Ant target to install the Oracle Self-Service E-Billing database.

You must configure the ebilling_olap.properties and ebilling_oltp.properties files with the same OLAP and OLTP database SID, user name, password, and tnsnames to provide a link between the databases. The ebilling_olap.properties, ebilling_oltp.properties, and ebilling_etl.properties files contain configuration parameters specific to each installation and are used by the Ant target that installs the Oracle Self-Service E-Billing database.

CAUTION: The automated ant target will not perform Transparent Data Encryption on your database. For information about choosing an encryption method, see "Choosing a Database Encryption Method" on page 29.

This task is a step in "Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25.

To create the Oracle Self-Service E-Billing database using the automated Ant target

1 Specify the following values in the ebilling_olap.properties file for the current installation. This file can be found in the *EDX_HOME*/db/ebilling/oracle directory or the *EDX_HOME*\db\ebilling\oracle directory on Windows, where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing:

■ ORACLE_HOME and ORACLE_ADMIN locations

■ OLAP and OLTP database SID, user name, and password

- ■ SYSDBA password for OLAP

- ■ Database file locations for OLAP

- ■ Redo file locations for OLAP

- ■ Trace file location for OLAP

2 Specify the following values in the `ebilling_oltp.properties` file, found in the `$EDX_HOME/db/ebilling/oracle` directory (or the `%EDX_HOME%\db\ebilling\oracle` directory on Windows) for the current installation:

- ■ ORACLE_HOME and ORACLE_BASE locations

- ■ OLTP and OLAP database SID, user name, and password

- ■ SYSDBA password for OLTP

- ■ Database file locations for OLTP

- ■ Redo file locations for OLTP

- ■ Trace file location for OLTP

3 Change directory to the location of the Oracle Self-Service E-Billing Database installation files in your software installation: `$EDX_HOME/db/ebilling/oracle` (`%EDX_HOME%\db\ebilling\oracle` on Windows).

4 Run the install-new target to create the new instances, schemas, and schema objects (tables, indexes, packages, procedures, and so on) for OLAP and OLTP, using the SIDs specified in the properties file:

```
ant install-new
```

Note that this command creates seed data for OLTP and OLAP.

Or, if the OLAP and OLTP instances are already created and you want to create schemas on them, use the install-existing target to create schemas and schema objects (tables, indexes, packages, procedures, and so on) with the user names and passwords specified in the properties file. Run:

```
ant install-existing
```

This command creates the E-billing seed data for OLTP and OLAP.

If you want to install sample data for both OLAP and OLTP, add the -DloadSampleData=true argument to the Ant call, for example:

```
ant install-existing -DloadSampleData=true
```

You can optionally run the OLAP-and OLTP-specific Ant build files, buildedxadmin.xml and buildedxolapadmin.xml. You can run any of the targets and flags using these instance-specific files with the -f <filename> flag.

For example, to install a new OLAP instance with sample data only, you would use the following command:

```
ant install-new -f buildedxolapadmin.xml -DloadSampleData=true
```

5 Install the ETL module and the Oracle Warehouse Builder Repository. For instructions on installing the ETL module, see "Installing the ETL Module" on page 131. For instructions on installing and setting up the Oracle Warehouse Builder Repository, see "Process of Installing the Oracle Warehouse Builder Repository" on page 122.

Configuring the Oracle Self-Service E-Billing Database on Oracle Exadata

Configuring Oracle Self-Service E-Billing on Oracle Exadata using Oracle Real Application Clusters (RAC) requires that you manually create database instances using Database Configuration Assistant (DBCA). You must also modify the Ant script before creating the schemas.

You do not need to configure Oracle Services.

NOTE: If you are using Oracle RAC One Node and creating a single instance of a database on Oracle Exadata, you do not need to follow these steps. Use the procedures for UNIX described in this chapter without any changes.

To configure the Oracle Self-Service E-Billing database on Oracle Exadata with Oracle RAC

1 Use the NETCA command to create the listener, using the default value for all options.

2 Create the database instances manually using DBCA. Create one OLAP and one OLTP instance on each node. The following table shows sample data.

SID	Host 1	Host 2
OLTPSID	oltp1	oltp2
OLAPSID	olap1	olap2

NOTE: DBCA automatically creates a TNS name for a new instance, so you do not need to manually configure the tnsnames.ora file.

3 Use SQL*Plus to log on the OLAP instance as sysdba. Run the following scripts individually as the OLAP sysdba user:

$ORACLE_HOME/owb/UnifiedRepos/clean_owbsys.sql

$ORACLE_HOME/owb/UnifiedRepos/cat_owb.sql SYSTEM

$ORACLE_HOME/owb/UnifiedRepos/unlock_owbsys.sql

$ORACLE_HOME/rdbms/admin/userlock.sql

$ORACLE_HOME/rdbms/admin/dbmslock.sql

4 Modify the Ant script for OLAP and OLTP. Edit the following items in both the buildedxolapadmin.xml OLAP file, found in the *db/ebilling/oracle* directory, and the buildedxadmin.xml OLTP file, found in the *db/eStatement/oracle* directory:

■ Remove the following code from the Checkdir target:

```
<available file="${filepath}" type="dir" property="direxists"/>

        <fail unless="direxists"
message="${line.separator}${line.separator}Directory '${filepath}' does not
exists. Create the folder and install again; or try a different path for the
Tablespace"/>
```

■ In the target "SetupDatabase"->replace->replacefilter, remove "${OLAP_SID}/data"
from the following code, and remove the same code for all tokens with _LOC at the end of the
name:

```
token="$L_DB_EDX_DATA_TB_FILE_LOC"
```

```
value="${L_DB_EDX_DATA_TB_FILE_LOC}/${OLAP_SID}/data"/>
```

5 Create the Oracle Self-Service E-Billing database on an existing instance using the ant script with
the following additions and modifications. See either *"Creating the Oracle Self-Service E-Billing
Database Using Ant" on page 33* or *"Creating the Oracle Self-Service E-Billing Database Using the
Automated Ant Target" on page 38.*

■ **Additional Modifications to OLAP Properties.** In the ebilling_olap.properties file, for the
OLAP_SID and OLTP_SID properties set the SID on your software database server, such as
OLAP_SID=olap1 and OLTP_SID=oltp1. Set all TB_FILE_LOC properties to this format:
DISK_GROUP/OLAP_TNS/datafile. For example: L_DB_EDX_DATA_TB_FILE_LOC=+DATA/
olap/datafile. Specify a local directory value for the property TRACE_FILE_LOCN, for
example: TRACE_FILE_LOCN=/export/oracle/olap. Skip the following properties:

❑ DB_CONTROL_FILE_LOCN1

❑ DB_CONTROL_FILE_LOCN2

❑ DB_CONTROL_FILE_LOCN3

❑ REDO_LOG_FILE_LOCN1

❑ REDO_LOG_FILE_LOCN2

❑ SYSTEM_FILE_LOCN

■ **Additional Modifications to OLTP Properties.** In the ebilling_oltp.properties file, for the
OLAP_SID and OLTP_SID properties, set the SID on your software database server, such as
OLAP_SID=olap1, OLTP_SID=oltp1. Set all TB_FILE_LOC properties to this format:
DISK_GROUP/OLTP_TNS/datafile, for example: L_DB_EDX_DATA_TB_FILE_LOC=+DATA/
oltp/datafile. Specify a local directory value for the property TRACE_FILE_LOCN, for
example: TRACE_FILE_LOCN=/export/oracle/oltp. Skip the following properties:

❑ DB_CONTROL_FILE_LOCN1

❑ DB_CONTROL_FILE_LOCN2

❑ DB_CONTROL_FILE_LOCN3

❑ REDO_LOG_FILE_LOCN1

❑ REDO_LOG_FILE_LOCN2

❑ SYSTEM_FILE_LOCN

❑ UNDO_FILE_LOCN

■ **Modifications When Running the Ant Script.** When you run the modified Ant script, use the install-existing option.

6 Follow the procedures in "Installing the ETL Module for Oracle Self-Service E-Billing" on page 119.

Enabling Oracle Auditing

You can enable Oracle auditing. You can audit as an administrator viewing the audit trail live, or run standard auditing to a database.

With standard auditing you can choose which tables to audit. It is recommended to audit the following tables, which contain sensitive data:

■ EDX_BSL_AUTH_SECPROFILE

■ EDX_UMF_SEC_PWD_HISTORY

■ CDA_ATTRIBUTES

■ USR_PASSWORD_ENTRIES

■ EDX_UMF_SEC_VALIDATIONCODE

■ PAYMENT_ACCOUNTS

■ EDX_PMT_CHK_ACCT_ONETIME

To enable standard auditing

1 Open the init.ora initialization file found in the *$ORACLE_HOME/dbs* directory (or the *%ORACLE_HOME%* database directory for Windows) for the Oracle Self-Service E-Billing OLTP database.

2 Set the Oracle static initialization parameter audit_trail to db, extended.

3 Restart the database instance.

4 Using SQL*Plus, connect to the OLTP database as the OLTP schema user.

5 For each table you want to audit, run the following command, where *Tablename* is the database tablename:

audit all on *Tablename* by access;

6 Query the system view DBA_AUDIT_TRAIL to view the audit records.

To enable auditing by a system administrator

1 Open the init.ora initialization file found in the *$ORACLE_HOME/dbs* directory (or the *%ORACLE_HOME%* database directory for Windows) for the Oracle Self-Service E-Billing OLTP database.

2 Set the Oracle static initialization parameter audit_sys_operations to true.

3 Restart the database instance.

4 View the audit trail using the Windows Event Viewer or syslog on UNIX.

4 Configuring Oracle WebLogic

This chapter describes how to configure Oracle WebLogic for Oracle Self-Service E-Billing. It includes the following topics:

- Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing on page 45
- Preparing to Configure Oracle WebLogic on page 46
- Process of Configuring Oracle WebLogic for the Billing and Payment Application on page 46
- Process of Configuring Oracle WebLogic for the Command Center Application on page 55
- Process of Configuring Oracle WebLogic for the Customer Service Representative Application on page 63
- Process of Repackaging the GNU Lesser General Public License on page 67
- Deploying Oracle Self-Service E-Billing Applications on Oracle WebLogic on page 70
- Specify the appropriate details for the database server, Oracle home directory, and application server. on page 72
- Running the Sample Oracle Self-Service E-Billing Applications on Oracle WebLogic on page 73

Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing

To configure Oracle WebLogic for Oracle Self-Service E-Billing, perform the following processes and tasks:

- "Preparing to Configure Oracle WebLogic" on page 46
- "Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 46
- "Process of Configuring Oracle WebLogic for the Command Center Application" on page 55
- "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 63
- "Process of Repackaging the GNU Lesser General Public License" on page 67
- "Deploying Oracle Self-Service E-Billing Applications on Oracle WebLogic" on page 70
- "Configuring and Starting Scheduler on Oracle WebLogic" on page 71
- "Running the Sample Oracle Self-Service E-Billing Applications on Oracle WebLogic" on page 73

This roadmap is part of "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

Preparing to Configure Oracle WebLogic

Before configuring Oracle WebLogic, you must complete steps described in this topic.

This task is a step in .

To prepare to configure Oracle WebLogic

1 Verify that the database server components are installed and configured for Oracle Self-Service E-Billing.

2 For distributed environments, verify that you have any required database client software installed on the Oracle WebLogic application server and any other client computers of your database server.

3 Start the Oracle WebLogic Administration Console. If you cannot start the Administration Console, you will be unable to proceed with configuring your application server for Oracle Self-Service E-Billing.

The instructions to configure Oracle WebLogic assume in-depth understanding of and practical experience with application server administration. Consult the Oracle WebLogic documentation as necessary.

NOTE: The installation and configuration examples in this guide use default Oracle Self-Service E-Billing paths, privileges and permissions. If you choose not to accept the default values, make sure your values are consistent on all servers across your installation of Oracle Self-Service E-Billing.

Process of Configuring Oracle WebLogic for the Billing and Payment Application

To configure Oracle WebLogic for the Billing and Payment application, perform the following tasks:

1

2

3

4

5

6

This process is a step in .

Creating the Oracle WebLogic Domain for the Billing and Payment Application

You must create an Oracle WebLogic domain for the Billing and Payment application EAR file.

This task is a step in "Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 46.

To create an Oracle WebLogic domain for the Billing and Payment application EAR file

1   Run the following command from the $WL_HOME/common/bin directory (or the
    %WL_HOME%\common\bin directory on Windows):

    ■   **UNIX.** config.sh

    ■   **Windows.** config.cmd

2   On the Oracle WebLogic Configuration Wizard select Create a new WebLogic domain, then click
    Next.

3   Leave the option to automatically configure the domain selected, then click Next.

4   Enter the user name and password of the user to administer the Oracle WebLogic domain, then
    click Next.

5   Select the SUNJDK to use for this domain, then click Next.

6   Choose Yes to configure the Oracle WebLogic domain, then click Next.

7   Enter the name of the domain to create, such as ebilling_domain, then enter a location for the
    domain, such as port 7001 and default Admin Server. Click Create.

Defining the Oracle WebLogic Environment for the Billing and Payment Application Domain

You must set environment variables and other options in the Oracle WebLogic environment to
correctly set up the Billing and Payment domain.

This task is a step in "Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 46.

To set environment variables for the Billing and Payment application domain

1   Open the setDomainEnv file in a text editor.

    This file is located in the domain's home directory, for example:

    ■   **UNIX.** $WEBLOGIC_HOME/user_projects/domains/ebilling_domain/bin/setDomainEnv.sh

    ■   **Windows.**
        %WEBLOGIC_HOME%\user_projects\domains\ebilling_domain\bin\setDomainEnv.cmd

2   In the setDomainEnv.sh file, define the environment variable *EDX_HOME* as the directory in which Oracle Self-Service E-Billing is installed, for example:

■   **Oracle Solaris.** export EDX_HOME=/opt/Oracle/eBilling

■   **Linux.** export EDX_HOME=/opt/Oracle/eBilling

■   **Windows.** set EDX_HOME=C:\oracle\eBilling

3   Add the following entries to the file:

■   **Oracle Solaris.** export CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/lib/ xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.0.jar

■   **Linux.** CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/lib/xercesImpl- 2.7.1.jar:$EDX_HOME/lib/xalan-2.7.0.jar

■   **Windows.** set CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\lib\xercesImpl- 2.7.1.jar;%EDX_HOME%\lib\xalan-2.7.0.jar

4   In the JAVA OPTIONS section, add the Dedx.home Java option section to the end of the definition:

■   Oracle Solaris:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} ${JAVA_PROFILE} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

■   Linux:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

■   Windows (The slashes (/) in the following statement are correct):

```
set JAVA_VM=%JAVA_VM% %JAVA_DEBUG% %JAVA_PROFILE% -Dedx.home=%EDX_HOME% -
Dlog4j.configuration=file:\\\%EDX_HOME%\config\log4j.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=%EDX_HOME%\config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl
```

5   Save the changes, and close the file.

Enabling HTTPS on Your Server for the Billing and Payment Application

Follow these steps to enable HTTPS on your server for the Billing and Payment application, required for compliance with the Payment Card Industry Data Security Standard.

This task is a step in *"Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 46*.

To enable HTTPS on your server for the Billing and Payment application

1   Log in to the Billing and Payment domain console, for example:

`http://localhost:7001/console/`

where:

- *localhost* is the name of the server where you installed the Billing and Payment application.

- *7001* is the port number where you installed the Billing and Payment application.

2   Click Lock and Edit.

3   Click your domain name, Environment, and Servers.

4   In the Servers table, click the server where you want to deploy your application.

5   Select Configuration, General tab, and then click SSL Listen Port Enabled. Enter an SSL port number, then click Save.

6   Click Activate Changes to commit the changes.

Configuring JDBC Resources for the Billing and Payment Application

You must configure the following JDBC resources for applications deployed with Oracle WebLogic using the Oracle WebLogic Administration Console:

- Data sources

- Connection pools

This task is a step in *"Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 46*.

Configuring the Data Sources for the Billing and Payment Application

You must create the following data sources for the Billing and Payment application:

■ edxAdminDataSource

■ edxXMADataSource

■ reportDataSource

To create the data sources for the Billing and Payment application

1 Start the newly created Billing and Payment domain, and open the Oracle WebLogic Administration Console in a browser. The default URL is

http://*Server_Name*:*Server_Port*/console

where:

- ■ *Server_Name* is the name of the server with the Billing and Payment domain.

- ■ *Server_Port* is the port number of the Billing and Payment domain server.

2 Log on to the Oracle WebLogic Administration Console using the user name and password defined when the domain was created.

3 Click Lock and Edit to allow you to make configuration changes.

4 Click the Data Sources link under JDBC in the services section.

5 Click New to create a new data source.

6 Enter the data source name and the JNDI name.

7 For each data source, select the Database Type as Oracle, and enter other settings as shown in the following table. Be sure to select the correct version of the driver for your database version, including patches. Click Next.

Data Source Name	JNDI Name	Database /Driver	Database Details
edxAdminDataSource	edx.databasePool	Oracle/ Driver (Thin XA)	OLTP/ Hostname:ListenPort:SID
edxXMADataSource	edx.xma.databasePool	Oracle/ Driver (Thin XA)	OLTP/ Hostname:ListenPort:SID
reportDataSource	edx.report.databasePool	Oracle/ Driver (Thin XA)	OLAP/ Hostname:ListenPort:SID

8 Leave the default transaction options unchanged, then click Next.

9   For Connection Properties, provide the correct values.

Connection Property	Value
Database Name	Your Oracle Database Alias Name
Host Name	Your Oracle Database server host name
Port	DB server listening port
Database User Name	Your Oracle Database User Name
Password	Your Oracle Database Password

10 Click Next, then click Test Configuration to test whether the database connection is configured correctly.

11 Click Next, then target the data source to the server where you want to deploy the application. The default is AdminServer. Click Finish.

12 Click Activate Changes to commit the changes.

13 Repeat from Step 5 to create each remaining data source.

Configuring the Connection Pools for the Billing and Payment Application

You must configure a connection pool for each data source.

To configure the connection pools for the Billing and Payment application

1   Go to the Summary of JDBC Data Sources.

2   Click any data source name link.

3   On the Connection Pool tab, enter the values for the connection pool settings for each JDBC data source you created.

Property	Value
Initial Capacity	5
Maximum Capacity	20
Capacity Increment	5
Statement Cache Type	FIXED
Statement Cache Size	300
Test Connections on Reserve	Checked
Test Frequency	120
Test Table Name	DUAL

Property	Value
Shrink Frequency	15
Login Delay	1

Setting the Mail Server Properties for the Billing and Payment Application

You must configure the notification mail server properties file, notification.xma.xml, with the mail host, message transport protocol, and mail account authentication properties for your organization.

This task is a step in .

To modify the notification mail server properties XMA

1   Open the notification.xma.xml file, found in the *EDX_HOME*/xma/config/com/edocs/common/ notification folder (or the *EDX_HOME*\xma\config\com\edocs\common\notification directory on Windows).

2   Find the bean ID in the TrueTransporterBean. Modify the mail server properties:

■   **mail.host**. Specify a fully qualified IP address or name of a host running the SMTP which can be used to send email.

■   **mail.transport.protocol**. Specify the default message transport protocol.

3   If your company mail server requires mail account authentication, set the following properties. If not, set the mail.smtp.auth property to false, or remove the three properties:

■   **mail.smtp.auth**. If the value is true, your mail server attempts to authenticate the user. Set a mail account registered in your mail server.

■   **mail.user.** If mail.smtp.auth is true, set the user name to use when connecting to the mail server.

■   **mail.password.** If mail.smtp.auth is true, set the user password to use when connecting to the mail server.

4   If your company mail server requires SSL connection, set the mail.smtp.socketFactory.class property. If not, then remove the following property:

```
"mail.smtp.socketFactory.class: If set, specifies the name of a class that
implements the javax.net.SocketFactory interface. This class will be used to
create SMTP sockets. For SSL connection, please set to
javax.net.ssl.SSLSocketFactory"
```

Example of a TrueTransporterBean:

```
<bean id="TrueTransporterBean"
class="com.edocs.common.notification.extensions.TrueTransporter"
scope="singleton">
```

```
<property name="template">

              <ref bean="XSLTemplateBean"/>

    </property>

    <property name="mailProperties">

     <props>

         <!-- For non authentication mail server


         <prop key="mail.yourcompanydomain.com</prop>

         <prop key="mail.transport.protocol">SMTP</prop>

          -->

         <prop key="mail.host">stbeehive.yourcompanydomain.com</prop>

         <prop key="mail.transport.protocol">SMTP</prop>

         <!-- For requiring authentication mail server -->

         <prop key="mail.smtp.auth">true</prop>

         <prop key="mail.user">eBillingAdmin_WW@yourcompanydomain.com</prop>

         <prop key="mail.password">emailAccountPassword</prop>

         <!-- For SSL connection mail server-->

         <prop
    key="mail.smtp.socketFactory.class">javax.net.ssl.SSLSocketFactory</prop>

         <prop key="mail.smtp.socketFactory.port">465</prop>

     </props>

   </property>

    </bean>
```

5   If your company mail server does not use the SMTP default port 25, set the mail.smtp.socketFactory.port property, indicating which port to use with the specified socket factory.

Setting the Global Configuration Properties

You must modify various properties in the global configuration file, globalConfig.xma.xml, including server IP addresses, names, and ports for the Billing and Payment and Customer Service Representative applications.

This task is a step in *"Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 46*.

To modify the global configuration properties

1 Open the globalConfig.xma.xml file, found in the *EDX_HOME*/xma/config/modules folder (or the *EDX_HOME*\xma\config\modules directory on Windows).

2 Find the bean ID called globalConfig. Modify the properties.

Property	Value
ebillingHostName	The correct server name or IP address where the Billing and Payment application is deployed.
csrHostName	The correct server name or IP address where the Customer Service Representative application is deployed.
ebillingApplicationName	The Billing and Payment application name.
csrApplicationName	The Customer Service Representative application name.
*e*billingHttpPort	The Billing and Payment application HTTP port.
ebillingSSLPort	The Billing and Payment application SSL port.
csrSSLPort	The Customer Service Representative application SSL port.
csrhttpPort	The Customer Service Representative application HTTP port.

For example:

```
<beans>

  <bean id="globalConfig"
class="com.edocs.common.configuration.core.GlobalConfig" scope="singleton">

        <property name="encryptAccountNumbers"><value>true</value></property>

    <!-- default value -->

    <property name="ebillingHostName"><value>sdcbd80a036</value></property>

    <property name="csrHostName"><value>sdcbd80a036</value></property>


    <property name="ebillingApplicationName"><value>ebilling</value></property>

    <property name="csrApplicationName"><value>ebillingcsr</value></property>

    <property name="ebillingHttpPort"><value>7010</value></property>

    <property name="ebillingSSLPort"><value>7011</value></property>

    <property name="csrhttpPort"><value>7021</value></property>
```

```
            <property name="csrSSLPort"><value>7022</value></property>

        </bean>

    </beans>
```

Process of Configuring Oracle WebLogic for the Command Center Application

To configure Oracle WebLogic for the Command Center EAR file, perform the following tasks:

1 "Creating the Oracle WebLogic Domain for Command Center" on page 55

2 "Defining the Oracle WebLogic Environment for the Command Center Domain" on page 56

3 "Enabling HTTPS for the Command Center Server" on page 57

4 "Configuring JDBC Resources for the Command Center" on page 57

5 "Setting the Transaction Time (Linux Only)" on page 60

6 "Configuring JMS Resources for the Command Center on Oracle WebLogic" on page 61

This process is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 45.

Creating the Oracle WebLogic Domain for Command Center

You must create an Oracle WebLogic domain for the Oracle Self-Service E-Billing Command Center EAR file.

This task is a step in "Process of Configuring Oracle WebLogic for the Command Center Application" on page 55.

To create an Oracle WebLogic domain for the Command Center

1 Run the following command from the $WL_HOME/common/bin directory (or the %WL_HOME%\common\bin directory on Windows):

■ **UNIX.** config.sh

■ **Windows.** config.cmd

2 On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain, then click Next.

3 Leave the option to automatically configure the domain selected, then click Next.

4 Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.

5 Select the SUNJDK to use for this domain, then click Next.

6 Choose Yes to configure the Oracle WebLogic domain, then click Next.

7 Enter the name of the domain to create, such as cc_domain, use the Customize Environment and Services Settings to configure the port to 7003 and the default Admin Server. Click Create.

Defining the Oracle WebLogic Environment for the Command Center Domain

You must set environment variables and other options in Oracle WebLogic environment for Command Center.

This task is a step in

To set environment variables for the Command Center application domain

1 Open the file setDomainEnv in a text editor. This file is located in the domain's home directory, for example:

■ **UNIX.** $WEBLOGIC_HOME/user_projects/domains/cc_domain/bin/setDomainEnv.sh

■ **Windows.** %WEBLOGIC_HOME%\user_projects\domains\cc_domain\bin\setDomainEnv.cmd

2 In the file, define the environment variable EDX_HOME as the directory in which Oracle Self-Service E-Billing is installed, for example:

■ **Oracle Solaris.** export EDX_HOME=/opt/Oracle/eBilling

■ **Linux.** export EDX_HOME=/opt/Oracle/eBilling

■ **Windows.** set EDX_HOME=C:\oracle\eBilling

3 Add the following entries to the file:

■ **Oracle Solaris.** export CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.0.jar

■ **Linux.** export CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.0.jar

■ **Windows.** set CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\lib\xercesImpl-2.7.1.jar;%EDX_HOME%\lib\xalan-2.7.0.jar

4 In the JAVA OPTIONS section, add the -Dedx.home Java option section to the end of the JAVA_VM variable definition:

■ Oracle Solaris:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} ${JAVA_PROFILE} -Dedx.home=${EDX_HOME} -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl

export JAVA_VM
```

■ Linux:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} -Dedx.home=${EDX_HOME} -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

■ Windows:

```
set JAVA_VM=%JAVA_VM% %JAVA_DEBUG% %JAVA_PROFILE% -Dedx.home=%EDX_HOME% -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl
```

5 Save the changes, and close the file.

Enabling HTTPS for the Command Center Server

Follow these steps to enable HTTPS on your server for the Command Center application, required for compliance with the Payment Card Industry Data Security Standard.

This task is a step in

To enable HTTPS on your server for the Command Center application

1 Log in to the Command Center domain console, for example:

http://localhost:7003/console/

2 Click Lock and Edit.

3 Click your domain name, Environment, and then click Servers.

4 In the Servers table, click the server where you want to deploy your application.

5 Select Configuration, General tab, and then click SSL Listen Port Enabled. Enter an SSL port number.

Configuring JDBC Resources for the Command Center

You must configure JDBC resources for the Command Center application deployed with Oracle WebLogic using the Oracle WebLogic Administration Console.

You must configure the following

■ Data sources

■ Connection pools

This task is a step in "Process of Configuring Oracle WebLogic for the Command Center Application" on page 55.

Configuring Data Sources for the Command Center Domain

You must configure the following data sources for running Oracle Self-Service E-Billing jobs and the reporting application:

■ edxAdminDataSource

■ edxXMADataSource

■ reportDataSource

■ edxLoggerDataSource

■ edxMessagingDataSource

■ edxUserDataSource

To create data sources for the Command Center domain

1 Start the newly created Command Center domain, and open the Oracle WebLogic Administration Console in a browser. The default URL

http://*Server_Name*:*Server_Port*/console

where:

■ *Server_Name* is the name of the server with the Command Center domain.

■ *Server_Port* is the port number of the Command Center domain server.

2 Log on to the Oracle WebLogic Administration Console using the user name and password defined when the domain was created.

3 Click Lock and Edit to allow you to make configuration changes.

4 Click the Data Sources link under JDBC in the services section.

5 Click New to create a new data source.

6 Enter the data source name and the JNDI name.

7 For each data source, select the database type as Oracle and enter other settings as shown in the following table. Be sure to select the correct version of the driver for your database version, including patches. Click Next.

Data Source Name	JNDI Name	DB/ Driver	Settings	DB Details
edxAdminDataSource	edx.databasePool	Oracle/ Driver (Thin)	Emulate Two- Phase Commit	OLTP/ Hostname:Listen Port:SID
edxXMADataSource	edx.xma.database Pool	Oracle/ Driver (Thin XA)	Not applicable	OLTP/ Hostname:Listen Port:SID
reportDataSource	edx.report.database Pool	Oracle/ Driver (Thin XA)	Not applicable	OLAP/ Hostname:Listen Port:SID
edxLoggerDataSource	edx.logger.database Pool	Oracle/ Driver (Thin)	Emulate Two- Phase Commit	OLTP/ Hostname:Listen Port:SID
edxMessagingDataSource	edx.messaging. databasePool	Oracle/ Driver (Thin XA)	Not applicable	OLTP/ Hostname:Listen Port:SID
edxUserDataSource	edx.user.database Pool	Oracle/ Driver (Thin)	Emulate Two- Phase Commit	OLTP/ Hostname:Listen Port:SID

8 Select Supports Global Transactions for all Data Sources, select the corresponding transaction option, then click Next.

9 On the Connections Properties page, provide the correct values for the following properties.

Connection Property	Description
Database Name	Your Oracle Database Alias Name
Host Name	Your Oracle Database Server Host Name
Port	Database Server Listening Port
Database User Name	Your Oracle Database User Name
Password	Your Oracle Database Password

10 Click Test Configuration to test whether the database connection is configured correctly. Click Next.

11 Target the data source to the Command Center domain (default is AdminServer), then click Finish.

12 Click Activate Changes to commit the changes.

Configuring Connection Pools for the Command Center Domain

You must configure connection pools for each JDBC data source you created.

To configure the connection pools for Command Center

1 Go to the Summary of JDBC Data Sources.

2 Click any data source name link.

3 On the Connection Pool tab, enter the values for the connection pool settings for each JDBC data source you created.

Property	Value
Initial Capacity	5
Maximum Capacity	20
Capacity Increment	5
Statement Cache Type	FIXED
Statement Cache Size	300
Text Connections on Reserve	Selected
Test Frequency	120
Test Table Name	DUAL
Shrink Frequency	15
Login Delay	1

Setting the Transaction Time (Linux Only)

Linux users must set the service transaction time.

This task is a step in

To set the transaction time (Linux only)

1 Select Services, JTA.

2 Update the value of Transaction Timeout from 30 to 60.

3 Save the change.

4   Stop and restart the Command Center domain.

Configuring JMS Resources for the Command Center on Oracle WebLogic

You must configure the following JMS resources for the Command Center:

■   JMS Persistence Store

■   JMS Servers

■   JMS Connection Factories

■   JMS Queues

This task is a step in "Process of Configuring Oracle WebLogic for the Command Center Application" on page 55.

Configuring a JMS Persistence Store for the Command Center

You must configure a JMS Persistence Store for the Command Center.

To configure a JMS Persistence Store

1   Select Services, Persistence Stores.

2   Select New, Create File Store. Create a file store with the following values.

Name	Target	Directory
LoggerFileStore	Select Your Server (AdminServer)	Directory Location to Save Store (such as `$WEBLOGIC_HOME/ user_projects/domains/ commandcenter_domain/ config/FileStore`)

Configuring a JMS Server

You must set up a JMS Server for the Command Center.

To configure a JMS Server

■   Select Services, Messaging, JMS Servers, and New. Create one JMS Server using the following values.

Name	Persistent Store	Target
LoggerJMSServer	LoggerFileStore	Select Your Server (AdminServer)

Configuring JMS Connection Factories for Command Center

To create JMS Connection Factories you must first create the Module.

To create the JMS Module

1 Click Services, Messaging, and the JMS Modules link on the Administration Console page for the Command Center domain. Click Lock and Edit to allow configuration changes to be made.

2 Click New to create a new JMS Module. Set the name of the JMS system Module, and provide the JMS Module properties. Click Next.

Property	Description	Example
Name	Your JMS Module Name	CC_JMSFactories
Descriptor File Name	Your Descriptor File Name	CC_JMSFactories
Location in Domain	Your Location in Domain	CC_JMSFactories
Targets	Select Your Server	AdminServer

3 Select the check boxes to target the system Module at the servers running the Command Center application, click Next, and then click Finish.

Once you have created the Module you can add a factory to the new module.

To create a JMS Connection Factory

1 Click the JMS Modules link on the Administration Console page for the Command Center domain. Click Lock and Edit to allow configuration changes to be made.

2 Click the link for the module you created in the previous procedure.

3 Click New, select Connection Factory, and then click Next.

4 Add one Connection Factories using the following parameters. Click Finish.

Connection Factory Type	Name	JNDI Name	Target
Connection Factory	LoggerConnectionFactory	edx.lcf	Select Your Server (AdminServer)

5 Click on the new connection factory, and select XA Transaction on the Transactions tab.

You must set up a JMS queue for the Command Center.

Configuring a JMS Queue for the Command Center

You must create a JMS Queue for the Command Center application.

To add the JMS Queue Destination

1 Click the JMS Modules link on the Administration Console page for the Command Center domain. Click Lock and Edit to allow configuration changes to be made.

2 Click the link for the module you created. Click New, select Queue, and then click Next.

3 Provide the Queue Name and the JNDI Name as shown in the following table. Click Next.

Name	JNDI Name	Target
LoggerQueue	edx.queue.logger	LoggerJMSServer

4 Click Create a New Subdeployment. Oracle WebLogic shows the newly created queue name as Subdeployment Name.

5 Accept the name, then click OK. If you plan to add more JMS resources into the subdeployment, you can rename it. Select the newly created Subdeployment and the corresponding target JMS Server.

6 Click Activate Changes to commit changes after the JMS configuration is complete.

Process of Configuring Oracle WebLogic for the Customer Service Representative Application

To configure Oracle WebLogic for the Customer Service Representative application, perform the following tasks:

1 *"Creating the Oracle WebLogic Domain for the Customer Service Representative Application" on page 64*

2 *"Defining the Oracle WebLogic Environment for the Customer Service Representative Domain" on page 64*

3 *"Accessing the Oracle WebLogic Console" on page 65*

4 *"Enabling HTTPS on Your Server for the Customer Service Representative Application" on page 66*

5 Set the same data sources and connection pool settings for the Customer Service Representative application as you did for the Billing and Payment application.

For details, see *"Configuring JDBC Resources for the Billing and Payment Application" on page 49*.

6 *"Modifying the csr.xma.xml File for Customer Service Representative Properties" on page 66*

This process is a step in *"Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 45*.

Creating the Oracle WebLogic Domain for the Customer Service Representative Application

You must create an Oracle WebLogic domain for the Customer Service Representative EAR file.

This task is a step in

To create an Oracle WebLogic domain for the Customer Service Representative EAR file

1 Run the following command from the WL_HOME/common/bin directory (or the WL_HOME\common\bin directory on Windows):

  ■ **UNIX.** config.sh

  ■ **Windows.** config.cmd

2 On the Oracle WebLogic Configuration Wizard select Create a new WebLogic domain. Click Next.

3 Leave the option to automatically configure the domain selected, then click Next.

4 Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.

5 Select the SUNJDK to use for this domain, then click Next.

6 Choose Yes to configure the Oracle WebLogic domain, then click Next.

7 Enter the name of the domain to create, such as CSR_domain, and enter a location for the domain, such as port 7006 and default Admin Server. Click Create.

Defining the Oracle WebLogic Environment for the Customer Service Representative Domain

You must set environment variables and other options in the Oracle WebLogic environment to correctly set up the Customer Service Representative domain.

This task is a step in

To set environment variables for the Oracle WebLogic Customer Service Representative domain

1 Go to the $WEBLOGIC_HOME/user_projects/domains/csr_domain/bin directory (the %WEBLOGIC_HOME%\user_projects\domains\csr_domain\bin directory on Windows), and open the following file in a text editor:

  ■ **UNIX.** setDomainEnv.sh

  ■ **Windows.** setDomainEnv.cmd

2   In the file, define the environment variable EDX_HOME as the directory where the Billing and Payment application is installed, for example:

■   **Oracle Solaris.** export EDX_HOME=/opt/Oracle/eBilling

■   **Linux.** export EDX_HOME=/opt/Oracle/eBilling

■   **Windows.** set EDX_HOME=C:\oracle\eilling

3   Add the following entries to the file:

■   **Oracle Solaris.** CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/lib/xalan-2.7.0.jar

■   **Linux.** CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/lib/xalan-2.7.0.jar

■   **Windows.** set CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\lib\xalan-2.7.0.jar

4   In the JAVA OPTIONS section, add the Java option section to the end of the JAVA_VM variable definition:

■   Oracle Solaris:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} ${JAVA_PROFILE} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_csr.xml -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

■   Linux:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_csr.xml -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

■   Windows (The slashes (/) in the following statement are correct):

```
set JAVA_VM=%JAVA_VM% %JAVA_DEBUG% %JAVA_PROFILE% -Dedx.home=%EDX_HOME% -
Dlog4j.configuration=file:\\\%EDX_HOME%\config\log4j_csr.xml -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl
```

5   Save the changes, and close the file.

Accessing the Oracle WebLogic Console

Once the server is running, you can access the Oracle WebLogic console.

To access the Oracle WebLogic console after the server is running

■ Use the following URL

http://*localhost*:*7006*/console

where:

■ *localhost* is the name of the server where you installed the Customer Service Representative application.

■ *7006* is the port number where you installed the Customer Service Representative application.

Enabling HTTPS on Your Server for the Customer Service Representative Application

Follow these steps to enable HTTPS on your server for the Customer Service Representative application, required for compliance with the Payment Card Industry Data Security Standard.

This task is a step in "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 63.

To enable HTTPS on your server for the Customer Service Representative application

1 Log in to the Customer Service Representative domain console, for example:

http://localhost:7006/console/

2 Click Lock and Edit.

3 Click your domain name, Environment, and Servers.

4 In the Servers table, click the server where you want to deploy your application.

5 Select Configuration, General tab, and then click SSL Listen Port Enabled. Enter an SSL port number.

Modifying the csr.xma.xml File for Customer Service Representative Properties

You must specify the Customer Service Representative application URLs for your implementation as well as your customer service phone number in the csr.xma.xml file.

This task is a step in "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 63.

To modify the csr.xma.xml file

1 Open the csr.xma.xml file, located in the *EDX_HOME*/xma/config/modules/application/csr directory (or the *EDX_HOME*\xma\config\modules\application\csr directory on Windows).

2 Modify the custAppURL property to point to the correct HTTPS server name or IP address where the Oracle Self-Service E-Billing Customer Service Representative application is deployed.

The Customer Service Representative application uses this URL string to impersonate users. The default is

https://localhost:7001/ebilling/j_acegi_security_check?

3 Modify the custLogoutAppURL property for the HTTPS URL directed to when logging out.

4 Modify the customerServicePhone property to display the correct customer service phone number (default is xxx-xxx-xxxx).

Process of Repackaging the GNU Lesser General Public License

You must repackage the GNU Lesser General Public License (LGPL) on Oracle Self-Service E-Billing (all platforms).

To complete repackaging the GNU LGPL, perform the following tasks:

1 "Setting Up Maven" on page 67

2 "Setting Up a Proxy Maven Configuration" on page 68

3 "Repackaging LGPL" on page 69

This process is a step in the following roadmaps:

■ "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 45.

■ "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156

Setting Up Maven

Before you can repackage the LGPL, you must set up Maven.

This task is a step in "Process of Repackaging the GNU Lesser General Public License" on page 67.

To set up Maven

1 Go to the following directory, and download Maven 2.0.7 (Oracle Self-Service E-Billing also supports versions 2.0.5 and 2.0.6):

http://maven.apache.org

2 Unzip the downloaded archive to your local Maven home directory, such as maven-2.0.7.

3 Create the environment variable M2_HOME, where *Maven_Home_Dir* is the Maven home path. For example:

■ **UNIX.** export M2_HOME=Maven_Home_Dir

■ **Windows.** set M2_HOME=Maven_Home_Dir

4 Add the bin directory from M2_HOME to a PATH variable:

- **UNIX.** export PATH=$M2_HOME/bin: $PATH;

- **Windows.** set PATH=%M2_HOME%\bin; %PATH%

Setting Up a Proxy Maven Configuration

If your computer does not have direct access to the Internet, set up a proxy in the Maven configuration.

This task is a step in

To set up a proxy Maven configuration

- Edit the settings.xml file, found in the %M2_HOME%/conf directory (or the %M2_HOME%\conf directory on Windows) to add the following lines to the appropriate section of the file:

```
<settings>

.

.

<proxies>

 <proxy>

     <active>true</active>

     <protocol>http</protocol>

     <host>www-YourCompanyproxy.com</host>

     <port>80</port>

<nonProxyHosts>localhost|*.YourCompanyX.com|YourCompanyY.com|YourCompanyZ.com</nonProxyHosts>

     </proxy>

  </proxies>

  .

  .

</settings>
```

where:

- *YourCompanyproxy.com* is your company proxy site.

■   *YourCompanyX.com, YourCompanyY.com*, and *YourCompanyZ.com* are your local host sites.

Repackaging LGPL

Follow these steps to repackage LGPL for Oracle Self-Service E-Billing.

This task is a step in

To repackage LGPL

1   Go to the directory to the *EDX_HOME/* repackage directory (or the *EDX_HOME*\repackage directory in Windows).

2   Run the following Maven installation command:

```
mvn install
```

This command places all EAR files in the packaged LGPL libraries in the $EDX_HOME/J2EEApps directory (or the %EDX_HOME%\J2EEApps directory on Windows).

If you receive an error similar to this one, or any error indicating jars are missing, the command was not able to access and download LGPL components from the Internet. Follow the instructions in this chapter to manually download and install these components.

```
[INFO] ------------------------------------------------------------------------
---

[INFO] Building eBilling application

[INFO]    task-segment: [install]

[INFO] ------------------------------------------------------------------------
---

[INFO] ------------------------------------------------------------------------
[ERROR] BUILD ERROR

[INFO] ------------------------------------------------------------------------
[INFO] Failed to resolve artifact.
```

To manually download and install the required LGPL jars

1   Go to the following location, and download the swarmcache-1.0RC2.jar file

http://mirrors.ibiblio.org/pub/mirrors/maven2/swarmcache/swarmcache/1.0RC2/

2   Install the swarmcache-1.0RC2.jar file using the following command:

```
mvn install:install-file -DgroupId=swarmcache -DartifactId=swarmcache \

    -Dversion=1.ORC2 -Dpackaging=jar -Dfile=/path/to/file
```

3 Go to the following location, and download the hibernate-3.1.3.jar file

http://mirrors.ibiblio.org/pub/mirrors/maven2/org/hibernate/hibernate/3.1.3/

4 Install the hibernate-3.1.3.jar file using the following command:

```
mvn install:install-file -DgroupId=org.hibernate -DartifactId=hibernate \

    -Dversion=3.1.3 -Dpackaging=jar -Dfile=/path/to/file
```

5 Go to the following location, and download the jgroups-all-2.4.1.jar file

http://mirrors.ibiblio.org/pub/mirrors/maven2/jgroups/jgroups-all/2.4.1/

6 Install the jgroups-all-2.4.1.jar file using the following command:

```
mvn install:install-file -DgroupId=jgroups -DartifactId=jgroups-all \

    -Dversion=2.4.1 -Dpackaging=jar -Dfile=/path/to/file
```

7 Repackage LGPL.

Deploying Oracle Self-Service E-Billing Applications on Oracle WebLogic

You must deploy each of the three EAR files for the following applications using your application server:

■ Billing and Payment (end-user interface)

■ Command Center (production environment)

■ Customer Service Representative

This task is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 45.

To deploy applications on Oracle WebLogic

1 Open the Oracle WebLogic Administration Console, then click Lock and Edit.

2 Click the Deployments link to display the deployments page for the Oracle WebLogic domain for the particular application you are deploying, such as ebilling_domain.

3 Click Install to start the Install Application Assistant, which guides you through the steps required to deploy the Oracle Self-Service E-Billing applications.

4 Click the links to navigate to the location of the EAR file to deploy.

Oracle Self-Service E-Billing J2EE Application	File Name and Location
Billing and Payment	File Name: ebilling-weblogic-10-6.0.4.EAR Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/ebilling/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\ebilling\weblogic\
Command Center	File Name: command-center-weblogic-10-6.0.4.EAR Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/commandcenter/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\commandcenter\weblogic
Customer Service Representative	File Name: csr-app-6.0.4.EAR Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/csr/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\csr\weblogic

5 Select the deployed EAR file, and start all services located at the Start, Servicing all requests.

6 Restart the server.

Repeat this procedure for each Oracle Self-Service E-Billing application.

Configuring and Starting Scheduler on Oracle WebLogic

Scheduler is a program that administrators use to schedule Command Center jobs; it must be running for scheduled jobs to execute.

Follow the procedure appropriate for your operating system to configure Scheduler on Oracle WebLogic:

■

■

NOTE: For information on using Scheduler, see *Administration Guide for Oracle Self-Service E-Billing*.

Configuring and Starting Scheduler on Oracle WebLogic and UNIX

Follow these steps to configure and start Scheduler on Oracle WebLogic and UNIX.

To configure and start Scheduler on Oracle WebLogic and UNIX

1 Go to the *EDX_HOME/*bin directory.

2 Run the following command:

    $./edx_config

3 Specify the appropriate details for the database server, Oracle home directory, and application server.

Field	Example Value
Enter database server	oracle
Enter oracle home directory	/export/home/oracle11gR2/ 11.2.0.1
Oracle database user name (DB_Username)	oltp
Oracle database password (DB_PWD)	oltp
Oracle database alias (tnsname)	oltp
Application server	wl
Java root directory	/opt/bea10mp1/jdk150_11
WebLogic Application Server root directory	/opt/bea10mp1/wlserver_10.0

4 Start Scheduler using the following command, located in the *EDX_HOME/*bin directory:

    ./wl_scheduler -start -url t3://*localhost*:*7003* –verbose

where:

■ *localhost* is the name of the Command Center application server.

■ *7003* is the port number of the Command Center application server.

To stop Scheduler, replace the -start parameter with the -stop parameter:

    ./wl_scheduler -stop -url t3://*localhost*:*7003* –verbose

Configuring and Starting Scheduler on Oracle WebLogic on Windows

Follow these steps to configure and start Scheduler on Oracle WebLogic and Windows.

To configure and start Scheduler on Oracle WebLogic and Windows

1 Open the edx_env.bat file in the *EDX_HOME*\config directory, and add the following lines, where *Your_Java_Home* is the directory where Java is installed:

> @set EDX_HOME=D:\oracle\ebilling
>
> @set APP_SERVER=wl
>
> @set WL_HOME=D:\bea\wlserver_10.0
>
> @set JAVA_HOME=*Your_Java_Home*

2 Go to the *EDX_HOME*\bin directory, and run the following command:

> wl_scheduler.bat -start -url t3://*localhost*:*7003* –verbose

where:

- ■ *localhost* is the name of the Command Center application server.
- ■ *7003* is the port number of the Command Center application server.

To stop Scheduler, replace the -start parameter with the -stop parameter in the same command:

> wl_scheduler.bat -stop -url t3://*localhost: 7003* –verbose

Running the Sample Oracle Self-Service E-Billing Applications on Oracle WebLogic

After successfully deploying the Oracle Self-Service E-Billing application EAR files, you can log in to the sample Oracle Self-Service E-Billing applications.

This task is a step in .

To run the sample Oracle Self-Service E-Billing applications

1 In your browser, point to the Oracle Self-Service E-Billing application name, shown in the following table, specifying the local host (server name) and port number where you deployed the Oracle Self-Service E-Billing application.

Sample Oracle Self-Service E-Billing Application	URL Example
Billing and Payment	http://localhost:7001/ebilling
Command Center	http://localhost:7003/eBilling
Customer Service Representative	http://localhost:7006/ebillingcsr

The sample log in page appears.

2 For the sample Billing and Payment application, log in as B2B admin user ftown or one of the other enrolled users.

Note that the user you log in as determines whether you see the Business (B2B) or Consumer (B2C) Edition of the Billing and Payment application. For information about enrolling for the first time, see *Application Guide for Oracle Self-Service E-Billing (Business Edition)* or *Application Guide for Oracle Self-Service E-Billing (Consumer Edition)*.

User Name	Password	Role
ftown	ftown	B2B Admin User
jsmith	jsmith	Admin
twalsh	twalsh	B2C User
tbrown	tbrown	B2C User

5 Configuring IBM WebSphere

This chapter describes how to configure IBM WebSphere for Oracle Self-Service E-Billing. It includes the following topics:

- Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing on page 75
- Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application on page 76
- Process of Configuring the IBM WebSphere Application Server for the Command Center Application on page 90
- Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application on page 102
- Process of Deploying Oracle Self-Service E-Billing Applications on IBM WebSphere on page 111
- Configuring and Starting Scheduler on IBM WebSphere on page 114
- Running the Sample Oracle Self-Service E-Billing Applications on IBM WebSphere on page 117

Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing

To configure IBM WebSphere for Oracle Self-Service E-Billing, perform the following processes and tasks:

- "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76
- "Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90
- "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102
- "Process of Deploying Oracle Self-Service E-Billing Applications on IBM WebSphere" on page 111
- "Configuring and Starting Scheduler on IBM WebSphere" on page 114
- "Running the Sample Oracle Self-Service E-Billing Applications on IBM WebSphere" on page 117

This roadmap is part of "Roadmap for Installing Oracle Self-Service E-Billing" on page 17.

Preparing to Configure IBM WebSphere

Before configuring IBM WebSphere, you must verify that the following tasks are complete:

■ The database server components must be installed and configured for Oracle Self-Service E-Billing.

For details on setting up the database, see "Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25.

■ For distributed environments, verify that you have any required database client software installed on the IBM WebSphere application server and any other client computers of your database server.

The instructions to configure IBM WebSphere assume in-depth understanding of and practical experience with application server administration. Consult the IBM WebSphere documentation as necessary.

NOTE: The installation and configuration examples in this guide use default Oracle Self-Service E-Billing paths, privileges and permissions. If you choose not to accept the default values, make sure your values are consistent on all servers across your installation of Oracle Self-Service E-Billing.

This task is a step in "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75.

Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application

To configure IBM WebSphere for the Billing and Payment J2EE application, perform the following tasks

1 "Creating an IBM WebSphere Profile for the Billing and Payment Application" on page 77

2 "Accessing the IBM WebSphere Administration Console on the Billing and Payment Application Server" on page 78

3 "Setting Up Application Security" on page 79

4 "Securing Access to the IBM WebSphere Administration Console for the Billing and Payment Application" on page 80

5 "Enabling HTTPS on Your Server for the Billing and Payment Application" on page 81

6 "Setting the JVM Options" on page 81

7 "Configuring the JDBC Providers" on page 82

8 "Configuring the Data Sources" on page 82

9 "Creating a J2C Authentication Alias for the Billing and Payment Application" on page 84

10 "Setting Up the Web Container Filter for Billing and Payment in IBM WebSphere Version 6.1.0.3 and Higher" on page 84

11 "Setting Up the Log4j Output for Billing and Payment in IBM WebSphere Version 6.1.0.3 and Higher" on page 85

12 "Verifying the Billing and Payment, and Command Center Configurations" on page 85

13 "Modifying the Configuration XMA File for the Billing and Payment Application" on page 87

This process is a step in "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75.

Creating an IBM WebSphere Profile for the Billing and Payment Application

You must create an IBM WebSphere profile for the Billing and Payment application. The profile calls a properties file, ebilling.props, which you must first create.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

Creating the Billing and Payment Application Properties File

You must create a properties file for the Billing and Payment profile.

To create the Billing and Payment properties file

1 Create a file called ebilling.props. The file can reside in any directory.

2 Add the following text to the file:

```
WC_defaulthost=9080

WC_adminhost=9060

WC_defaulthost_secure=9443

WC_adminhost_secure=9043

BOOTSTRAP_ADDRESS=2809

SOAP_CONNECTOR_ADDRESS=8880

SAS_SSL_SERVERAUTH_LISTENER_ADDRESS=9401

CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS=9403

CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS=9402

ORB_LISTENER_ADDRESS=9100

DCS_UNICAST_ADDRESS=9353

SIB_ENDPOINT_ADDRESS=7276

SIB_ENDPOINT_SECURE_ADDRESS=7286
```

SIB_MQ_ENDPOINT_ADDRESS=5558

SIB_MQ_ENDPOINT_SECURE_ADDRESS=5578

Creating a Profile for the Billing and Payment Application

Once you have created the properties file you can create the IBM WebSphere profile for the Billing and Payment application.

To create a profile for the billing application

1   From the command line, go to the *App_Server_Root/*bin directory, where *App_Server_Root* is your IBM WebSphere installation home path

2   Use the wasprofile command to create an IBM WebSphere profile:

    wasprofile.sh

    -create

    -profileName eBilling_profile

    -profilePath "*App_Server_Root*/profiles/eBilling_profile"

    -templatePath "*App_Server_Root*/profileTemplates/default"

    -nodeName eBilling_node

    -cellName eBilling_cell

    -hostName localhost

    -portsFile "/export/home/ebilling.props"

NOTE: Make sure the ebilling.props file is located in the directory you specify.

3   List the profile to make sure it was created successfully:

    ./wasprofile.sh -listProfiles

Accessing the IBM WebSphere Administration Console on the Billing and Payment Application Server

Once you have created the Billing and Payment profile you can start the Billing and Payment application server and access the IBM WebSphere Administration Console.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To access the IBM WebSphere Administration Console

1   Run the StartServer script, located in the *App_Server_Root*/profiles/*Profile_Name*/bin directory:

       `./startServer.sh` *Server_Name*

where:

- *App_Server_Root* is your IBM WebSphere installation home path.
- *Profile_Name* is the name of Billing and Payment profile.
- *Server_Name* is the name of the Billing and Payment application server.

For example:

       `./startServer.sh server1`

2 Access the following URL

    http://*Host_Name:WC_adminhost*/ibm/console

where:

- *Host_Name* is the name of the Billing and Payment application server.
- *WC_adminhost* is the port number, and can be found in the ebilling.props file.

Setting Up Application Security

After starting the IBM WebSphere server you must set up application security.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To set up application security

1 Log on to the IBM WebSphere Administration Console on the Billing and Payment application server.

2 Select Security, Global security.

3 Click the Security Configuration Wizard button.

4 Select Enable application security, then click Next.

5 Select Federated repositories, then click Next.

6 Enter the password websphere, and enter it again when prompted to confirm.

7 Enter the following option values.

Configure Security Option	Value
Enable administrative security	True
Enable application security	True
Use Java 2 security to restrict application access to local resources	False

Configure Security Option	Value
User repository	Federated repositories
Primary administrative user name	websphere

8 Click Next, then click Finish. Click Save to save to the master configuration.

Once you have set up application security you must stop and start the Billing and Payment application server.

To stop and restart the application server

1 Run the StopServer script file, located in the *App_Server_Root*/profiles/*Profile_Name*/bin directory:

./stopServer.sh *Server_Name*

where:

■ *App_Server_Root* is your IBM WebSphere installation home path.

■ *Profile_Name* is the name of Billing and Payment profile.

■ *Server_Name* is the name of the Billing and Payment application server.

For example: ./stopServer.sh server1

2 Run the StartServer script, located in the same directory:

./startServer.sh *Server_Name*

Securing Access to the IBM WebSphere Administration Console for the Billing and Payment Application

Once you have set up application security you must secure access to the IBM WebSphere Administration console.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To secure access to the IBM WebSphere Administration console

■ Enter the URL

https://*Host_Name*:*WC_adminhost_secure*/ibm/console

where:

■ *Host_Name* is the name of the Command Center application server.

■ *WC_adminhost_secure* is the port number, which can be found in the ebilling.props file.

Enabling HTTPS on Your Server for the Billing and Payment Application

Follow these steps to enable HTTPS on your server for the Billing and Payment application, required for compliance with the Payment Card Industry Data Security Standard.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To enable HTTPS on your server for the Billing and Payment application

1 Log on to the IBM WebSphere console, for example:

https://localhost:9043/ibm/console/

2 Click Servers, then Application Servers. In the Application servers table, click the server where you want to deploy your application.

3 Under the Configuration tab, click Web Container Settings, then click Web container transport chains.

4 Enable WCInboundDefault and WCInboundDefaultSecure.

5 Click Apply.

Setting the JVM Options

You must set the JVM options for running the Billing and Payment application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To set the JVM options

1 Select Servers, Application servers, server1, Java and Process Management, Process Definition, and Java Virtual Machine.

2 In the Classpath properties, enter the full path to *EDX_HOME*/config, where *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed, such as:

/export/home/qa1/ebilling601/config

3 In Generic JVM arguments properties, enter the following values:

-Dedx.home=<EDX_HOMNE>

-Dedx.jms.authen.username=websphere

-Dedx.jms.authen.password=websphere

-Dlog4j.configuration=file:<EDX_HOME>/config/log4j.xml

-Dorg.owasp.esapi.resources=<EDX_HOME>/config

Configuring the JDBC Providers

You must configure two JDBC providers for the Billing and Payment application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To configure the JDBC providers

1 Select Resource, JDBC, and JDBC Provider.

2 Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

- ■ *Node_Name* is the Billing and Payment application server node.

- ■ *Server_Name* is the name of the Billing and Payment application server.

3 Click New to create a new JDBC provider.

4 You must create two JDBC providers as shown in the following table. Specify the data for the first provider.

Database Type	Provider Type	Implementation Type	Name
Oracle	Oracle JDBC Driver	XA data source	Oracle JDBC Driver (XA)
Oracle	Oracle JDBC Driver	Connection pool data source	Oracle JDBC Driver

5 Click Next, and enter database class path information.

For Oracle Database 10*g*, specify the directory location for the ojdbc14.jar file. You can find this file in the *EDX_HOME/*lib directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

6 Click Next to view a summary of the provider you just created, then click Finish. Click Save to save directly to the master configuration.

7 Repeat from Step 3 to create the second JDBC provider.

Configuring the Data Sources

You must configure three data source for the Billing and Payment application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To configure the data sources

1   Select Resource, JDBC, and Data Source.

2   Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

- ■   *Node_Name* is the Billing and Payment application server node.

- ■   *Server_Name* is the name of the Billing and Payment application server.

3   Click New, then click the Create a new J2C authentication alias link. You must create J2C
authentication aliases to store the user name and password of the OLTP and OLAP databases.

4   Click New, and create a J2C authentication alias to store the user name and password of the OLTP
database. In the Alias field enter OLTP, and enter OLTP User for the description.

The user ID and password are the database schema user ID and password the alias uses to
connect to the database.

5   Click Apply, then click Save to save change directly to the master configuration.

6   Repeat from Step 3 to create a J2C authentication alias for the OLAP database. In the Alias field
enter OLAP, and enter OLAP User as the description.

7   Select Resource, JDBC, and Data Source.

8   Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

- ■   *Node_Name* is the Billing and Payment application server node.

- ■   *Server_Name* is the name of the Billing and Payment application server.

9   Click New. You must create three data sources for the Billing and Payment application as shown
in the following table. Specify the data source name (edxAdminDataSource), JNDI name
(edx.databasePool), and J2C authentication alias (OLTP) for the first data source.

Data Source Name	JNDI Name	J2C Alias	JDBC Provider
edxAdminDataSource	edx.databasePool	OLTP	Oracle JDBC Driver (XA)
edxXMADataSource	edx.xma.databasePool	OLTP	Oracle JDBC Driver (XA)
reportDataSource	edx.report.databasePool	OLAP	Oracle JDBC Driver (XA)

10 Click Next, select the Select an existing JDBC provider option, and select the option from the list.

11 Click Next and specify the URL for the database in the form:

jdbc:oracle:thin:@*Host_Name:Port_Name:Database*

where:

- ■   *Host_Name* is the name of the Billing and Payment database server.

■ *Port_Name* is the name of the Billing and Payment database listener port.

■ *Database* is the name of the Billing and Payment database SID.

For example:

■ For OLTP: jdbc:oracle:thin:@sdcbd80a036:1521:oltp10g

■ For OLAP: jdbc:oracle:thin:@sdcbd80a036:1521:olap10g

12 Select Oracle 10g data store helper as the data store helper class name. The data source helper must be consistent with the database that you connect to. For example, if the current database version you connect to is Oracle Database 11*g*, then choose Oracle 11g data store helper as the data store helper class name.

13 Click Next to view a summary of the data source you just created, then click Finish. Click Save to save the new data source directly to the master configuration.

14 To test the newly created data source, select the data source, then click Test connection.

15 Repeat from Step 9 to create the remaining data sources.

Creating a J2C Authentication Alias for the Billing and Payment Application

You must create a J2C authentication alias for the EJB modules Logger and Events.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

To create the J2C authentication alias for the Billing and Payment application

1 Select Security, Secure administration, application, infrastructure, Authentication, Java Authentication and Authorization Services, J2C authentication data.

2 Click New, and enter MessageDispatcher_ASAuthAlias as the alias.

3 Enter websphere as the user ID and as the password.

4 Enter the description as eBilling admin console user name and password.

5 Click Apply, then click Save to save to the master configuration.

Setting Up the Web Container Filter for Billing and Payment in IBM WebSphere Version 6.1.0.3 and Higher

Complete the following procedure to set up the Web container filter on IBM WebSphere version 6.0.1.3 and higher for the Billing and Payment application.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and Payment J2EE Application" on page 76.

*To set up the Web container filter in IBM WebSphere 6.1.0.3 and higher for the
billing application*

1 Log on to the IBM WebSphere console.

2 Select Application servers, server1 (eBilling), Web container, Custom Properties.

3 Create a new property called com.ibm.ws.webcontainer.invokefilterscompatibility. Specify the
Value field as True.

4 Restart the Billing and Payment application server.

Setting Up the Log4j Output for Billing and Payment in IBM WebSphere Version 6.1.0.3 and Higher

Complete the following procedure to set up log4j output on IBM WebSphere Version 6.1.0.17 and
higher for the Billing and Payment application.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and
Payment J2EE Application" on page 76.

*To set up log4j output in IBM WebSphere 6.1.0.17 and higher for the Billing and
Payment application*

1 Create a new file called commons-logging.properties in the $WAS_HOME/profiles/
eBilling_profile/properties directory.

2 Add the following content:

    org.apache.commons.logging.LogFactory=org.apache.commons.logging.impl.LogFactor
    yImpl

3 Correct the log4j file configuration. Access the IBM WebSphere console for the Billing and
Payment application. Select Servers, Application servers, your server, Server Infrastructure, Java
and Process Management, Process Definition, Additional Properties, Java Virtual Machine,
Generic JVM arguments. Add file: to the -Dlog4j.configuration definition:

    -Dlog4j.configuration=file:${your log4j full path and name} example: -
    Dlog4j.configuration=file:/export/home/qa1/eBilling602/config/log4j.xml

4 Restart the Billing and Payment application server.

Verifying the Billing and Payment, and Command Center Configurations

After you configure the Billing and Payment, and Command Center profiles on IBM WebSphere, you
must stop and start each profile, then verify that the messaging engine and service integration bus
links are working in both profiles.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Billing and
Payment J2EE Application" on page 76.

To verify the messaging engine link in the Billing and Payment profile

1   Stop and start the Billing and Payment profile.

2   Select Service integration, buses, eBillingServiceBus, Messaging engines.

3   Check the color of the arrow in the Status column for eBilling_node.server1-eBillingServiceBus.

   If the color of the arrow is dark green, the link status is ON; if it is pale green, the connection has not been made successfully.

Once you have verified the messaging engine link in the Billing and Payment profile, verify the service integration bus link in the Billing and Payment profile.

To verify the service integration bus link in the Billing and Payment profile

1   In the Billing and Payment profile, select Service integration, Buses, eBillingServiceBus, Messaging engines, ebilling_node.server1-eBillingServiceBus, Service integration bus links.

2   Check the color of the arrow in the Status column for eBillingApp_Link.

   If the color of the arrow is dark green, the link status is ON and the connection is running; if it is pale green, the connection has not been made.

Once you have verified the service integration bus link in the Billing and Payment profile, verify the messaging engine link in the Command Center profile.

To verify the messaging engine link in the Command Center profile

1   Stop and start the Command Center profile.

2   Select Service integration, buses, CCServiceBus, Messaging engines.

3   Check the color of the arrow in the Status column for the CC_node.server1-CCServiceBus bus.

   If the color of the arrow is dark green, the link status is ON and the connection is running; if it is pale green, the connection has not been made.

Once you have verified the messaging engine link in the Command Center profile, verify the service integration bus link in the Command Center profile.

To verify the service integration bus link in the Command Center profile

1   In the Command Center profile, select Service integration, Buses, CCServiceBus, Messaging engines, CC_node.server1-CCServiceBus, Service integration bus links.

2   Check the color of the arrow in the Status column for the eBillingApp_Link bus link.

   If the color of the arrow is dark green, the link status is ON and the connection is running; if it is pale green, the connection has not been made.

Modifying the Configuration XMA File for the Billing and Payment Application

Follow these steps to modify the configuration for the notification mail server.

This task is a step in .

To modify the configuration XMA file for the Billing and Payment application

1 Open the notification.xma.xml file, found in the *EDX_HOME*/xma/config/com/edocs/common/ notification folder (or the *EDX_HOME*\xma\config\com\edocs\common\notification folder on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Find the bean ID in the TrueTransporterBean. Modify the mail server properties to point to your company mail server:

> "mail.host: Fully qualified IP address or name of a host running the SMTP which can be used to send email.

> "mail.transport.protocol: Specifies the default message transport protocol.

3 If your company mail server requires mail account authentication, set the following properties. If not, set the mail.smtp.auth property to false, or remove the three properties:

> "mail.smtp.auth: If true, attempt to authenticate the user. Need set a mail account registered in your mail server.

> "mail.user: If mail.smtp.auth is true, set the user name to use when connecting to the mail server.

> "mail.password: If mail.smtp.auth is true, set the user password to use when connecting to the mail server.

4 If your company mail server requires SSL connection, set the mail.smtp.socketFactory.class property. If not, then remove the property:

> "mail.smtp.socketFactory.class: If set, specifies the name of a class that implements the javax.net.SocketFactory interface. This class will be used to create SMTP sockets. For SSL connection, please set to javax.net.ssl.SSLSocketFactory

Example of a TrueTransporterBean:

```
<bean id="TrueTransporterBean"
class="com.edocs.common.notification.extensions.TrueTransporter"
scope="singleton">

<property name="template">

            <ref bean="XSLTemplateBean"/>

    </property>

<property name="mailProperties">
```

```
<props>

    <!-- For non authentication mail server

    <prop key="mail.host">mail.oracle.com</prop>

    <prop key="mail.transport.protocol">SMTP</prop>

     -->

    <prop key="mail.host">mailserver.mycompany.com</prop>

    <prop key="mail.transport.protocol">SMTP</prop>

    <!-- For requiring authentication mail server -->

    <prop key="mail.smtp.auth">true</prop>

    <prop key="mail.user"> emailaccount@mycompany.com </prop>

    <prop key="mail.password"> emailAccountPassword </prop>

    !-- For SSL connection mail server-->

    <prop
key="mail.smtp.socketFactory.class">javax.net.ssl.SSLSocketFactory</prop><prop
key="mail.smtp.socketFactory.port">465</prop>


    </props>

  </property>

  </bean>
```

5 If your company mail server does not use the SMTP default port 25, set the
mail.smtp.socketFactory.port property, indicating which port to use with the specified socket
factory.

6 If your company mail server requires an SSL connection, follow "Exporting and Linking the PayPal
Payflow Pro Certificate (AIX Only)" on page 153 to install your mail server certification into IBM
WebSphere.

7 If your mail server has properties not shown in the default properties list, see *Administration
Guide for Oracle Self-Service E-Billing* and *Implementation Guide for Oracle Self-Service E-Billing*
for details.

The following procedure describes how to modify the configuration in the globalConfig.xma.xml file.

To modify the configuration in the globalConfig.xma.xml file

1 Open the globalConfig.xma.xml file, found in the *EDX_HOME*/xma/config/modules folder (or the
EDX_HOME\xma\config\modules directory on Windows).

2 Find the bean ID called globalConfig. Modify the properties.

Property	Value
ebillingHostName	The correct server name or IP address where the Billing and Payment application is deployed.
csrHostName	The correct server name or IP address where the Customer Service Representative application is deployed.
ebillingApplicationName	The Billing and Payment application name.
csrApplicationName	The Customer Service Representative application name.
ebillingHttpPort	The Billing and Payment application HTTP port.
ebillingSSLPort	The Billing and Payment application SSL port.
csrhttpPort	The Customer Service Representative application HTTP port.
csrSSLPort	The Customer Service Representative application SSL port.

For example:

```
<beans>

  <bean id="globalConfig"
class="com.edocs.common.configuration.core.GlobalConfig" scope="singleton">

        <property name="encryptAccountNumbers"><value>true</value></property>

     <!-- default value -->

     <property name="ebillingHostName"><value>sdcbd80a036</value></property>

     <property name="csrHostName"><value>sdcbd80a036</value></property>


     <property name="ebillingApplicationName"><value>ebilling</value></property>

     <property name="csrApplicationName"><value>ebillingcsr</value></property>


     <property name="ebillingHttpPort"><value>9083</value></property>

     <property name="ebillingSSLPort"><value>9443</value></property>

     <property name="csrhttpPort"><value>9084</value></property>

     <property name="csrSSLPort"><value>9444</value></property>

   </bean>

  </beans>
```

Process of Configuring the IBM WebSphere Application Server for the Command Center Application

To configure the IBM WebSphere Application Server for the Command Center application, complete the following tasks:

1 *"Replacing the Persistence Command" on page 90*

2 *"Replacing the Batch Report Configuration Command" on page 91*

3 *"Creating an IBM WebSphere Profile for the Command Center Application" on page 91*

4 *"Accessing the IBM WebSphere Administration Console for the Command Center Application Server" on page 92*

5 *"Setting Up Application Security" on page 93*

6 *"Securing Access to the IBM WebSphere Administration Console for the Command Center Application" on page 94*

7 *"Enabling HTTPS for the Command Center Application Server" on page 95*

8 *"Setting the JVM Options" on page 95*

9 *"Configuring the JDBC Providers" on page 96*

10 *"Configuring the Data Sources" on page 96*

11 *"Configuring the JMS Resources for the Command Center on IBM WebSphere" on page 98*

12 *"Creating J2C Authentication Aliases" on page 101*

13 *"Setting Up Last Participation Support" on page 102*

14 *"Setting Up Last Participation Support" on page 102*

This process is a step in *"Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75*.

Replacing the Persistence Command

Before you can create the Command Center IBM WebSphere profile you must replace the Persistence command.

This task is a step in *"Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90*.

To replace the Persistence command

■   Replace the *persistence.xma.xml* in the *EDX_HOME*\xma\config\modules\file directory with the one in the *EDX_HOME*\xma\config\modules\websphere\cba directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

Replacing the Batch Report Configuration Command

Before you can create the Command Center IBM WebSphere profile you must replace the batch report configuration command.

This task is a step in

To replace the batch report configuration command

■ Replace the reporting.batch.xma.xml file found in the
  EDX_HOME\xma\config\com\edocs\common\reporting directory with the one in the
  EDX_HOME\xma\config\modules\websphere\ directory.

Creating an IBM WebSphere Profile for the Command Center Application

You must create an IBM WebSphere profile for the Command Center application.

This task is a step in

Creating the Command Center Properties File
The profile for the Command Center application calls a properties file, cc.props, which you must first create.

To create the Command Center properties file

1 Create a file called cc.props. The file can reside in any directory.

2 Add the following text to the file:

        WC_defaulthost=9082

        WC_adminhost=9062

        WC_defaulthost_secure=9445

        WC_adminhost_secure=9045

        BOOTSTRAP_ADDRESS=2811

        SOAP_CONNECTOR_ADDRESS=8882

        SAS_SSL_SERVERAUTH_LISTENER_ADDRESS=9407

        CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS=9409

        CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS=9408

        ORB_LISTENER_ADDRESS=9102

```
DCS_UNICAST_ADDRESS=9355

SIB_ENDPOINT_ADDRESS=7278

SIB_ENDPOINT_SECURE_ADDRESS=7288

SIB_MQ_ENDPOINT_ADDRESS=5560

SIB_MQ_ENDPOINT_SECURE_ADDRESS=5580
```

Creating a Profile for the Command Center Application

Once you have created the properties file you can create the IBM WebSphere profile for the Command Center application.

To create a profile for the Command Center application

1 From the command line, go to the *App_Server_Root*/bin directory, where *App_Server_Root* is your IBM WebSphere installation home path.

2 Use the wasprofile command to create IBM WebSphere profile:

```
./wasprofile.sh

-create

-profileName CC_profile

-profilePath "App_Server_Root/profiles/CC_profile"

-templatePath "App_Server_Root/profileTemplates/default"

-nodeName CC_node

-cellName CC_cell

-hostName localhost

-portsFile "/export/home/cc.props"
```

NOTE: Make sure the cc.props file is located in the directory you specify.

3 List the profile to make sure it was created successfully:

```
./wasprofile.sh -listProfiles
```

Accessing the IBM WebSphere Administration Console for the Command Center Application Server

Once you have created the Command Center profile you can start the Command Center application server and access the IBM WebSphere Administration Console.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90.

To access the IBM WebSphere Administration Console

1 Run the StartServer script, located in the *App_Server_Root*/profiles/*Profile_Name*/bin directory:

`./startServer.sh` *Server_Name*

where:

■ *App_Server_Root* is your IBM WebSphere installation home path.

■ *Profile_Name* is the name of the Command Center profile.

■ *Server_Name* is the name of the server with the Command Center domain.

For example:

`./startServer.sh server1`

2 Access the following URL

http://*Host_Name*:*WC_adminhost*/ibm/console.

where:

■ *Host_Name* is the name of the Command Center application server.

■ *WC_adminhost* is the port number, found in the cc.props file.

Setting Up Application Security

After starting the IBM WebSphere server you must set up application security.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90.

To set up application security

1 Log on to the IBM WebSphere Administration Console on the Command Center application server.

2 Select Security, Secure administration, applications, and infrastructure.

3 Click the Security Configuration Wizard button.

4 Select Enable application security, then click Next.

5 Select Federated repositories, then click Next.

6 Enter the following option values:

Configure Security Option	Value
Enable administrative security	True
Enable application security	True
Use Java 2 security to restrict application access to local resources	False

Configure Security Option	Value
User repository	Federated repositories
Primary administrative user name	websphere

7 Enter the password websphere, and enter it again when prompted to confirm. Click Next, then click Finish. Click Save to save to the master configuration.

8 Click RMI/IIOP security (under Authentication), then click the CSIv2 inbound authentication link.

9 Select Never under both the Basic authentication and Client certificate authentication sections. Click Apply, then click Save to save the master configuration

Once you have set up application security you must stop and start the Command Center application server.

To stop and restart the Command Center application server

1 Run the StopServer script, located in the *App_Server_Root*/profiles/*Profile_Name*/bin directory:

> `./stopServer.sh` *Server_Name*

where:

- ■ *App_Server_Root* is your IBM WebSphere installation home path.
- ■ *Profile_Name* is the name of the Command Center profile.
- ■ *Server_Name* is the name of the Command Center application server.

For example: `./stopServer.sh server1`

2 Run the StartServer script, located in the same directory:

> `./startServer.sh` *Server_Name*

Securing Access to the IBM WebSphere Administration Console for the Command Center Application

Once you have set up application security you must secure access to the IBM WebSphere Administration console.

This task is a step in

To secure access to the IBM WebSphere Administration console

- ■ Enter the URL

  https:*//Host_Name*:*WC_adminhost_secure*/ibm/console

  where:

■ *Host_Name* is the name of the Command Center application server.

■ *WC_adminhost_secure* is the port number, which can be found in the cc.props file.

Enabling HTTPS for the Command Center Application Server

Follow these steps to enable HTTPS on your server for the Command Center application, required for compliance with the Payment Card Industry Data Security Standard.

This task is a step in

To enable HTTPS on your server for the Command Center application

1 Log on to the IBM WebSphere console, for example:

   https://localhost:9043/ibm/console/

2 Click Servers, then Application Servers. In the Application servers table, click the server where you want to deploy your application.

3 Under the Configuration tab, click Web Container Settings, then click Web Container Transport Chains.

4 Enable WCInboundDefault and WCInboundDefaultSecure. Click Apply.

Setting the JVM Options

You must set the JVM options for running Command Center on IBM WebSphere.

This task is a step in

To set the JVM options

1 Select Servers, Application servers, server1, Java and Process Management, Process Definition, and Java Virtual Machine.

2 In the Classpath properties, enter the full path to *EDX_HOME*/config, such as /export/home/qa1/ebilling601/config.

3 In Generic JVM arguments properties, enter the following values, where *EDX_HOME* is the Oracle Self-Service E-Billing installation directory:

   -Dedx.home=*EDX_HOME*

   -Dlog4j.configuration=file:*EDX_HOME*/config/log4j_cc.xml

Configuring the JDBC Providers

You must configure two JDBC providers for the Command Center application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90.

To configure the JDBC providers

1   Select Resource, JDBC, and JDBC Provider.

2   Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

■   *Node_Name* is the Command Center application server node.

■   *Server_Name* is the name of the Command Center application server.

3   Click New to create a new JDBC provider.

4   You must create two JDBC providers as shown in the following table. Specify the data for the first provider.

Database Type	Provider Type	Implementation Type	Name
Oracle	Oracle JDBC Driver	XA data source	Oracle JDBC Driver (XA)
Oracle	Oracle JDBC Driver	Connection pool data source	Oracle JDBC Driver

5   Click Next and enter database class path information.

For Oracle Database 10*g*, specify the directory location for the ojdbc14.jar file. You can find this file in the *EDX_HOME/*lib directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

6   Click Next to view a summary of the provider you just created, click Finish, and then click Save.

7   Repeat from Step 3 to create the second JDBC provider.

Configuring the Data Sources

You must configure six data sources for the Command Center application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90.

To configure the data sources

1 Select Resource, JDBC, and Data Source.

2 Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

■ *Node_Name* is the Command Center application server node.

■ *Server_Name* is the name of the Command Center application server.

3 Click New, then click the Create a new J2C authentication alias link. You must create J2C authentication aliases to store the user name and password of the OLTP and OLAP databases.

4 Click New, then click the Create a J2C authentication alias to store the user name and password of the OLTP database. In the Alias field enter OLTP, enter oltp for the user ID and password, and enter OLTP User for the description.

The user ID and password are the database schema user ID and password the alias uses to connect to the database. Click Apply, then click Save to save changes directly to the master configuration.

5 Repeat from Step 3 to create a J2C authentication alias for the OLAP database. In the Alias field enter OLAP, enter olap for the user ID and password, and enter OLAP User as the description.

6 Select Resource, JDBC, and Data Source.

7 Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

■ *Node_Name* is the Command Center application server node.

■ *Server_Name* is the name of the Command Center application server.

8 Click New. You must create six data sources for the Command Center application as shown in the following table. Specify the data source name (edxAdminDataSource), JNDI name (edx.databasePool), and J2C authentication alias (OLTP) for the first data source.

Data Source Name	JNDI Name	J2C Alias	JDBC Provider
edxAdminDataSource	edx.databasePool	OLTP	Oracle JDBC Driver
edxLoggerDataSource	edx.logger.databasePool	OLTP	Oracle JDBC Driver
edxMessagingDataSource	edx.messaging.databasePool	OLTP	Oracle JDBC Driver (XA)
edxXMADataSource	edx.xma.databasePool	OLTP	Oracle JDBC Driver (XA)

Data Source Name	JNDI Name	J2C Alias	JDBC Provider
edxUserDataSource	edx.user.databasePool	OLTP	Oracle JDBC Driver
reportDataSource	edx.report.databasePool	OLAP	Oracle JDBC Driver (XA)

9 Click Next, select the Select an existing JDBC provider option, and then select the appropriate option from the list.

10 Click Next, and specify the URL for the database in the form:

jdbc:oracle:thin:@*Host_Name:Port_Name:Database*

where:

■ *Host_Name* is the name of the Command Center application server.

■ *Port_Name* is the name of the Command Center application server port.

■ *Database* is the name of the Command Center database SID.

For example:

■ For OLTP: jdbc:oracle:thin:@sdcbd80a036:1521:oltp10g

■ For OLAP: jdbc:oracle:thin:@sdcbd80a036:1521:olap10g

11 Select Oracle 10g data store helper as the data store helper class name. The data source helper must be consistent with the database that you connect to. For example, if the current database version you connect to is Oracle Database 11*g*, then choose Oracle 11g data store helper as the data store helper class name.

12 Click Next to view a summary of the data source you just created, click Finish, and then click Save.

13 To test the newly created data source, select the data source, then click Test connection.

14 Repeat from Step 8 to create the remaining data sources.

Configuring the JMS Resources for the Command Center on IBM WebSphere

You must configure the following JMS resources for the Command Center application on IBM WebSphere:

■ Bus

■ Bus member

■ Bus destination for loggerServiceBus

■ Connection factory

■ Queue

■ Activation specifications

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Command
Center Application" on page 90.

To configure the bus

1 Select Service integration, then Buses.

2 Under Buses, click New.

3 Create one bus, as shown in the following table. Enable security for the loggerService Bus.

Bus Name	Description
loggerServiceBus	Logging

4 Click Next, then click Finish to confirm that you want to create this new bus.

After configuring the buses you must configure a bus members.

To configure the bus member

1 Click on the loggerServiceBus.

2 On the Configuration screen, click Bus Members (under Topology).

3 Click Add to create a new Bus member.

4 Leave the default server value, then click Next.

5 Select File store as the message store, then click Next.

6 Set the message store properties. Leave the default values, and click Next unless you want to
specify a parameter store directory path. Click Finish, then click Save.

After configuring the bus members, you must configure the security for loggerServiceBus.

To configure security for loggerServiceBus

1 Click on loggerServiceBus.

2 On the Configuration screen, click Security (Additional Properties).

3 Click Users, and Groups in the bus connector role (Additional Properties).

4 Click New, select All Authenticated as the Bus Connector Role, and then click OK.

5 Click Save.

After configuring the security, you must configure the bus destination for loggerServiceBus.

To configure the bus destination for loggerServiceBus

1 Select Service integration, Buses, loggerServiceBus.

2   Click Destinations (under Destination resources).

3   Click New, select Queue as the destination type, and then click Next. Enter loggerQueue as the
identifier, then click Next. Leave the default bus member value, click Next, click Finish, and then
click Save.

After configuring bus destinations for loggerServiceBus, you must configure the connection factory.

To configure the connection factory

1   Select Resources, JMS, Queue Connection Factory.

2   Select Node=CC_node.Server=server1, then click New.

3   Select Default messaging provider, then click OK.

4   You must configure a connection factory as shown in the following table. Specify the name of the
connection factory, and the corresponding JNDI and bus names.

Connection Factory Name	JNDI Name	Bus Name
LoggerConnectionFactory	edx/lcf	loggerServiceBus

5   Specify the provider endpoints using the following format:

Host_Name:*Port_Number*:*BootstrapBasicMessaging*

where:

■   *Host_Name* is the Command Center profile host name you defined.

■   *Port_Number* (SIB_ENDPOINT_ADDRESS) of the Command Center profile, such as *7278*.

■   *BootstrapBasicMessaging* is the messaging name.

6   Click Apply, then click Save.

After configuring the connection factories, you must configure the queue.

To configure the queue

1   Select Resource, JMS, Queues.

2   Select Node=CC_node.Server=server1 as the scope.

3   Click New. Select Default messaging provider, and click OK.

4   You must configure an event handler queue as shown in the following table. Enter the name of
the queue, the corresponding JNDI, bus, and queue name.

Name	JNDI Name	Bus Name	Queue Name
loggerQueue	edx/queue/logger	loggerServiceBus	loggerQueue

5   Click Apply, then click Save.

After configuring the queues, you must configure three activation specifications, one for each queue.

To configure the activation specifications

1 Select Resources, JMS, Activation Specifications.

2 Select Node=CC_node.Server=server1 as the scope.

3 Click New. Select Default messaging provider, and click OK.

4 You must configure a queue activation specification as shown in the following table. Enter the name of the queue activation, corresponding JNDI, destination JNDI, and bus name.

Activation Name	JNDI Name	Destination JNDI Name	Bus Name
LoggerActivation	jms/Logger	edx/queue/logger	loggerServiceBus

5 Click OK, and click Save.

Creating J2C Authentication Aliases

You must create two J2C authentication aliases for the EJB modules Logger and Events.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Command Center Application" on page 90.

To create J2C authentication aliases

1 Select Security, Secure administration, application, infrastructure, Authentication, Java Authentication and Authorization Services, J2C authentication data.

2 Click New, and enter the name of the first alias, Logger_ASAuthAlias, and the corresponding User ID, Password, and description.

Alias	User ID	Password	Description
Logger_ASAuthAlias	websphere	websphere	CC admin console user name and password
MessageDispatcher_ASAuthAlias	websphere	websphere	CC admin console user name and password

3 Click Apply, then click Save to save to the master configuration.

4 Repeat from Step 2 for the remaining authentication alias.

Setting Up Last Participation Support

Complete the following procedure to set up last participation support in the Command Center if it is processing two-phase commit global transactions.

Last participant support is an extension to the transaction service that makes it possible for a single one-phase resource to participate in a two-phase transaction with one or more two-phase resources.

This task is a step in

To set up last participation support for the Command Center application

1   Log on to the IBM WebSphere console.

2   Select Applications, Enterprise Applications.

3   Select the command center application.

4   Under Detail Properties, click Last participant support extension.

5   Select Accept heuristic hazard, click Apply, and then click Save changes.

6   Restart the IBM WebSphere server.

Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application

This topic describes the tasks necessary to configure IBM WebSphere for the Customer Service Representative application.

Complete the following procedures to configure IBM WebSphere Application Server for the Customer Service Representative application:

1

2

3

4

5

6

7

8 "Setting Up the Web Container Filter for the Customer Service Representative Application in IBM WebSphere Version 6.1.0.3 and Higher" on page 110

9 "Setting Up the Log4j Output for the Customer Service Representative Application in IBM WebSphere Version 6.1.0.3 and Higher" on page 111

This process is a step in "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75.

Creating an IBM WebSphere Profile for the Customer Service Representative Application

You must create an IBM WebSphere profile for the Customer Service Representative application.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102.

Creating the Customer Service Representative Properties File

The Customer Service Representative profile calls a properties file, csr.props, which you must first create.

To create the Customer Service Representative properties file

1 Create a file called csr.props. The file can reside in any directory.

2 Add the following text to the file:

```
WC_defaulthost=9081

WC_adminhost=9061

WC_defaulthost_secure=9444

WC_adminhost_secure=9044

BOOTSTRAP_ADDRESS=2810

SOAP_CONNECTOR_ADDRESS=8881

SAS_SSL_SERVERAUTH_LISTENER_ADDRESS=9404

CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS=9406

CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS=9405

ORB_LISTENER_ADDRESS=9101

DCS_UNICAST_ADDRESS=9354

SIB_ENDPOINT_ADDRESS=7277

SIB_ENDPOINT_SECURE_ADDRESS=7287
```

```
SIB_MQ_ENDPOINT_ADDRESS=5559

SIB_MQ_ENDPOINT_SECURE_ADDRESS=5579
```

Creating a Profile for the Customer Service Representative Application

Once you have created the properties file you can create the IBM WebSphere profile for the Customer Service Representative application.

To create a profile for the Customer Service Representative application

1 From the command line, go to the *App_Server_Root*/bin directory. In the path, *App_Server_Root* is your IBM WebSphere installation directory.

2 Use the wasprofile command to create an IBM WebSphere profile, where *App_Server_Root* is your IBM WebSphere installation directory:

```
./wasprofile.sh

-create

-profileName CSR_profile

-profilePath "App_Server_Root/profiles/CSR_profile"

-templatePath "/App_Server_Root/profileTemplates/default"

-nodeName CSR_node

-cellName CSR_cell

-hostName localhost

-portsFile "/export/home/csr.props"
```

NOTE: Make sure the csr.props file is located in the directory you specify.

3 List the profile to make sure it was created successfully:

```
./wasprofile.sh -listProfiles
```

Accessing the IBM WebSphere Administration Console on the Customer Service Representative Application Server

Once you have created the Customer Service Representative profile you can start the Customer Service Representative application server and access the IBM WebSphere Administration Console.

This task is a step in

To access the IBM WebSphere Administration Console

1 Run the StartServer script, located in the *App_Server_Root*/profiles/*Profile_Name*/bin directory:

`./startServer.sh` *Server_Name*

where:

■ *App_Server_Root* is your IBM WebSphere installation home path.

■ *Profile_Name* is the name of Billing and Payment profile.

■ *Server_Name* is the name of the Billing and Payment application server.

For example:

`./startServer.sh server1`

2 Access the URL

`http://`*Host_Name*`:`*WC_adminhost*`/ibm/console`.

where:

■ *Host_Name* is the name of the server where you installed the Customer Service Representative application.

■ *WC_adminhost* is the port number, which you can find in the csr.props file.

For example:

`http://sdcb80a036.siebel.com:9062/ibm/console`

NOTE: To log in, click the Log in button.

Setting Up Application Security

After starting the IBM WebSphere server you must set up application security.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102.

To set up application security

1 Log on to the IBM WebSphere Administration Console on the Customer Service Representative server.

2 Select Security, and select Secure administration, applications, and infrastructure.

3 Click the Security Configuration Wizard button.

4 Select Enable application security, and click Next.

5 Select Federated repositories, and click Next.

6 Enter the password websphere, and enter it again when prompted to confirm.

7 Enter the following option values.

Configure Security Option	Value
Enable administrative security	True
Enable application security	True
Use Java 2 security to restrict application access to local resources	False
User repository	Federated repositories
Primary administrative user name	websphere

8 Click Next, then click Finish.

9 Click Save to save to the master configuration.

10 Click RMI/IIOP Security (under Authentication,) and click the CSIv2 inbound authentication link.

11 Select Never under both the Basic authentication and Client certificate authentication sections. Click Apply, then click Save to save the master configuration.

Once you have set up application security you must stop and start the Customer Service Representative application server.

To stop and restart the Customer Service Representative application server

1 Run the StopServer script, located in the *App_Server_Root*/profiles/*Profile_Name*/bin directory:

> `./stopServer.sh` *Server_Name*

where:

- ■ *App_Server_Root* is your IBM WebSphere installation home path.
- ■ *Profile_Name* is the name of the Command Center profile.
- ■ *Server_Name* is the name of the Command Center application server.

For example: `./stopServer.sh server1`

2 Run the StartServer script, located in the same directory:

> `./startServer.sh` *Server_Name*

Securing Access to the IBM WebSphere Administration Console for the Customer Service Representative Application

Once you have set up application security you must secure access to the IBM WebSphere Administration console.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102.

To secure access to the IBM WebSphere Administration console

■ Enter the following URL:

https://*Host_Name*:*WC_adminhost_secure*/ibm/console

where:

- ■ *Host_Name* is the name of the Customer Service Representative application server.

- ■ *WC_adminhost_secure* is the port number, which can be found in the csr.props file.

Enabling HTTPS on Your Server for the Customer Service Representative Application

Follow these steps to enable HTTPS on your server for the Customer Service Representative, required for compliance with the Payment Card Industry Data Security Standard.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102.

To enable HTTPS on your server for the Customer Service Representative application

1 Log on to the IBM WebSphere console, for example

   https://localhost:9043/ibm/console/

2 Click Servers, then Application Servers. In the Application servers table, click the server where you want to deploy your application.

3 Under the Configuration tab, Click Web Container Settings, then click Web container transport chains.

4 Enable WCInboundDefault and WCInboundDefaultSecure.

5 Click Apply.

Setting the JVM Options

You must set the JVM options for running the Customer Service Representative application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102.

To set the JVM options

1 Select Servers, Application servers, server1, Java and Process Management, Process Definition, and Java Virtual Machine.

2 In the Classpath properties, enter the full path to the *EDX_HOME*/config directory, such as:

> /export/home/qa1/ebilling601/config

3 In Generic JVM arguments properties, enter the following values, where *EDX_HOME* is the Oracle Self-Service E-Billing installation directory:

> –Dedx.home=*EDX_HOME*

> –Dlog4j.configuration=file:*EDX_HOME*/config/log4j_csr.xml

Configuring the JDBC Providers

You must configure two JDBC providers for the Customer Service Representative application on IBM WebSphere.

This task is a step in

To configure the JDBC providers

1 Select Resource, JDBC, and JDBC Provider.

2 Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

■ *Node_Name* is the Customer Service Representative application server node.

■ *Server_Name* is the name of the Customer Service Representative application server.

3 Click New to create a new JDBC provider. You must create two JDBC providers as shown in the following table. Specify the data for the first provider.

Database Type	Provider Type	Implementation Type	Name
Oracle	Oracle JDBC Driver	XA data source	Oracle JDBC Driver (XA)
Oracle	Oracle JDBC Driver	Connection pool data source	Oracle JDBC Driver

4 Click Next, and enter database class path information.

For Oracle Database 10*g*, specify the directory location for the ojdbc14.jar file. You can find this file in the *EDX_HOME*/lib directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

5 Click Next to view a summary of the provider you just created.

6 Click Finish, then click Save to save directly to the master configuration.

7 Repeat from Step 3 to create the remaining JDBC provider.

Configuring the Data Sources

You must configure three data source for the Customer Service Representative application on IBM WebSphere.

This task is a step in "Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102.

To configure the data sources

1 Select Resource, JDBC, and Data Source.

2 Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

■ *Node_Name* is the Customer Service Representative application server node.

■ *Server_Name* is the name of the Customer Service Representative application server.

3 Click New, and click the Create a new J2C authentication alias link. You must create J2C authentication aliases to store the user name and password of the OLTP and OLAP databases.

4 Click New, and create a J2C authentication alias to store the user name and password of the OLTP database. In the Alias field enter OLAP, enter oltp5 for the User ID and password, and enter OLTP User for the description.

The user ID and password are the database schema user ID and password the alias uses to connect to the database.

5 Click Apply, then click Save to save change directly to the master configuration.

6 Repeat from Step 3 to create a J2C authentication alias for the OLAP database. In the Alias field enter OLAP, enter olap5 for the User ID and password, and enter OLAP User as the description.

7 Select Resource, JDBC, and Data Source. Select the following scope from the list:

Node=*Node_Name*, Server=*Server_Name*

where:

■ *Node_Name* is the Customer Service Representative application server node.

■ *Server_Name* is the name of the Customer Service Representative application server.

8 Click New. You must create three data sources for the Customer Service Representative application as shown in the following table. Specify the data source name (edxAdminDataSource), JNDI name (edx.databasePool), and J2C authentication alias (OLTP) for the first data source.

Data Source Name	JNDI Name	J2C Alias	Transaction
edxAdminDataSource	edx.databasePool	OLTP	Oracle JDBC Driver (XA)
edxXMADataSource	edx.xma.databasePool	OLTP	Oracle JDBC Driver (XA)
reportDataSource	edx.report.databasePool	OLAP	Oracle JDBC Driver (XA)

9 Click Next, select an existing JDBC provider option, and select Oracle JDBC Driver from the list.

10 Click Next, and specify the URL for the database in the form:

jdbc:oracle:thin:@*Host name:Port name:Database*

where:

■ *Host_Name* is the name of the server where the Customer Service Representative application is installed.

■ *Port_Name* is the name of the port where the Customer Service Representative is installed.

■ *Database* is the name of the database SID.

For example:

■ OLTP: jdbc:oracle:thin:@sdcbd80a036:1521:oltp10g

■ OLAP: jdbc:oracle:thin:@sdcbd80a036:1521:olap10g

11 Select Oracle 10g data store helper as the data store helper class name. The data source helper must be consistent with the database that you connect to. For example, if the current database version you connect to is Oracle Database 11*g*, then choose Oracle 11g data store helper as the data store helper class name.

12 Click Next to view a summary of the data source you just created, then click Finish. Click Save to save the new data source directly to the master configuration.

13 To test the newly created data source, select the data source, and click Test connection.

14 Repeat from Step 8 to create the remaining data sources.

Setting Up the Web Container Filter for the Customer Service Representative Application in IBM WebSphere Version 6.1.0.3 and Higher

Complete the following procedure to set up the Web container filter on IBM WebSphere version 6.0.1.3 and higher for the Customer Service Representative application.

This task is a step in *"Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102*.

To set up the Web container filter in IBM WebSphere 6.1.0.3 and higher for the Customer Service Representative application

1 Log on to the IBM WebSphere console.

2 Select Application servers, server1 (csr), Web container, Custom Properties.

3 Create a new property called com.ibm.ws.webcontainer.invokefilterscompatibility. Specify true in the Value field.

4 Restart the server.

Setting Up the Log4j Output for the Customer Service Representative Application in IBM WebSphere Version 6.1.0.3 and Higher

Complete the following procedure to set up log4j output on IBM WebSphere Version 6.1.0.17 and higher for the Customer Service Representative application.

This task is a step in *"Process of Configuring the IBM WebSphere Application Server for the Customer Service Representative Application" on page 102*.

To set up log4j output in IBM WebSphere 6.1.0.17 and higher for the Customer Service Representative application

1 Create a new file called commons-logging.properties in the $WAS_HOME/profiles/CSR_profile/ properties directory.

2 Add the following content to the file:

    org.apache.commons.logging.LogFactory=org.apache.commons.logging.impl.LogFactoryImpl

3 Restart the Customer Service Representative application server.

Process of Deploying Oracle Self-Service E-Billing Applications on IBM WebSphere

You must deploy each of the three J2EE application EAR files using your application server.

Complete the following procedures to deploy each J2EE application:

■ *"Deploying the Billing and Payment Application on IBM WebSphere" on page 112*

■ *"Deploying the Command Center Application on IBM WebSphere" on page 112*

■ "Deploying the Customer Service Representative Application on IBM WebSphere" on page 114

This process is a step in "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75.

Deploying the Billing and Payment Application on IBM WebSphere

Follow these steps to deploy the Billing and Payment application EAR file on IBM WebSphere.

This task is a step in "Process of Deploying Oracle Self-Service E-Billing Applications on IBM WebSphere" on page 111.

To deploy the Billing and Payment application on IBM WebSphere

1 Replace the persistence.xma.xml file found in the *EDX_HOME*\xma\config\modules directory with the same file found in the *EDX_HOME*\xma\config\modules\websphere\cba directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Select Applications, Enterprise Applications, and Install.

3 Select Local file system if your EAR file is on the local computer or select Remote file system, and browse to the full path of the EAR file: *EDX_HOME*/J2EEApps/ebilling/websphere.

4 Click Next, and give an appropriate name to the Billing and Payment application, leaving the settings as they are.

5 Click Next, and select all the modules to map to servers.

6 Map the two EJBs, and give the target resource JNDI names as shown.

Resource Reference	Class	Target Resource JNDI Name
ejb/ PaymentAccountManager	com.edocs.payment.remote. IPaymentAccountManager	edx/ejb/ PaymentAccountManager
ejb/PayServer	com.edocs.payment.remote.IPay Server	edx/ejb/PayServer

7 Select the Billing and Payment application, and click Next. View the summary, then click Finish.

Deploying the Command Center Application on IBM WebSphere

Follow these steps to deploy the Command Center application EAR file on IBM WebSphere.

This task is a step in "Process of Deploying Oracle Self-Service E-Billing Applications on IBM WebSphere" on page 111.

To deploy the Command Center application on IBM WebSphere

1 Select Applications, Enterprise Applications, and Install.

2 Select Local file system if your EAR file is on the local computer or select Remote file system, and browse to the full path of the EAR file: *EDX_HOME*/ebilling603/J2EEApps/commandcenter/ websphere. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Click Next, and give an appropriate name to the Command Center application, leaving the settings as they are.

4 Click Next, and select all the modules to map to servers and click Next.

5 Leave the EJB Logger module options as they are.

6 To bind the listeners for the message-driven enterprise bean to the Activation Specification for the Logger, use the default values for the Target Resource JNDI and Destination JNDI names. For the Activation Spec authentication alias, add the prefix CC_node/ to the default value Logger_ASAuthAlias to create CC_node/Logger_ASAuthAlias.

7 To bind the listeners for the message-driven enterprise bean to the Activation Specification for Events, specify the following values.

Field	Value
Target Resource JNDI Name	jms/MessageDispatcher
Destination JNDI name	edx.queue.outbound
Activation Spec authentication alias	CC_node/ MessageDispatcher_ASAuthAlias

8 Ignore the application resource warnings, and click Continue.

9 Map the empty EJBs whose target Resource JNDI name is blank, and give the target resource JNDI names as shown.

Resource Reference	Class	Target Resource JNDI Name
ejb/AppMgr	com.edocs.services.application.IAppMgr	edx/ejb/AppMgr
ejb/VersionManager	com.edocs.services.versioning.IVersion Manager	edx/ejb/ VersionManager

10 In the Specify authentication method section, check None. Specify the Target Resource JNDI Name as edx.databasePool for the JNDI name, then click Next.

11 On the Map virtual hosts for Web modules screen, select all application names, and click Next.

12 View the summary, then click Finish. Click Save to save directly to the master configuration.

Deploying the Customer Service Representative Application on IBM WebSphere

Follow these steps to deploy the Customer Service Representative application EAR file on IBM WebSphere.

This task is a step in .

To deploy the Customer Service Representative application on IBM WebSphere

1 Replace the persistence.xma.xml file found in the *EDX_HOME*\xma\config\modules directory with the one found in the *EDX_HOME*\xma\config\modules\websphere\cba directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Replace the reporting.batch.xma.xml file found in the *EDX_HOME*\xma\config\com\edocs\common\reporting directory with the same file found in the *EDX_HOME*\xma\config\modules\websphere\ directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Select Applications, Enterprise Applications, and Install.

4 Select Local file system if your EAR file is on the local computer or select Remote file system, and browse to the full path of the EAR file: *EDX_HOME*/J2EEApps/CSR/websphere.

5 Click Next, and give an appropriate name to the Customer Service Representative application, leaving the settings as they are.

6 Click Next, and select all the modules to map to servers.

7 Map the two EJBs, and give the target resource JNDI names as shown.

Resource Reference	Class	Target Resource JNDI Name
ejb/ PaymentAccountManager	com.edocs.payment.remote. IPaymentAccountManager	edx/ejb/ PaymentAccountManager
ejb/PayServer	com.edocs.payment.remote. PayServer	edx/ejb/PayServer

8 Click Next to view the summary, then click Finish. Click Save to save directly to the master configuration.

Configuring and Starting Scheduler on IBM WebSphere

Scheduler manages the Command Center jobs. Scheduler must be configured and running for scheduled jobs to execute.

Follow the procedure appropriate for your operating system to configure Scheduler on Oracle WebLogic:

■ "Configuring and Starting Scheduler on IBM WebSphere and AIX" on page 115

■ "Configuring and Starting Scheduler on IBM WebSphere and Windows" on page 116

NOTE: For information on using Scheduler, see *Administration Guide for Oracle Self-Service E-Billing.*

This process is a step in "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75.

Configuring and Starting Scheduler on IBM WebSphere and AIX

Follow these steps to configure and start Scheduler if you are running IBM WebSphere on AIX.

To configure and start Scheduler on IBM WebSphere and AIX

1 On the command line, go to the *EDX_HOME*/bin directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Run the following command:

$./edx_config

3 Specify the appropriate details for the database server, Oracle home directory, and application server. For example.

Parameter	Example Value
Database server	oracle
Oracle home directory	/vol1/oracle/11.2.0.1
Oracle database user name (DB_Username)	oltp
Oracle database password (DB_PWD)	oltp
Oracle database alias (tnsname)	oltp
Application server to use	ws
Version of WebSphere Application Server	6.1
WebSphere Application Server profile directory	/export/home/IBM/WebSphere/AppServer/profiles/CC_profile
Java root directory	/export/home/IBM/WebSphere/AppServer/java

4 Specify the IBM WebSphere Application Server profile directory:

```
$WS_HOME/AppServer/profiles/CC_profile
```

5   Specify the Java root directory:

```
$WS_HOME/AppServer/java
```

6   Run Scheduler using the following command, located in the *EDX_HOME*/bin directory:

```
./ws_scheduler -start -url iiop://Host_Name:Bootstrap_Address -verbose
```

where:

■   *Host_Name* is the name of the application server.

■   *Bootstrap_Address* is the bootstrap address, which you can find in the cc.props file.

For example:

```
./ws_scheduler -start -url iiop://localhost:2811 –verbose
```

7   To stop Scheduler, replace the -start parameter with the -stop parameter.

```
./ws_scheduler -stop -url iiop://localhost:2811 –verbose
```

Configuring and Starting Scheduler on IBM WebSphere and Windows

Follow these steps to configure and start Scheduler if you are running IBM WebSphere on Windows.

To configure and start Scheduler on IBM WebSphere and Windows

1   Open the edx_env.bat file found in the *EDX_HOME*\config directory, and add the following commands, where *Your_Java_Home* is the directory where Java is installed:

```
@set EDX_HOME=D:\oracle\ebilling
```

```
@set APP_SERVER=ws
```

```
@set WAS_HOME= D:\IBM\WebSphere\AppServer
```

```
@set WAS_VERSION=6.1
```

```
@set JAVA_HOME=Your_Java_Home
```

2   Change following content in the ws_scheduler.bat file found in the *EDX_HOME*\bin directory:

```
SET CMD=%JAVA_HOME%\bin\java %EDX_OPTS% com.ibm.ws.bootstrap.WSLauncher
com.ibm.websphere.client.applicationclient.launchClient
%EDX_HOME%\J2EEApps\websphere\Deployed_ear-eStatement.ear -
CCclasspath=%EDX_CLASSPATH% -CCverbose=true -CCjar=app-scheduler.jar -start

%CMD%
```

Replace it with the following content, where *bootstrapport* is your bootstrap port number:

```
SET CMD=%JAVA_HOME%\bin\java -Dwas.install.root=%WAS_HOME% %EDX_OPTS%
com.ibm.ws.bootstrap.WSLauncher
com.ibm.websphere.client.applicationclient.launchClient
%EDX_HOME%\J2EEApps\websphere\Deployed_ear-eStatement.ear -
CCclasspath=%EDX_CLASSPATH% -CCverbose=true -CCjar=app-scheduler.jar -
CCBootstrapPort=bootstrapport -start

@for /F "tokens=1,2 delims=: " %%i IN ('time/t') do @set vTime=%%i_%%j
@for /F "tokens=1,2 delims= " %%i IN ('date/t') do @set vDate=%%j
@for /F "tokens=1,2,3 delims=/ " %%i IN ("%vDate%") do @set vDate=%%k_%%i_%%j%
@set fName=%vDate%_%vTime%
@set _log="%EDX_HOME%\logs\edx_scheduler_%fName%.log"
%CMD%>>%_log%
```

3 Run the following command:

ws_scheduler.bat -start -url iiop://*Host_Name*:*Bootstrap_Address* -verbose

where:

■ *Host_Name* is the name of the application server.

■ *Bootstrap_Address* is the bootstrap address.

4 To stop Scheduler, replace the -start parameter with the -stop parameter.

./ws_scheduler.bat –stop –url iiop://localhost:2811 –verbose

Running the Sample Oracle Self-Service E-Billing Applications on IBM WebSphere

After successfully deploying the Oracle Self-Service E-Billing applications, you can log in to the sample Oracle Self-Service E-Billing applications.

To run the sample Oracle Self-Service E-Billing applications

1 Select Applications, and Enterprise Applications.

2 Select all applications (Billing and Payment, Command Center, and Customer Service Representative) and click Start. (To stop an application, click Stop.)

3 In your browser, point to the Oracle Self-Service E-Billing application name, shown in the
following table, specifying the server name (*Host_Name)* and port number (*WC_defaultHost)*
where you deployed the particular Oracle Self-Service E-Billing application.

Sample Application	URL Example
Billing and Payment UI	http://*Host_Name*:*WC_defaultHost*/ebilling
Command Center	http://*Host_Name*:*WC_defaultHost*/eBilling
Customer Service Representative UI	http://*Host_Name*:*WC_defaultHost*/ebillingCSR

The sample log in page appears.

4 For the sample Billing and Payment application, log in as an administrator using the one of the
following admin or user roles.

Note that the user you log in as determines whether you see the Business (B2B) or Consumer
(B2C) Edition of the Billing and Payment application. For information about enrolling for the first
time, see *Application Guide for Oracle Self-Service E-Billing (Business Edition)* or *Application
Guide for Oracle Self-Service E-Billing (Consumer Edition)*.

User Name	Password	Role
jsmith	jsmith	Admin
ftown	ftown	B2B Admin User
twalsh	twalsh	B2C User
tbrown	tbrown	B2C User

6 Installing the ETL Module for Oracle Self-Service E-Billing

This chapter describes how to install and deploy the Extract Transform Loading (ETL) module into the Oracle Self-Service E-Billing production environment. It includes the following topics:

- Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing on page 119
- Verifying ETL Module System Requirements on page 119
- Installing the Temporary Patch for Oracle Warehouse Builder 11g on page 120
- Creating the Oracle Workflow Manager on page 121
- Process of Installing the Oracle Warehouse Builder Repository on page 122
- Installing the ETL Module on page 131
- Granting File Location Privileges to MAP_USER for Oracle Database 11g Release 2 on page 134
- Running the ETL Loader Job Using Sample Data on page 135

Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing

To install the ETL Module on a correctly configured Oracle Self-Service E-Billing installation, perform the following processes and tasks:

1 Verifying ETL Module System Requirements on page 119

2 Installing the Temporary Patch for Oracle Warehouse Builder 11g on page 120

3 Creating the Oracle Workflow Manager on page 121

4 Process of Installing the Oracle Warehouse Builder Repository on page 122

5 Installing the ETL Module on page 131

6 Running the ETL Loader Job Using Sample Data on page 135

Related Topics
"Roadmap for Installing Oracle Self-Service E-Billing" on page 17

Verifying ETL Module System Requirements

This guide assumes you are installing the Oracle Self-Service E-Billing ETL Module on a correctly configured Oracle Self-Service E-Billing installation and that you have installed the required software.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119.

To verify that you have the correct ETL Module system requirements

■ For your version of Oracle Database, verify that your system has the necessary software installed:

  ■ Installations on Oracle Database 10*g* Release 2 require the following software:

    ❏ Oracle RDBMS 10*g* 10.2.0.2

    ❏ Oracle Warehouse Builder 10.2.0.2 or 10.2.0.1 with Oracle Warehouse Builder patches 5384468 and 5696353. Installation packs are downloadable from My Oracle Support.

    ❏ Oracle Workflow 2.6.4. Install this using the Oracle Companion CD-ROM from the Oracle Database 10*g* (10.2.0.2) companion CD-ROM. Installation packs are downloadable from My Oracle Support.

    ❏ Apache Ant 1.6.5

  ■ Installations on Oracle Database 11*g* Release 2 require the following software:

    ❏ Oracle Database 11*g* 11.2.0.1.

    ❏ Oracle Warehouse Builder 11g Release 2 and Oracle Workflow; both products are automatically included with the Oracle Database installation.

    ❏ Patch 9300150 for Oracle Warehouse Builder 11g Release 2.

Installing the Temporary Patch for Oracle Warehouse Builder 11*g*

If you created the repository using Oracle Warehouse Builder 11*g*, follow this procedure to install the temporary patch. The temporary patch is not required for Release 2 of Oracle Warehouse Builder 11*g*.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119.

To install the temporary patch for Oracle Warehouse Builder 11g

1 Go to the *EDX_HOME*/db/ebilling/oracle/olap/etl/packages (or the *EDX_HOME*\db\ebilling\oracle\olap\etl\packages folder on Windows), and verify that the BUG7497106.sql file exists in that folder.

2 Log in to the OLAP database using SQL*Plus as sysdba, where *OLTP_Instance* is the name of the OLTP instance:

    export ORACLE_SID=*OLTP_Instance*

    SQLPLUS "/ as sysdba"

3 At the SQL*Plus prompt, run:

```
SQL>@BUG7497106.sql

SQL>exit

SQL>exit
```

Creating the Oracle Workflow Manager

You must create and configure an Oracle Workflow Manager.

This task is a step in

To create the Oracle Workflow Manager

1 Go to the Workflow directory.

2 Run the Oracle Workflow Configuration Assistant.

NOTE: The UNIX Repository Assistant requires a correctly configured X Window environment to run.

On an Oracle Database 10g database:

■ **UNIX.** Log in as the Oracle user, change the directory to `$ORACLE_HOME/wf/install`, and run the csh `wfinstall.csh script`.

■ **Windows.** Log in as an administrator, change the directory to `ORACLE_HOME\wf\install`, and click `wfinstall.bat`.

On an Oracle Database 11g database:

■ **UNIX.** Log in as the Oracle user, change the directory to `ORACLE_HOME/owb/wf/install`, and run the csh `wfinstall.csh script`.

■ **Windows.** Log in as an administrator, change the directory to `ORACLE_HOME\owb\wf\install`, and click `wfinstall.bat`.

3 Enter following values.

Field	Value
Install Option	Server Only
Workflow Account	owf_mgr
Workflow Password	Enter a workflow password, such as owf_mgr.
SYS Password	Enter the system DBA password.
TNS Connect Descriptor	Format: HostName:Portno:olap_sid
	For example, if the host name is sdc50010qe.us.oracle.com, Port no of the OLAP database is 1521, and Olap_sid is olap, the format would be: Sdc50010qe.us.oracle.com:1521:olap.

4 Use default values for the remaining parameters.

5 Click Submit. You can monitor the status of the installation process by viewing the output on your UNIX console.

6 When installation completes, a Workflow Configuration has completed successfully message appears. Click OK.

Process of Installing the Oracle Warehouse Builder Repository

To install the Oracle Warehouse Builder Repository you must perform the following procedures:

1 "Tuning the OLAP Database" on page 122

2 "Creating the Oracle Warehouse Builder Repository Owner" on page 123

3 "Creating the Oracle Warehouse Builder Repository User" on page 125

4 "Registering Users in Oracle Warehouse Builder 11g" on page 126

5 "Creating Database Modules" on page 127

6 "Creating Database Modules" on page 127

7 "Creating a Process Flow Module" on page 128

8 "Creating File System Locations" on page 129

9 "Registering Locations" on page 129

10 "Granting Rights to MAP_USER" on page 130

This process is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119.

Tuning the OLAP Database

There are several important database parameters you must tune for Oracle Warehouse Builder and the ETL Module to function properly.

Verify that the values set are appropriate for your operating system and database environment.

This task is a step in "Process of Installing the Oracle Warehouse Builder Repository" on page 122.

To verify the database parameters

1 Open the init*SID*.ora initialization file found in the $ORACLE_HOME/dbs directory (in the %ORACLE_HOME%\database directory for Windows) for the Oracle Self-Service E-Billing OLAP database. In the file name, *SID* is your OLAP database SID. Verify that these parameters are set appropriately for your organization. The following values are the minimum requirement:

   ■ db_cache_size=256m

- large_pool_size=64m

- pga_aggregate_target=512m

- shared_pool_size=500m

- job_queue_processes=10

- aq_tm_processes=1

- o7_dictionary_accessibility=true

2 Restart the OLAP database.

The database is now ready for installation of the Oracle Warehouse Builder repository schema.

Creating the Oracle Warehouse Builder Repository Owner

You must create an Oracle Warehouse Builder repository owner using the Oracle Warehouse Builder Repository Assistant. For additional information about the Repository Assistant, see the Oracle Warehouse Builder documentation on Oracle Technology Network.

Follow the appropriate procedure for your version of the Oracle Warehouse Builder.

This task is a step in .

To create the Oracle Warehouse Builder repository owner using Oracle Warehouse Builder 10g Release 2

1 Start the Oracle Warehouse Builder Repository Assistant.

- **UNIX.** Log in as the oracle user, change the directory to OWB_HOME/bin/unix, and run the ./ reposinst.sh script.

  NOTE: The Oracle Solaris and Linux Repository Assistant requires a correctly configured X Window environment in order to run.

- **Windows.** Log in as an administrator. On the Start Menu, click the link to the Repository Assistant (Administration).

2 On the Welcome screen click Next.

3 Enter the host name, port number, and service name for the Oracle Self-Service E-Billing OLAP database, and click Next.

4 Select Manage a Warehouse Builder repository owner, and click Next.

5 Select Create a new Warehouse Builder repository owner, and click Next.

6 Enter repository owner user name rep_admin and a password, and click Next.

7 Re-enter the password, and click OK.

8 Enter the following tablespace information:

- Tablespace for Data: EDX_REPORT_CDR_DATA

- Tablespace for Indexes: EDX_REPORT_CDR_IDX

- Tablespace for Temporary Data: TEMP (Default)

- Tablespace for Snapshots: EDX_REPORT_CDR_DATA

9   Select the language required for the repository. The default is American English. Click Next.

10 Click Next to skip selecting a user.

11 The Repository Assistant Summary displays all the information related to repository. Click Finish.

12 On the Repository Users screen, click OLAP on the Available pane, then click the right arrow key to move it to the Selected pane. Click Next.

13 On the Request Password Information screen enter the password for OLAP. Click OK. The Summary Page appears. Verify the information on this screen.

14 Click Finish. A progress bar shows the installation status.

15 When complete, an Installation Successful message appears. Click OK.

To create the Oracle Warehouse Builder repository owner using Oracle Warehouse Builder 11g

1   Start the Oracle Warehouse Builder Repository Assistant.

- **UNIX.** Log in as the Oracle user, change the directory to `OWB_HOME/bin/unix`, and run the `./reposinst.sh script`.

   NOTE: The Oracle Solaris and Linux Repository Assistant requires a correctly configured X Window environment in order to run.

- **Windows.** Log in as an administrator. On the Start Menu, click the link to the Repository Assistant (Administration).

2   On the Welcome screen, click Next.

3   Enter the host name, port number, and service name for the Oracle E-Billing OLAP database, and click Next.

4   Select Manage Warehouse Builder workspaces, and click Next.

5   Select Create a new Warehouse Builder workspace, and click Next.

6   Select Create a workspace with a new user as workspace owner, and click Next.

7   Enter the SYSTEM password as `manager`, and click Next.

8   (Windows Only) If you are creating the first workspace for the repository, click Next to accept the default values on the Enable Optional Features page.

9   Enter the Workspace owner's user name `rep_admin`, the Workspaces Owner password, the Workspace Owner Password Confirmation, and Workspace Name `rep_admin`. Click Next.

10 Enter the OWBSYS password, typically OWBSYS, and click next.

11 Enter the following tablespace information, and click Next:

- Tablespace for Data: EDX_REPORT_CDR_DATA

■ Tablespace for Indexes: EDX_REPORT_CDR_IDX

■ Tablespace for Temporary Data: TEMP (Default)

■ Tablespace for Snapshots: EDX_REPORT_CDR_DATA

12 Click Next.

13 If necessary, select the language required for the repository. The default is American English.
Click Next.

14 Click Next to skip selecting a user.

15 The Repository Assistant Summary displays all the information related to repository. Click Finish.

16 The progress bar shows the installation status. When it finishes, click OK on the Installation
Successful dialog.

17 Click Finish.

Creating the Oracle Warehouse Builder Repository User

You must create an Oracle Warehouse Builder repository user using the Oracle Warehouse Builder
Design Center.

Follow the appropriate procedure for your version of the Oracle Warehouse Builder.

This task is a step in .

To create the repository user using Oracle Warehouse Builder 10g Release 2

1 Start the Oracle Warehouse Builder Design Center.

■ **UNIX.** Log in as the oracle user, change the directory to `OWB_HOME/bin/unix`, and run the `./
owbclient.sh` script.

■ **Windows**. Log in as an administrator, and click the link in the Start Menu to the Design
Center.

If you want to connect to a UNIX-hosted Oracle Warehouse Builder repository you can use
the Windows native client if Oracle Warehouse Builder is installed.

2 Log in to the Design Center as the repository owner by providing the host name, port, and SID
of the OLAP database.

3 After successfully logging in to the Design Center, expand the security tree in the Global
Navigator pane. Right-click the User element, and click New to start the Oracle Warehouse
Builder User Wizard.

4 Click Create DB User.

5 Enter the SYSDBA password, user name map_user (required), and a password for the new
database user. Click OK.

6 Check the Used as target schema box, then click Next.

To create the repository user using Oracle Warehouse Builder 11g

1 Start the Oracle Warehouse Builder Design Center.

- ■ **UNIX.** Log in as the Oracle user, change the directory to `OWB_HOME/bin/unix`, and run the `./owbclient.sh` script.

- ■ **Windows**. Log in as an administrator. On the Start Menu, click the link to the Design Center.

  If you want to connect to a UNIX-hosted Oracle Warehouse Builder repository you can use the Windows native client if Oracle Warehouse Builder is installed.

2 Log in to the Design Center as the repository owner, providing the host name, port, and SID of the OLAP database.

3 After successfully logging in to the Design Center, expand the security tree in the Global Explorer pane. Right-click the User element, and click New to start the Oracle Warehouse Builder User Wizard.

4 Click Create DB User.

5 Enter the SYSTEM password, user name map_user (required), and a password for the new database user. Leave the default values for Tablespace, click OK, and then click Next.

6 (Oracle Database 11*g* Release 2 only) Uncheck the To Create a Location option, and click Next.

7 Click OK to create a user (on Oracle Database 11*g* Release 2 click Finish).

Creating an Oracle Warehouse Builder Project

You must create an Oracle Warehouse Builder project in the Oracle Warehouse Builder Design Center.

This task is a step in *"Process of Installing the Oracle Warehouse Builder Repository" on page 122*.

To create an Oracle Warehouse Builder project

1 Start the Oracle Warehouse Builder Design Center, and log in as the repository owner.

2 On the Design menu, click New to create a new project.

3 Enter the name of the project, such as EBILLING_ETL, and click OK.

Registering Users in Oracle Warehouse Builder 11*g*

You must register the OLAP and OWF_MGR users when creating the repository using Oracle Warehouse Builder 11*g*.

This task is a step in *"Process of Installing the Oracle Warehouse Builder Repository" on page 122*.

To register users in Oracle Warehouse Builder 11g

1 Start the Oracle Warehouse Builder Design Center.

■ **UNIX.** Log in as the Oracle user, change the directory to ORACLE_HOME/owb/bi n/uni x, and
run the ./owbclient.sh script.

■ **Windows.** Log in as an administrator. On the Start Menu, click the link to the Design Center.
If you want to connect to a UNIX-hosted Oracle Warehouse Builder repository, you can use
the Windows native client if Oracle Warehouse Builder is installed.

2 Log in to the Design Center as the repository owner. Select the EBILLING_ETL project.

3 Expand the security tree in the Global Explorer (Global Navigator On Oracle Database 11*g*
Release 2). Right-click the User element, and click New to start the Oracle Warehouse Builder
User Wizard.

4 In the list of Available DB Users, double click OWF_MGR, and OLAP schema names to add them
to the list of selected users. Click OK.

5 (Oracle Database 11*g* Release 2 only) For both OWF_MGR, and OLAP users, uncheck the To
Create a Location option, and click Next.

6 Click Next to create a user (on Oracle Database 11*g* Release 2, click Finish).

Selected users appear in the Global Explorer (Global Navigator On Oracle Database 11*g* Release
2) under the user list.

NOTE: Always re-register the OLAP schema whenever the OLAP schema refresh or reimage is
done. If the OLAP schema is not registered the following error message appears:

ETL_SOURCE_SCANNER:GET_DIRECTORIES_MAPPING

Error ORA-20000: You are not a registered user on workspace: REP_ADMIN, and you cannot
access the public views of the workspace.

ORA-06512: at "OWBSYS.START_ENABLE_OWB_ROLES", line 23

ORA-06512: at line 1

Creating Database Modules

Create the following Oracle database modules using the Oracle Warehouse Builder Design Center:

■ STAGE_TABLES

■ TARGET_TABLES

■ MAPPINGS

■ OLTP

This task is a step in .

To create the Oracle database modules

1 Expand the newly created project tree, and navigate to Databases, Oracle.

2 Right-click the Oracle element, and click New to start the Create Module Wizard.

3 Enter the name of the module to create, such as STAGE_TABLES. The database parameters for each module must be those used for the Oracle Self-Service E-Billing installation. For each module, enter the OLAP database schema user name and password (OLTP and OLTP for the OLTP Module). Use the default values for other fields (default module type is Warehouse Target), and click Next.

Module Name	Module Location	Service Name	Schema
STAGE_TABLES	STAGE_TABLES_LOCATION	OLAP	OLAP
TARGET_TABLES	TARGET_TABLES_LOCATION	OLAP	OLAP
MAPPINGS	MAPPINGS_LOCATION	OLAP	MAP_USER
OLTP	OLTP_LOCATION	OLTP	OLTP

4 Click Edit next to the location field. Enter the location name and database connection details of the Oracle Self-Service E-Billing OLAP database.

5 Verify that the correct database version is selected.

6 Click Test Connection to test the database connection.

7 Click OK, then click Finish to create the new database module.

8 Repeat from Step 2 for each remaining module.

Creating a Process Flow Module

You must create a process flow module in the Oracle Warehouse Builder Repository.

This task is a step in "Process of Installing the Oracle Warehouse Builder Repository" on page 122.

To create a process flow module

1 Expand the newly created project tree, and navigate to the Process Flows, Process Flow Modules.

2 Right-click the Process Flow Modules element, and click New to start the Create Module Wizard.

3 Enter the name of the module to create, such as ETL_PF_MODULE, and click Next.

4 Click Edit next to the location field. Enter the location name as EBILLING_ETL_LOCATION. Enter the database connection details of the Oracle Self-Service E-Billing OLAP database, user name, and password for the Oracle Workflow user (OWF_MGR). Verify that the correct Oracle Workflow version is selected.

Module Name	Module Location Name	Service Name	Schema
ETL_PF_MODULE	EBILLING_ETL_LOCATION	OLAP	OWF_MGR

5 Select Test Connection to verify that you have entered the correct parameters, then click OK.

6 Click Finish to create the new process flow module. Click Cancel if the Create Process Flow
Package window appears.

Creating File System Locations

You must create file locations within the Oracle Warehouse Builder repository to define where the
data files used in the ETL process flow are located.

This task is a step in *"Process of Installing the Oracle Warehouse Builder Repository" on page 122*.

To create a file system location

1 Expand the Locations node in the Connection Explorer, and navigate to Files. (On Oracle
Database 11*g* Release 2 the Locations Node is on the Locations panel.)

2 Right-click the Files node, and click New to open the Create File System Location dialog.

3 Enter a name and file system location, then click OK. Define the following locations. The specific
file system location depends on your individual installation.

Name	Description	Example
DATA_FILES_LOCATION	Where incoming data is located	/export/home/oracle/testdata/ OWF_OUTDATA
BAD_FILE_LOCATION	Where bad data files are written	/export/home/oracle/testdata/ bad
DISCARD_FILE_LOCATION	Where discarded data is written	/export/home/oracle/testdata/ discard
LOG_FILE_LOCATION	Where log files are written	/export/home/oracle/testdata/ log

4 On the toolbar, click Save All to save all the changes in the Design Center.

Registering Locations

You must register all the database and file locations in the Oracle Warehouse Builder Design Center
using Control Center.

This task is a step in *"Process of Installing the Oracle Warehouse Builder Repository" on page 122*.

To register a location

1 In the Design Center, select the Tools menu, and click Control Center Manager.

If you receive error RTC-5301, The Runtime Platform Service cannot be started (Not Available),
verify that you have added the parameter job_queue_processes to the OLAP database
configuration.

If the Control Center Service is still not available, you can try starting it in one of the following
ways:

■ Connect to SQL*Plus as the runtime environment owner. Go to $OWB_HOME\rtp\sql , and run
the start_service.sql script. If it runs successfully, you receive the message Available. If it
says Not Available, shut down and restart both OLAP and OLTP instances and try the script
again.

■ If Oracle Warehouse Builder is installed locally, use the Start Control Center Service script on
the local computer; provide connection parameters to connect to the host on which the
Control Center has to be started.

2 Right click each of the locations listed in the left hand pane, and click Register.

3 Verify that the name and location path are correct, then click OK. Repeat this step for each
database, and file location you created previously.

Granting Rights to MAP_USER

You must grant the MAP_USER database user permissions to use the Oracle Warehouse Builder
resources.

This task is a step in "Process of Installing the Oracle Warehouse Builder Repository" on page 122.

You can grant the rights to MAP_USER using either individual grant commands or the grant repository
script.

To grant the necessary rights to MAP_USER using individual grant commands

1 Using SQL*Plus, log on to OLAP database as SYSDBA, where *OLAP_Instance* is the name of the
OLAP instance:

    export ORACLE_SID=*OLAP_Instance*

    SQLPLUS "/ as sysdba"

2 Enter the following commands:

    grant SELECT ANY DICTIONARY to map_user;

    grant SELECT ANY TABLE to map_user;

    grant INSERT ANY TABLE to map_user;

    grant UPDATE ANY TABLE to map_user;

    grant DELETE ANY TABLE to map_user;

```
grant EXECUTE ANY PROCEDURE to map_user;

grant SELECT ANY SEQUENCE to map_user;
```

If the repository is created using Oracle Warehouse Builder 10g Release 2, enter:

```
grant OWB_O_REP_ADMIN to owf_mgr;
```

If the repository is created using Oracle Warehouse Builder 11g, enter:

```
grant OWB_USER to owf_mgr;
```

To grant the necessary rights to MAP_USER using the grant repository script

■ Go to the Oracle Self-Service E-Billing database 6.0.3 installation folder *EDX_HOME*/db/ebilling/
oracle/olap/etl/packages (or the *EDX_HOME*\db\ebilling\oracle\olap\etl\packages folder
on Windows), and make sure that the repository_grants.sql file exists in the folder. In the path,
EDX_HOME is the location where Oracle Self-Service E-Billing is installed. Then log in to the OLAP
database using as sysdba using the following command at the SQL*Plus prompt:

```
export ORACLE_SID=OLAP_Instance

SQLPLUS "/ as sysdba"

SQL>@repository_grants.sql
```

Installing the ETL Module

This topic describes how to install the ETL module. The ETL Module is included in the Oracle Self-
Service E-Billing software installation directory.

This topic assumes you have installed the Oracle Warehouse Builder Repository. For details on setting
up the Oracle Warehouse Builder Repository, see "Process of Installing the Oracle Warehouse Builder
Repository" on page 122.

For Windows, change the slashes (/ or \) and root as necessary.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on
page 119.

To install the ETL Module

1 Open the ebilling_etl.properties ETL module configuration file in a text editor. This file is located in the *EDX_HOME/*db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory in Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

NOTE: If you are installing on Oracle Database 11*g* Release 2, OWB_ORACLE_HOME is the same as ORACLE_HOME.

Modify the following properties to reflect your Oracle Self-Service E-Billing environment. On Windows you must use two slashes (//) when entering file system paths. Note that the OUTPUT_DIR directory in the properties file must be the same as the directory of the DATA_FILES_LOCATION in the Oracle Warehouse Builder repository. The directory of the OUTPUT_DIR directory must *not* be the same as the directory of the INPUT_DIR.

Property	Value
OWB_ORACLE_HOME	export/home/oracle/11.2.0.1
TCL_SCRIPTS_DIRECTORY	/opt/Oracle/eBilling/db/ebilling/oracle/olap/etl
MDL_LOCATION	/opt/Oracle/eBilling/db/ebilling/oracle/olap/etl
MDL_NAME	ebilling_etl_v6.0.4.mdl
PROJECT_NAME	EBILLING_ETL
TARGET_MODULE	MAPPINGS
REP_USER	The repository admin user name.
TGT_USER	The repository admin user name.
TGT_PWD	The repository admin password.
OWB_HOST	The host name of the OLAP database. Specify the HOST name of the OLAP database where the Oracle Warehouse Builder repository has been created.
OWB_PORT	The port number of the database, such as 1521.
OWB_SERVICE	The ORACLE_SID of the OLAP database.
SYS_PASSWD	The SYS password of the OLAP database.
OLAP_USER	The OLAP user name.
OLAP_PASSWD	The OLAP password.
OLAP_SID	The ORACLE_SID of the OLAP database.
INPUT_DIR	/export/home/oracle/testdata/OWF_INDATA Specify the input directory path.
OUTPUT_DIR	/export/home/oracle/testdata/OWF_OUTDATA Specify the output directory path.

Property	Value
REJECT_DIR	/export/home/oracle/testdata/OWF_REJDATA Specify the reject directory path.
SHELL_VAR	/usr/bin/sh

Specify the following mail server parameters only if you have Oracle Database 11*g*. You can ignore these parameters if you have Oracle Database 10*g*.

Parameter	Value
MAIL_SERVER	Your company mail server name.
LOWER_PORT	The lower port of the mail server.
UPPER_PORT	The upper port of the mail server.

2 Open a command prompt, and go to the directory containing the ebilling_etl.properties file.

3 Run the following commands to set up your Oracle Warehouse Builder and Apache Ant environments, substituting the paths of your Oracle Warehouse Builder and Ant installations and correct slashes (/ or \) for your platform:

> set OWB_HOME=/export/home/oracle/product/owb_home (This command is not required if installing on Oracle Database 11*g*.)

> set ANT_HOME=/opt/apache-ant-1.6.5

> set PATH=%PATH%;%ANT_HOME%/bin

4 From the *EDX_HOME*/db/ebilling/oracle directory (the *EDX_HOME*\db\ebilling\oracle directory in Windows), run the following command to run the build.xml file:

> ant

5 Select Option 5, Install the ETL database.

6 Select Option 1, and press Enter to create the ETL package. When finished, check the etl_packages.log file found in the *EDX_HOME*/db/ebilling/oracle/olap/etl directory, for errors.

You can ignore the following error message shown on screen and in the log file:

> Errors for PACKAGE BODY "EDX_RPT_ETL":

> LINE/COL ERROR

> -------- --

> 863/7   PL/SQL: Statement ignored

> 864/10 PLS-00201: identifier 'EDX_RPT_ETL_IO_UTIL.MOVE_FILE' must be declared

7   After the create ETL package task completes, run Ant again, and choose Option 5, Install the ETL
database, then choose Option 2, Import MDL. When finished, check the *MDL File Name*.log file,
found in the *EDX_HOME*/db/ebilling/oracle/olap/etl directory, for errors. Log in to the Design
Center, and make sure all modules are connected to the database locations. For
ETL_PF_MODULE, check the LOCATION and Evaluation Location.

8   (Oracle Database 11g Release 2 only) Adjust JVM settings in OMBPlus.sh, found in the
ORACLE_HOME/owb/bin/unix directory. In the following line, change MaxPermSize to 512M:

    *) OPTS="-XX:MaxPermSize=256M -XX:+DisableExplicitGC";;

In this line, change Xms to 1024M Xmx to 2046M, and -Dlimit to 2046M:

    $JAVAPATH/bin/java -Xms64M -Xmx768M $OPTS -Dlimit=768M

9   After the import MDL task completes, run Ant again and choose Option 5, Install the ETL
database, then choose Option 3, OWB Mappings Deployment. When finished, check the
deploy.log file, found in the *EDX_HOME*/db/ebilling/oracle/olap/etl directory, for errors.

Granting File Location Privileges to MAP_USER for Oracle Database 11*g* Release 2

If you are using Oracle Database 11*g* Release 2, after mapping deployment you must grant the
MAP_USER database user permissions to access the file location.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on
page 119.

To grant file location privileges to MAP_USER

1   Using SQL*Plus, log on to the OLAP database as SYSDBA, where *OLAP_Instance* is the name of
the OLAP instance:

    export ORACLE_SID=*OLAP_Instance*

    SQLPLUS "/ as sysdba"

2   Run the following SQL statements:

    grant read,write on directory DATA_FILES_LOCATION to map_user;

    grant read,write on directory BAD_FILE_LOCATION to map_user;

    grant read,write on directory DISCARD_FILE_LOCATION to map_user;

    grant read,write on directory LOG_FILE_LOCATION to map_user;

Running the ETL Loader Job Using Sample Data

The ETL module is now ready to load data. Sample data is provided with Oracle Self-Service E-Billing for testing purposes.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119.

To run the ETL Loader job using sample data

1 Copy the following sample data files from the
EDX_HOME\db\ebilling\oracle\olap\etl\sample_data directory (or the *EDX_HOME*/db/
ebilling/oracle/olap/etl/sample_data directory on Windows) to the ETL input directory. In
the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed:

- ■ MASTER-DATA-FILE-20070509.dat

- ■ EBILLING_B2B-DATA-FILE-20070510.dat

- ■ EBILLING_B2B-DATA-FILE-20070512.dat

- ■ EBILLING_B2C-DATA-FILE-20070516.dat

- ■ EBILLING_B2B-DATA-FILE-20070551.dat

- ■ EBILLING_B2B-DATA-FILE-20070552.dat

- ■ EBILLING_B2B-DATA-FILE-20070553.dat

For consistency, it is recommended that you create the input directory in the same directory as the file locations previously defined in Oracle Warehouse Builder Design Center.

2 Log in to the Oracle Warehouse Builder Design Center as the repository owner. Start the Control Center from the Tools menu.

3 Select Tools, then Preferences. (In Oracle Database 11*g* Release 2, select OWB, then Deployment Menu.) Make sure the Prompt for execution parameters option is enabled.

4 Expand the EBILLING_ETL_LOCATION, ETL_PF_MODULE, and ETL_SUP nodes.

5 Right click the ETL_SUPER_PF_2 node (the ETL Loader job), and click Start. A dialog box might appear stating that the object must be deployed before execution. If this dialog appears, click OK. Or you can run the ant command again from the *EDX_HOME*/db/ebilling/oracle directory (the *EDX_HOME*\db\ebilling\oracle directory in Windows,) and choose Option 5, Install the ETL database, then choose Option 3, OWB Mappings Deployment. When finished, check the deploy.log file, found in the *EDX_HOME*/db/ebilling/oracle/olap/etl directory, for errors.

NOTE: The following files support single statement and multiple period functionality: EBILLING_B2B-DATA-FILE-20070552.dat; EBILLING_B2B-DATA-FILE-20070553.dat. Before loading these files, run the Hierarchy Copy job in the Command Center to the period specified in the data file.

6 You must remove sample data before going live. For information about the process of purging
sample data, see *Administration Guide for Oracle Self-Service E-Billing*.

CAUTION: You must remove sample data from your production environment to remain in
compliance with the Payment Card Industry Data Security Standard.

7 You must run the master key update script, which updates the master key as well as related
subkeys and validation code in the Oracle Self-Service E-Billing database. The master key is used
when setting up the OLTP and OLAP databases. For instructions on running the master key
update, see *Administration Guide for Oracle Self-Service E-Billing*.

CAUTION: You must run the master key update after loading sample data to remain in
compliance with the Payment Card Industry Data Security Standard.

7 Migrating to Oracle Self-Service E-Billing Version 6.0.4

This chapter describes how to migrate Oracle Self-Service E-Billing to Version 6.0.4. It includes the following topics:

- Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.0.4 on page 137
- Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1 on page 138
- Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2 on page 146
- Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3 on page 156
- Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4 on page 163

Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.0.4

This section describes the tasks necessary to migrate from Oracle Self-Service E-Billing 6.0.x to Oracle Self-Service E-Billing 6.0.4. You must migrate to each version of 6.0.x incrementally to reach version 6.0.4. For example, if you currently have version 6.0.2 installed, you must migrate first to version 6.0.3, then migrate from version 6.0.3 to 6.0.4.

To migrate to Oracle Self-Service E-Billing version 6.0.4, perform the following tasks and processes:

1 Make a full backup of your current Oracle Self-Service E-Billing database.

For additional information, see Oracle Database 10g or Oracle Database 11g documentation on Oracle Technology Network.

2 Start the database instance that accesses the database you are upgrading. Check the status of all user objects. If any of them indicate an INVALID status, contact the database administrator to correct this problem.

For additional information, see Oracle Database 10g or Oracle Database 11g documentation on Oracle Technology Network.

3 Have any existing Oracle Self-Service E-Billing database passwords available. Check with your database administrator.

4 "Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1" on page 138 (Optional)

5 "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146 (Optional)

6 "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156 (Optional)

7 "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163

Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1

To migrate from Oracle Self-Service E-Billing 6.0 to 6.0.1, perform the following tasks to migrate both your OLAP and OLTP databases:

1 Back up your existing OLTP and OLAP Oracle Self-Service E-Billing databases.

For additional information, see Oracle Database 10*g* or Oracle Database 11*g* documentation on Oracle Technology Network.

2 Install Oracle Self-Service E-Billing version 6.0.1. For information about installing, see "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 19.

3 "Migrating to Oracle Database 11g" on page 138

4 "Migrating Oracle Self-Service E-Billing 6.0 OLTP to 6.0.1 OLTP on UNIX" on page 141 or "Migrating Oracle Self-Service E-Billing 6.0 OLTP to 6.0.1 OLTP on Windows" on page 142

5 "Migrating Oracle Self-Service E-Billing 6.0 OLAP to 6.0.1 OLAP (UNIX and Windows)" on page 143

6 "Compiling the Schema for the Oracle Self-Service E-Billing 6.0.1 OLTP and OLAP Databases" on page 145

7 Uninstall Oracle Self-Service E-Billing version 6.0. For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 23.

This process is a step in the following roadmaps:

■ "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.0.4" on page 137

■ "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 186

Migrating to Oracle Database 11*g*

If you are migrating Oracle Self-Service E-Billing from Oracle Database 10*g* to Oracle Database 11*g*, you must follow these procedures. These steps are for both UNIX and Windows environments.

CAUTION: This procedure is not required if your Oracle Self-Service E-Billing installation is on Oracle Database 10*g*.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1" on page 138.

To migrate Oracle Self-Service E-Billing from Oracle Database 10g to Oracle Database 11g

1 On UNIX, log in as the ORACLE user for migration activity. Export your existing OLAP and OLTP 6.0 databases from Oracle Database 10*g* using the following commands:

```
exp system/manager@olap_tnsname file=olap_export.dmp FULL=Y log=olap_export.log
```

```
exp system/manager@oltp_tnsname file=oltp_export.dmp FULL=Y log=oltp_export.log
```

2 Install Oracle Database 11*g*.

3 Create a new Oracle Database 11*g* database instance, tablespace, and schema for OLAP.

4 Verify that the following values in the edxadmin_unix.`properties file, found in the`
EDX_HOME/db/ebilling/oracle `directory` (or the *EDX_HOME*\db\ebilling\oracle directory on
Windows) are correct for the current installation. In the path, *EDX_HOME* is the location where
Oracle Self-Service E-Billing is installed:

- ■ ORACLE_BASE and ORACLE_HOME locations

- ■ OLTP and OLAP database SID, user name, and password

- ■ SYSDBA password and tnsnames for OLTP and OLAP

- ■ Database file locations

- ■ Redo file locations

- ■ Trace file location

5 Create the Oracle Self-Service E-Billing database. Change directory to the location of the Oracle
Database installation files in your software installation: *EDX_HOME*/db/ebilling/oracle (or
EDX_HOME\db\ebilling\oracle for Windows). If you have not configured the Apache Ant
environment, do so now:

- ■ UNIX:

  ```
  export ANT_HOME=/opt/apache-ant-1.6.5
  ```

  ```
  export PATH=$ANT_HOME/bin:$PATH
  ```

  If you are on Oracle WebLogic, run this command, where *JDK150_11* is your JDK version:

  ```
  export JAVA_HOME= $WEBLOGIC_HOME/JDK150_11
  ```

  If you are on IBM WebSphere, run this command:

  ```
  export $JAVA_HOME=$WS_HOME/java for IBM WebSphere)
  ```

  Also, for all application servers, run this command:

  ```
  export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$PATH
  ```

- ■ Windows:

  ```
  set ANT_HOME=C:\apache-ant-1.6.5
  ```

  ```
  set PATH=%PATH%;%ANT_HOME%\bin
  ```

  ```
  set JAVA_HOME= %WEBLOGIC_HOME%\JDK150_11
  ```

  where *JDK150_11* is your JDK version.

a Run the build script by typing ant.

  By default, Ant picks up the build.xml file in the current directory.

b Select Option 1, Install the OLAP database. Complete Options 1-5 on the OLAP menu to create the OLAP database instance, tablespaces, and schema. Review all log files for possible errors even if a Build Successful message appears. When done, choose Q (Quit).

6 Create a new Oracle Database 11*g* database instance, tablespace, and user for OLTP (eStatement):

a Verify that the following values are correct in the edxadmin.properties file, found in the *EDX_HOME*/db/eStatement/oracle directory (or the *EDX_HOME*\db\eStatement\oracle directory for Windows):

❏ ORACLE_BASE and ORACLE_HOME

❏ OLTP database SID, user name, and password

❏ SYSTEM password for OLTP

❏ Database file locations

❏ Redo file locations

❏ Trace file location

b Go to the directory location of the eStatement Oracle Database installation files in your software installation:

❏ **UNIX.** *EDX_HOME*/db/eStatement/oracle

❏ **Windows.** *EDX_HOME*\db\eStatement\oracle

c Run the following commands to configure the Apache Ant environment:

❏ UNIX:

export ANT_HOME=/opt/apache-ant-1.6.5

If you are on Oracle WebLogic, run this command, where *JDK150_11* is your JDK version:

export JAVA_HOME=$WEBLOGIC_HOME/*JDK150_11*

If you are on IBM WebSphere, run this command:

export $JAVA_HOME=$WS_HOME/java

Also, for all application servers, run this command:

export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$PATH

❏ Windows:

set ANT_HOME=C:\apache-ant-1.6.5

set JAVA_HOME=%WEBLOGIC_HOME%*JDK150_11*

set PATH=%PATH%;%JAVA_HOME%\bin;%ANT_HOME%\bin

where *JDK150_11* is your JDK version.

d Run the build script by typing ant.

By default, Ant picks up the build.xml file in the current directory.

    e From the top level Main Menu, select Option 1, Install eStatement Database to start.

        The Install eStatement Database menu appears.

    f Complete Options 1-4 on the Install eStatement Database Menu to create the OLTP instance, tablespace, and user. Review all log files for possible errors even if a Build Successful message appears. When done, choose Q (Quit).

7 Create the database link, TAM_LINK, in the OLTP schema.

8 Log on to the OLTP instance as SYSDBA, and execute the following command, where *OLTP_Schema* is the name of the OLTP schema:

    SQL> GRANT CREATE DATABASE LINK TO *OLTP_Schema*;

9 Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows).

10 Log on to the OLTP schema using SQL*Plus (not as SYSDBA).

11 Execute the following SQL script, providing the three input parameters:

    SQL> DROP DATABASE LINK TAM_LINK;

    SQL>@ crt_db_link.sql *OLAP_User OLAP_Password OLAP_TNS_Name*

    SQL>exit

    where:

    ■ *OLAP_User* is the name of the OLAP schema user.

    ■ *OLAP_Password* is the OLAP schema user's password.

    ■ *OLAP_TNS_Name* is the name of the OLAP instance.

12 Import Oracle Database 10*g* database into the new database:

    imp system/manager@oltp_sid fromuser=oltp touser=oltp file=oltp_export.dmp log=oltp_import.log

    imp system/manager@olap_sid fromuser=olap touser=olap file=olap_export.dmp log=oltp_import.log

    NOTE: If the importing and exporting schema names are different, you might receive a message indicating that the user does not exist for snapshot logs after the importing some objects. You can ignore this message.

Migrating Oracle Self-Service E-Billing 6.0 OLTP to 6.0.1 OLTP on UNIX

If you are migrating from Oracle Self-Service E-Billing 6.0 to 6.0.1 on UNIX, follow these steps to migrate your OLTP database.

This task is a step in .

To migrate your Oracle Self-Service E-Billing 6.0 OLTP database to 6.0.1

1 Log in as the ORACLE user for migration activity on UNIX. Go to the *EDX_HOME*/eBilling/db/ ebilling/oracle/oltp/migration/Ebilling_6.0_to_6.0.1 directory, where *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed:

2 Verify that the following files exist:

■ migrate_oltp_6.0_to_6.0.1.sh

■ migrate_oltp_6.0_to_6.0.1.sql

3 Run the following commands at the shell prompt:

Bash$ chmod 777 migrate_oltp_6.0_to_6.0.1.sh

Bash$. <space>migrate_oltp_6.0_to_6.0.1.sh

4 Provide the correct values for your environment.

Field	What to Enter
Database ID	Instance name
Database Username	Schema name
Database Password	Schema password
SYS Password	Password of SYS

5 Check the db_oltp_migrate_6.0_6.0.1.log and migrate_oltp_6.0_to_6.0.1.log files for errors.

Migrating Oracle Self-Service E-Billing 6.0 OLTP to 6.0.1 OLTP on Windows

If you are migrating from Oracle Self-Service E-Billing 6.0 to 6.0.1 on Windows, follow these steps to migrate your OLTP database.

This task is a step in .

To migrate your Oracle Self-Service E-Billing 6.0 OLTP database to 6.0.1

1 Go to the *EDX_HOME*\eBilling\db\ebilling\oracle\oltp\migration\Ebilling_6.0_to_6.0.1 directory, where *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Verify that the migrate_oltp_6.0_to_6.0.1.sql file exists.

3 Execute the userlock.sql script from the SQL*Plus prompt, where *sys_password* is the password of the sys user:

C:\set ORACLE_SID=oltp

C:\sqlplus sys/*sys_password* as sysdba

4 Run the userlock.sql as shown, where *ORACLE_HOME* is the directory where the Oracle database
software is installed:

> SQL>@ *ORACLE_HOME*\rdbms\admin\userlock.sql

> SQL> exit;

5 Run the migration script as follows:

> C:\set ORACLE_SID=oltp

> C:\sqlplus oltp/oltp

> SQL> @ migrate_oltp_6.0_to_6.0.1.sql

> SQL> exit;

6 Check the db_oltp_migrate_6.0_6.0.1.log and migrate_oltp_6.0_to_6.0.1.log files for
errors.

Migrating Oracle Self-Service E-Billing 6.0 OLAP to 6.0.1 OLAP (UNIX and Windows)

If you are migrating from Oracle Self-Service E-Billing 6.0 to 6.0.1 one UNIX or Windows, follow
these steps to migrate your OLAP database.

This task is a step in .

To migrate your Oracle Self-Service E-Billing 6.0 OLAP database to 6.0.1

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/
oracle/olap/migration/ebilling_6.0_to_6.0.1 directory (or the
EDX_HOME\db\ebilling\oracle\olap\migration\ebilling_6.0_to_6.0.1 directory on
Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed:

2 Verify that the following files exist:

■ migrate_olap_6.0_to_6.0.1.sh

■ migrate_olap_6.0_to_6.0.1.sql

■ Create_Tables.sql

■ Load_Data.sql

■ Create_Procedure.sql

■ Create_Indexes.sql

■ Create_Constraints.sql

3 Create the database link (OLTP_LINK) in the OLAP schema. Log on to the OLAP instance as
SYSDBA, and run the following command, where *OLAP_Schema* is the name of the OLAP schema:

> GRANT CREATE DATABASE LINK TO *OLAP_Schema*;

4   Go to the *EDX_HOME/*eBilling/db/ebilling/oracle directory (or the
EDX_HOME\eBilling\db\ebilling\oracle directory on Windows).

5   Log on to the OLAP schema using SQL*Plus (not as SYSDBA).

6   Run the following script at SQL prompt, providing the three input parameters:

SQL> DROP DATABASE LINK OLTP_LINK;

SQL>CREATE DATABASE LINK OLTP_LINK CONNECT TO *OLTP_User* IDENTIFIED BY
OLTP_Password USING '*OLTP_TNS_Name*';

SQL>exit

where:

■   *OLTP_User* is the name of the OLTP schema user.

■   *OLTP_Password* is the OLTP schema user's password.

■   *OLTP_TNS_Name* is the name of the OLTP instance.

7   (UNIX only) Perform the following steps:

a   Run the following commands at the shell prompt:

Bash$ chmod 777 migrate_olap_6.0_to_6.0.1.sh

Bash$. <space>migrate_olap_6.0_to_6.0.1.sh

b   Substitute the correct values for your environment where appropriate.

Field	What to Enter
OLAP Database SID	OLAP instance name
OLAP Database Username	OLAP schema name
OLAP Database Password	OLAP schema password
OLAP SYS Password	Password of SYS

c   If you are migrating on to an Oracle Database 11g database, enter the following additional
parameters.

Field	What to Enter
Mail server name	Mail server name
Lower port	Lower port of the mail server
Upper Port	Upper port of the mail server

d   Check db_oltp_migrate_6.0_6.0.1.log, and migrate_oltp_6.0_to_6.0.1.log files for errors.

8   (Windows only) Go to the
EDX_HOME\db\ebilling\oracle\olap\migration\Ebilling_6.0_to_6.0.1 directory, and
perform the following steps:

a Verify that the `migrate_olap_6.0_to_6.0.1.sql` file exists.

b Copy EDX_RPT_ETL.sql from the
    EDX_HOME`\db\oracle\olap\migration\Ebilling_6.0_to_6.0.1` directory to the
    EDX_HOME`\db\oracle\olap\etl\packages` directory.

c Run the migration script as follows:

```
C:\set ORACLE_SID=olap

C:\sqlplus olap/olap

SQL> @ migrate_olap_6.0_to_6.0.1.sql

SQL> exit;
```

d Check the migrate_olap_6.0_to_6.0.1.log file for errors.

e If you are migrating to Oracle Database 11*g*, run the acl.sql script, specifying the input values
    appropriate for your implementation:

```
CD EDX_HOME\db\ebilling\oracle\olap\etl\packages

c:\set ORACLE_SID=olap

c:\Sqlplus olap/olap

SQL>acl.sql Mail_Servername Lower_port_of_mail_server
Upper_port_of_mail_server

SQL>exit
```

where:

❏ *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing.

❏ *Mail_servername* is the name of your mail server.

❏ *Lower_port_of_mail_server* is the number of the lower port on your mail server.

❏ *Upper_port_of_mail_*server is the number of the upper port on your mail server.

f Check the acl.log file for errors. Ignore errors ORA-31003 and ORA-06512.

Compiling the Schema for the Oracle Self-Service E-Billing 6.0.1 OLTP and OLAP Databases

After successfully migrating the OLTP and OLAP databases, you must compile the schema.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1" on page 138.

To compile the schema

1 On UNIX, log in as the ORACLE user for migration activity. Go to the `/opt/Oracle/eBilling/db/`
    `ebilling/oracle` directory (or the `\Oracle\eBilling\db\ebilling\oracle` directory on
    Windows).

2 Log on to the OLTP schema using SQL*Plus (not as SYSDBA)

3 Run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

4 Log on to the OLAP schema using SQL*Plus (not as SYSDBA).

5 Run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2

To migrate from Oracle Self-Service E-Billing version 6.0.1 to 6.0.2, perform the following tasks:

1 Back up your existing OLTP and OLAP Oracle Self-Service E-Billing databases.

For additional information, see Oracle Database 10*g* or Oracle Database 11*g* documentation on Oracle Technology Network.

2 Install Oracle Self-Service E-Billing version 6.0.2. For information about installing, see "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 19.

3 "Migrating to Oracle Database 11g" on page 138 (Optional)

4 "Migrating Oracle Self-Service E-Billing Version 6.0.1 OLTP to 6.0.2 on UNIX" on page 148 or Migrating Oracle Self-Service E-Billing Version 6.0.1 OLTP to 6.0.2 on Microsoft Windows on page 150

5 "Migrating Oracle Self-Service E-Billing Version 6.0.1 OLAP to 6.0.2 on UNIX" on page 149 or Migrating Oracle Self-Service E-Billing Version 6.0.1 OLAP to 6.0.2 on Microsoft Windows on page 151

6 "Compiling the Schema for the Oracle Self-Service E-Billing 6.0.2 OLTP and OLAP Databases" on page 152

7 "Migrating the Payment Gateway from Verisign to PayPal Payflow Pro" on page 152

8 "Exporting and Linking the PayPal Payflow Pro Certificate (AIX Only)" on page 153

9 "Setting Up Last Participation Support for the Command Center (AIX Only)" on page 154

10 "Setting Up the Web Container Filter for IBM WebSphere 6.1.0.3 and Higher Versions" on page 155

11 Uninstall Oracle Self-Service E-Billing version 6.0.1. For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 23.

This process is a step in the following roadmaps:

■ "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.0.4" on page 137

■ "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 186

Migrating to Oracle Database 11*g*

If you are migrating Oracle Self-Service E-Billing from Oracle Database 10*g* to Oracle Database 11*g*, you must follow these procedures. These steps are for both UNIX and Windows environments.

CAUTION: This procedure is not required if your Oracle Self-Service E-Billing installation is on Oracle Database 10*g*.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146.

To migrate Oracle Self-Service E-Billing from Oracle Database 10g to Oracle Database 11g

1 On UNIX, log in as the ORACLE user for migration activity. Export your OLAP and OLTP Oracle Self-Service E-Billing 6.0.1 databases.

2 Create a new database instance, tablespaces, and user for the new OLAP and OLTP databases. For complete instructions on running the Ant script, see "Creating the Oracle Self-Service E-Billing Database Using Ant" on page 33. Do not use the automated Ant target.

3 Create the database link (TAM_LINK) in the OLTP schema:

    a Log on to the OLTP instance as SYSDBA, and run the following command, where *OLTP_Schema* is the name of the OLTP schema:

        `GRANT CREATE DATABASE LINK TO `*OLTP_Schema*`;`

    b Go to the *EDX_HOME*/db/ebilling/oracle directory. Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following command:

        `DROP DATABASE LINK TAM_LINK;`

    c Run the following script at SQL prompt:

        `SQL>@ crt_db_link.sql `*OLAP_User OLAP_Password OLAP_TNS_Name*

        `SQL>exit`

    where:

      ❏ *OLAP_User* is the name of the OLAP schema user.

      ❏ *OLAP_Password* is the OLAP schema user's password.

      ❏ *OLAP_TNS_Name* is the name of the OLAP instance.

4 Create the database link (OLTP_LINK) in the OLAP schema:

    a Log on to the OLAP instance as SYSDBA, and run the following command, where *OLAP_Schema* is the name of the OLAP schema:

        `GRANT CREATE DATABASE LINK TO `*OLAP_Schema*`;`

    b Log on to the OLAP schema using SQL*Plus (not as SYSDBA), and run the following commands:

         SQL> DROP DATABASE LINK OLTP_LINK;

         SQL>CREATE DATABASE LINK OLTP_LINK CONNECT TO OLTP_USER IDENTIFIED BY *OLTP_ Password* USING '*OLTP_TNS_Name*';

         SQL>exit

    where:

- *OLTP_User* is the name of the OLTP schema user.
- *OLTP_Password* is the OLTP schema user's password.
- *OLTP_TNS_Name* is the name of the OLTP instance.

5   Import the OLAP and OLTP schemas for Oracle Self-Service E-Billing 6.0.1 into the new database.

Migrating Oracle Self-Service E-Billing Version 6.0.1 OLTP to 6.0.2 on UNIX

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.1 OLTP to 6.0.2 on UNIX.

This task is a step in .

To migrate Oracle Self-Service E-Billing 6.0.1 OLTP to 6.0.2 on UNIX

1   On UNIX, log in as the ORACLE user for migration activity. Run the following commands from SQL*Plus:

    bash$ export ORACLE_SID=oltp

    bash$ sqlplus sys/*sys_password* as sysdba

    SQL> @ *ORACLE_HOME*\rdbms\admin\userlock.sql

    SQL> exit

    where:

- *sys_password* is the password of the sys user.
- *Oracle_Home* is the exact path where Oracle database is installed.

2   Go to the *EDX_HOME*/db/ebilling/oracle/oltp/migration/ebilling_6.0.1_to_6.0.2 directory.

3   Verify that the following files exist:

- migrate_oltp_6.0.1_to_6.0.2.sh
- migrate_oltp_6.0.1_to_6.0.2.sql
- Create_Procedure.sql

4   Run the following commands at the shell prompt:

Bash$ chmod 777 migrate_oltp_6.0.1_to_6.0.2.sh

Bash$. migrate_oltp_6.0.1_to_6.0.2.sh

5 Provide the correct values for your environment.

the

Field	What to Enter
OLTP Database SID	OLTP instance name
OLTP Database Username	OLTP schema name
OLTP Database Password	OLTP schema password
OLTP SYS Password	Password of SYS

6 Check the db_oltp_migrate_6.0.1_6.0.2.log, and migrate_oltp_6.0.1_to_6.0.2.log files for errors.

Migrating Oracle Self-Service E-Billing Version 6.0.1 OLAP to 6.0.2 on UNIX

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.1 OLAP to 6.0.2 on UNIX.

This task is a step in

To migrate Oracle Self-Service E-Billing 6.0.1 OLAP to 6.0.2 on UNIX

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/oracle/olap/migration/ebilling_6.0.1_to_6.0.2 directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed:

2 Verify that the following files exist:

■ migrate_olap_6.0.1_to_6.0.2.sh

■ migrate_olap_6.0.1_to_6.0.2.sql

■ Create_Procedure.sql

■ Version.sql

3 Run the following commands at the shell prompt:

Bash$ chmod 777 migrate_olap_6.0.1_to_6.0.2.sh

Bash$. migrate_olap_6.0.1_to_6.0.2.sh

4 Provide the correct values for your environment.

Field	What to Enter
OLAP Database SID	OLAP instance name
OLAP Database Username	OLAP schema name
OLAP Database Password	OLAP schema password
OLAP SYS Password	Password of SYS
Mail server name. Appears only if migrating to Oracle Database 11*g*.	Mail server name
Lower Port. Appears only if migrating to Oracle Database 11*g*.	Lower port number of the mail server
Upper Port. Appears only if migrating to Oracle Database 11*g*.	Upper port number of the mail server

5 Check the db_olap_migrate_6.0.1_6.0.2.log, and migrate_olap_6.0.1_to_6.0.2.log files for errors.

Migrating Oracle Self-Service E-Billing Version 6.0.1 OLTP to 6.0.2 on Microsoft Windows

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.1 OLTP to 6.0.2 on Microsoft Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146.

To migrate Oracle Self-Service E-Billing 6.0.1 OLTP to 6.0.2 on Microsoft Windows

1 Go to the *EDX_HOME*\db\ebilling\oracle\oltp\migration\ebilling_6.0.1_to_6.0.2 directory, where *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Verify that the following files exist:

■ migrate_oltp_6.0.1_to_6.0.2.sh

■ migrate_oltp_6.0.1_to_6.0.2.sql

■ Create_Procedure.sql

3 Run the following commands from SQL*Plus:

C:\set ORACLE_SID=oltp

C:\sqlplus sys/*sys_password* as sysdba

SQL> @ *Oracle_Home*\rdbms\admin\userlock.sql

SQL> exit;

where:

■ *sys_password* is the password of the sys user.

■ *Oracle_Home* is the exact path where Oracle database is installed.

4 Run the following commands from SQL*Plus:

    `C:\set ORACLE_SID=oltp`

    `C:\sqlplus oltp/oltp`

    `SQL> @ migrate_oltp_6.0.1_to_6.0.2.sql`

    `SQL> exit;`

5 Check the migrate_oltp_6.0.1_to_6.0.2.log, and db_oltp_migrate_6.0.1_to_6.0.2.log files for
errors.

Migrating Oracle Self-Service E-Billing Version 6.0.1 OLAP to 6.0.2 on Microsoft Windows

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.1 OLAP to 6.0.2 on Microsoft Windows.

This task is a step in

To migrate Oracle Self-Service E-Billing 6.0.1 OLAP to 6.0.2 on Microsoft Windows

1 Go to the *EDX_HOME*`\db\ebilling\oracle\olap\migration\ebilling_6.0.1_to_6.0.2`
`directory,` where *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Verify that the following files exist:

■ migrate_olap_6.0.1_to_6.0.2.sh

■ migrate_olap_6.0.1_to_6.0.2.sql

■ Create_Procedure.sql

■ Version.sql

3 Run the following commands from SQL*Plus:

    `C:\set ORACLE_SID=olap`

    `C:\sqlplus olap/olap`

    `SQL> @ migrate_olap_6.0.1_to_6.0.2.sql`

    `SQL> exit;`

4 Check the migrate_olap_6.0.1_to_6.0.2.log file for Oracle errors.

5 If migrating to Oracle Database 11*g*, follow these steps:

    a Run the acl.sql script, located in *EDX_HOME*`\db\ebilling\oracle\olap\etl\packages`
directory, as follows:

        `set ORACLE_SID=olap`

Sqlplus olap/olap

SQL>acl.sql *Mail_Server_Name Lower_Port_of_Mail_Server Upper_Port_of_Mail_Server*

SQL>exit

where:

❑ *Mail_Server_Name* is the name of your mail server.

❑ *Lower_Port_of_Mail_Server* is the lower port number of your mail server.

❑ *Upper_Port_of_Mail_Server* is the upper port number of your mail server.

b Check acl.log for errors. Ignore ORA-31003 and ORA-06512.

Compiling the Schema for the Oracle Self-Service E-Billing 6.0.2 OLTP and OLAP Databases

After successfully migrating the OLTP and OLAP databases, you must compile the schema. Follow these steps for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146.

To compile the schema for Oracle Self-Service E-Billing 6.0.2

1 After successfully migrating the OLAP and OLTP databases, you must compile the schema. On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

3 Log on to the OLAP schema using SQL*Plus (not as SYSDBA), and run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

Migrating the Payment Gateway from Verisign to PayPal Payflow Pro

Follow these steps to migrate your credit card payment gateway from Verisign to PayPal Payflow Pro.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146.

To migrate your credit card payment gateway from Verisign to PayPal Payflow Pro

1 In the Command Center, delete the Verisign payment gateway.

2 Create a new credit card gateway for PayPal Payflow Pro using the configuration values shown in the following table. Use default values for the rest of the fields.

Field	Values
JNDI name of IAccount	edx/ejb/AdminAccount
Implementation of IUserAccountAccessor	com.edocs.common.services.payment.plugin.DummyUserAccountAccessor
IPaymentAccountAccessor	com.edocs.payment.payenroll.payacct.SSOPaymentAccountAccessor
Paypal host name	pilot-payflowpro.paypal.com
Paypal port	443
Proxy address	(Optional field) If you are using proxy for network connection, specify www-*YourCompanyproxy.com* where *YourCompanyproxy.com* is your proxy setting.
Proxy port	(Optional field) 80. If you are using proxy for network connection, fill it with your proxy setting.
Paypal user	eaSuite47
Paypal vender	eaSuite47
Paypal partner	PayPal
Paypal password	eaSuite47
URLStreamHandler class of application server	IBM WebSphere 6.1: com.ibm.net.ssl.www2.protocol.https.Handler Oracle WebLogic: sun.net.www.protocol.https.Handler

Exporting and Linking the PayPal Payflow Pro Certificate (AIX Only)

To use PayPal Payflow Pro as a gateway, IBM WebSphere users must export the PayPal Payflow Pro digital certificate and link it in the IBM WebSphere console.

This task is a step in

To export and link the PayPal Payflow Pro certificate

1 In Firefox 3.0 or later, go to

   https://pilot-payflowpro.paypal.com/

2 Click the blue icon to the left of the URL. Click More Information.

3 In the Security window, click View Certificate.

4 Click the Details tab. Click Export to save the certificate.

5 Add the PayPal Payflow Pro certificate link in your IBM WebSphere console. Complete this step for both the Billing and Payment, and Command Center profiles:

   a Log on to the IBM WebSphere console.

   b Select Security, SSL certificate and key management, Key stores and certificates, NodeDefaultTrustStore, Signer certificates, and then Add signer certificate.

   c In the Alias field, type payflow.

   d In the File name field, specify the full file path where the certificate you downloaded is located, such as $ebilling/pilot-payflowpro.paypal.com.

6 Save and restart application server.

Setting Up Last Participation Support for the Command Center (AIX Only)

Complete the following procedure to set up the last participation support in the Command Center if it is processing two-phase commit global transactions.

Last participant support is an extension to the transaction service that makes it possible for a single one-phase resource to participate in a two-phase transaction with one or more two-phase resources.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146.

To set up last participation support for the Command Center application

1 Log on to the IBM WebSphere console.

2 Select Applications, Enterprise Applications.

3 Select the Command Center application.

4 Under Detail Properties, click Last participant support extension.

5 Select Accept Heuristic Hazard.

6 Click Apply, and Save changes.

7 Restart the IBM WebSphere server.

Setting Up the Web Container Filter for IBM WebSphere 6.1.0.3 and Higher Versions

Users of IBM WebSphere 6.1.0.3 and higher must follow these procedures to set up the Web container filter and log4j output for the Billing and Payment application and the Customer Service Representative application.

This task is a step in .

To set up the Web container filter in IBM WebSphere 6.1.0.3 and higher for the Billing and Payment application

1 Log on to the IBM WebSphere console.

2 Select Application servers, ebilling, Web container, Custom Properties.

3 Create a new property called com.ibm.ws.webcontainer.invokefilterscompatibility. Specify the Value field as True.

4 Restart the Billing and Payment application server.

Complete the following procedure to set up log4j output on IBM WebSphere 6.1.0.17 and higher for the Billing and Payment application.

To set up log4j output in IBM WebSphere 6.1.0.17 and higher for the Billing and Payment application

1 Create a new file called commons-logging.properties in the `$WAS_HOME/profiles/eBilling_profile/properties` directory.

2 Add the following content:

> `org.apache.commons.logging.LogFactory=org.apache.commons.logging.impl.LogFactoryImpl`

3 Correct the log4j file configuration. Access the IBM Websphere console for the Billing and Payment application. Select Servers, Application servers, your server, Server Infrastructure, Java and Process Management, Process Definition, Additional Properties, Java Virtual Machine, Generic JVM arguments. Add file: in the -Dlog4j.configuration definition:

> `-Dlog4j.configuration=file:${your log4j full path and name} example: -Dlog4j.configuration=file:/export/home/qa1/eBilling602/config/log4j.xml`

4 Restart the Billing and Payment application server.

To set up the Web container filter in IBM WebSphere 6.1.0.3 and higher for the Customer Service Representative application

1 Log on to the IBM WebSphere console.

2 Select Application server1 (csr), Web container, Custom Properties.

3 Create a new property called com.ibm.ws.webcontainer.invokefilterscompatibility. Specify true in the Value field.

4   Restart the server.

Complete the following procedure to set up log4j output on IBM WebSphere 6.1.0.17 and higher for the Customer Service Representative application.

To set up log4j output in IBM WebSphere 6.1.0.17 and higher for the Customer Service Representative application

1   Create a new file called commons-logging.properties in the $WAS_HOME/profiles/$profile/ properties directory.

2   Add the following content to the file:

org.apache.commons.logging.LogFactory=org.apache.commons.logging.impl.LogFactoryImpl

3   Restart the Customer Service Representative application server.

Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3

CAUTION: You must run the migration process on the Oracle Self-Service E-Billing 6.0.3 database server only.

To migrate from Oracle Self-Service E-Billing version 6.0.2 to 6.0.3, perform the following processes and tasks:

1   Back up your existing OLTP and OLAP Oracle Self-Service E-Billing databases.

For additional information, see Oracle Database 10*g* or Oracle Database 11*g* documentation on Oracle Technology Network.

2   Process all pending batch reports in Oracle Self-Service E-Billing 6.0.2 before migrating Oracle Self-Service E-Billing version 6.0.2 to 6.0.3.

For details on how to run batch report jobs, see *Administration Guide for Oracle Self-Service E-Billing.*

3   Install Oracle Self-Service E-Billing version 6.0.3. For information about installing, see "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 19, choosing Option 2, Oracle E-Billing and Migration Tools on the Product Features Screen.

4   "Process of Repackaging the GNU Lesser General Public License" on page 67.

5   "Removing Sensitive Authentication Data from Oracle Self-Service E-Billing 6.0.2" on page 157.

6   "Migrating the OLTP Database from Oracle Self-Service E-Billing 6.0.2 to a New Character Set (Optional)" on page 157

7   "Migrating Oracle Self-Service E-Billing Version 6.0.2 OLTP to 6.0.3" on page 159.

8   "Migrating Oracle Self-Service E-Billing Version 6.0.2 OLAP to 6.0.3" on page 161.

9   "Compiling the Schema for the Oracle Self-Service E-Billing 6.0.3 OLTP and OLAP Databases" on page 161.

10

11 After successfully migrating, remove the old cryptographic materials by securely removing the previous Oracle Self-Service E-Billing installation folder. Use an industry-standard security tool such as SDelete and Eraser for Windows, and Shredder for Linux to remove the files and folders.

12 Uninstall Oracle Self-Service E-Billing version 6.0.2. For information about uninstalling, see .

This process is a step in the following roadmaps:

■

■

Removing Sensitive Authentication Data from Oracle Self-Service E-Billing 6.0.2

You must securely delete sensitive historical authentication data that was stored in Oracle Self-Service E-Billing 6.0.2. The sensitive data includes magnetic stripe data, card validation codes, PINs, and PIN blocks.

This task is a step in .

To securely remove the sensitive authentication data

1 Connect to the OLTP database using SQL*Plus.

2 Run the following SQL script in SQL*Plus. This script is found in the *EDX_HOME*/db/ebilling/Oracle/oltp/migration/ebilling6.0.2_to_6.0.3 directory (use a backslash (\) as the path separation character on Windows):

    @sec_del.sql creditcard_payments_history FLEXIBLE_FIELD_1

Migrating the OLTP Database from Oracle Self-Service E-Billing 6.0.2 to a New Character Set (Optional)

The OLTP character set changed from WE8ISO8859P1 in Oracle Self-Service E-Billing 6.0.2 to AL32UTL8 in Oracle Self-Service E-Billing 6.0.3. Character set AL32UTL8 can store languages that are not available in character set WE8ISO8859P1.

If you want to support a language that is available only in the new character set, you must migrate your Oracle Self-Service E-Billing 6.0.2 OLTP database to the new character set. For more information about which languages can be stored in most common character sets, see Article ID 62421.1 on My Oracle Support.

To use the new character set, you must perform a full export and import of the OLTP database.

This task is a step in "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156.

To migrate the Oracle Self-Service E-Billing 6.0.2 OLTP database to the new character set

1 It is strongly recommended to create a full backup of your OLTP database and check your source OLTP database with Csscan to avoid data loss.

2 It is recommended to review the following notes on My Oracle Support:

■ **Article ID 745809.1**. Installing and configuring CSSCAN in Oracle Database 10*g* and Oracle Database 11*g*.

■ **Article ID 260192.1.** Changing the NLS_CHARACTERSET to AL32UTF8 / UTF8 (Unicode).

3 Verify the NLS_CHARACTERSET on the source database. Log on to the OLTP instance as SYSDBA, and run the following command:

```
SQL>select * from nls_database_parameters where parameter = 'NLS_CHARACTERSET';
```

4 Export the NLS_LANG set to the new character set:

■ UNIX:

```
export NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

exp system/manager@oltp_tnsname file=oltp_export.dmp FULL=Y
log=oltp_export.log
```

■ Windows:

```
set NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

exp system/manager@oltp_tnsname file=oltp_export.dmp FULL=Y
log=oltp_export.log
```

5 Create a new OLTP database with the new character set.

a In "Creating the Oracle Self-Service E-Billing Database Using Ant," follow Step 1 through Step 5 on page 35.

b Follow Step 12 through Step 15 on page 35 in the same procedure.

c Select Option 4, Install Application Database - Create tablespace and user.

6 Create the database link, TAM_LINK, in the OLTP schema. Log on to the OLTP instance as SYSDBA, and run the following command, where *OLTP_Schema* is the name of the OLTP schema:

```
SQL> GRANT CREATE DATABASE LINK TO OLTP_Schema;
```

7 Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows). Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following SQL script:

```
SQL> DROP DATABASE LINK TAM_LINK;

SQL>@ crt_db_link.sql OLAP_User OLAP_Password OLAP_TNS_Name
```

```
SQL>exit
```

where:

- *OLAP_User* is the name of the OLAP schema user.
- *OLAP_Password* is the OLAP schema user's password.
- *OLAP_TNS_Name* is the name of the OLAP instance.

8 Import the exported data into the new AL32UTF8 character set. Note that the NLS_LANG setting is uses the source character set:

- UNIX:

  ```
  export NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

  imp system/manager@oltp_sid fromuser=oltp touser=oltp file=oltp_export.dmp
  log=oltp_import.log
  ```

- Windows

  ```
  set NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

  imp system/manager@oltp_sid fromuser=oltp touser=oltp file=oltp_export.dmp
  log=oltp_import.log
  ```

Migrating Oracle Self-Service E-Billing Version 6.0.2 OLTP to 6.0.3

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.2 OLTP to 6.0.3 for both UNIX and Windows.

This task is a step in *“Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3” on page 156*.

To migrate Oracle Self-Service E-Billing 6.0.2 OLTP to 6.0.3

1 The following directories are required for migrating OLTP database. Make sure they are accessible on the OLTP database server:

- $EDX_HOME/xma (%EDX_HOME%\xma on Windows)
- $EDX_HOME/keystore(%EDX_HOME%\keystore on Windows)

2 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/ oracle/oltp/migration/ebilling 6.0.2_to_6.0.3 directory (or the *EDX_HOME*\db\ebilling\oracle\oltp\migration\ebilling6.0.2_to_6.0.3 directory on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Open the migrate_oltp_6.0.2_to_6.0.3.properties file, and set the correct value for each property in the file.

Property	Value
ORACLE_HOME	ORACLE HOME directory
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password
OLTP_SID	OLTP instance name
OLTP_HOST	OLTP database host name or IP address
LISTEN_PORT	OLTP database listen port
EBILLING_EAR_DIR	Directory containing the ebilling-weblogic-10-6.0.3.ear file after packaging LGPL libraries

NOTE: Use a backslash (\) as the path separation character on Windows.

4 Save and close the migrate_oltp_6.0.2_to_6.0.3.properties file.

5 Set the Java environment variable (JRE 1.5 is required), where *JDK150_11* is your JDK version. For example:

■ UNIX:

export JAVA_HOME=/usr/local/bea10/*jdk150_11*

export PATH=$JAVA_HOME/bin: $PATH

■ Windows:

set JAVA_HOME=D: \bea*jdk150_11*

set PATH=%JAVA_HOME%\bin; %PATH%

6 Run following command to start migration:

ant -f migrate_oltp_6.0.2_to_6.0.3.xml

7 Select Option 1, Migrate OLTP Database.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

8 Review all log files for possible errors. If there were no errors, select Option 2, Post Migration Cleanup. You can ignore the following error (later in the migration process you compile the schema and validate the package):

Errors for PROCEDURE PR_LOAD_LATEST_STATEMENT:

LINE/COL ERROR

-------- ---

```
22/1 PL/SQL: SQL Statement ignored

34/9 PL/SQL: ORA-00904: "ERSF"."MEDIA_TYPE": invalid identifier
```

Migrating Oracle Self-Service E-Billing Version 6.0.2 OLAP to 6.0.3

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.2 OLAP to 6.0.3 for both UNIX and Windows.

This task is a step in "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156.

To migrate Oracle Self-Service E-Billing 6.0.2 OLAP to 6.0.3

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/
oracle/olap/migration/ebilling6.0.2_to_6.0.3 folder (or the
EDX_HOME\db\ebilling\oracle\olap\migration\ebilling6.0.2_to_6.0.3 folder on Windows).
In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Open the migrate_olap_6.0.2_to_6.0.3.properties file, and set the correct value for each property in this file:

   ■ **OLAP_SID.** OLAP instance name.

   ■ **OLAP_USER.** OLAP schema name.

   ■ **OLAP_PASSWD.** OLAP schema password.

3 Save and close the migrate_olap_6.0.2_to_6.0.3.properties file.

4 Run following command to start migration.

   ant -f migrate_olap_6.0.2_to_6.0.3.xml

5 Select Option 1, Migrate OLAP db.

   BUILD SUCCESSFUL displays on the console screen if migration is successful.

6 Review all log files for possible errors.

Compiling the Schema for the Oracle Self-Service E-Billing 6.0.3 OLTP and OLAP Databases

After successfully migrating the OLTP and OLAP databases, you must compile the schema. Follow these steps for both UNIX and Windows.

This task is a step in "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156.

To compile the schema for Oracle Self-Service E-Billing 6.0.3

1 After successfully migrating the OLAP and OLTP databases, you must compile the schema. On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

3 Log on to the OLAP schema using SQL*Plus (not as SYSDBA), and run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

4 After migrating to Oracle Self-Service E-Billing 6.0.3, you must create a new Command Center administrator user; the old user ID and password will not work.

For details on creating a new administrator user in the Command Center, see *Administration Guide for Oracle Self-Service E-Billing*.

Migrating Batch Reports from Oracle Self-Service E-Billing 6.0.2 to 6.0.3

Once you have successfully migrated from Oracle Self-Service E-Billing 6.0.2 to 6.0.3, you must copy the batch reports to the directory you want to use for reports in the 6.0.3 installation.

If you copy the reports to a different location than you used in version 6.0.2, you must update the batch report file location in the OLTP database. Also, if you install Oracle Self-Service E-Billing 6.0.3 in a location other than the default *EDX_HOME*, or if you create a custom subdirectory for storing batch reports (and do not use the default *output/reportapp*), you must update the related properties in the reporting.batch.xma.xml file.

This task is a step in .

To migrate your batch reports from Oracle Self-Service E-Billing 6.0.2 to version 6.0.3

1 On the application server, copy your existing batch reports to *EDX_HOME*\output\reportapp, where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing 6.0.3. Copying the batch reports to the same location in each version maintains the existing OLTP database links to the reports.

2 If you copy your batch reports to a different file location in version 6.0.3 than the one you used in 6.0.2, then you must update the batch report file location in the OLTP database. Log on to the OLTP schema using SQL*Plus, and run the following script:

update edx_rpt_batch_report set file_location = '*NEW_LOCATION*' || substr(file_location,length('*OLD_LOCATION*')+1) where file_location is not null;

```
SQL>commit;
```

where:

- *NEW_LOCATION* is the new location where you will store batch reports in Oracle Self-Service E-Billing 6.0.3.

- *OLD_LOCATION* is the old location where you stored batch reports in Oracle Self-Service E-Billing 6.0.2.

For example:

```
SQL> update edx_rpt_batch_report set file_location = '/export/home/oracle/
eBilling603/output/reportapp' || substr(file_location,length('/export/home/
oracle/eBilling/output/reportapp')+1) where file_location is not null;

SQL>commit;
```

3 If you install Oracle Self-Service E-Billing 6.0.3 in a directory other than the default *EDX_HOME*, or if you plan to use a customized batch report location (the default is *output/reportapp*), you must update the properties in the *EDX_HOME*/xma/config/com/edocs/common/reporting/reporting.batch.xma.xml file. Specify your installation root directory in the rootDir property, and specify the custom report subdirectory in the path property (report files are stored in *rootDir/path*):

```
<!-- use this property to override the default base, the system property for
edx.home

<property name="rootDir"><value>C:/edocs</value></property>

-->

<property name="path">

<value>/output/reportapp</value>

</property>
```

Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4

CAUTION: You must run the migration process on the Oracle Self-Service E-Billing 6.0.4 database server only.

To migrate from Oracle Self-Service E-Billing version 6.0.3 to 6.0.4, perform the following processes and tasks:

1 Back up your existing OLTP and OLAP Oracle Self-Service E-Billing databases.

For additional information, see Oracle Database 10*g* or Oracle Database 11*g* documentation on Oracle Technology Network.

2   Process all pending batch reports Oracle Self-Service E-Billing 6.0.3 before migrating.

For details about running batch reports, see *Administration Guide for Oracle Self-Service E-Billing*.

3   Run the Notifier job to send all email notifications before migrating.

For details about running the Notifier job, see *Administration Guide for Oracle Self-Service E-Billing*.

4   Perform the following installation tasks:

a   "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 18

b   "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 19. Choose Option 2, Oracle E-Billing and Migration Tools on the Product Features Screen.

c   "Configuring Log File Paths for Log4j" on page 21

d   "Adding Foreign Language Fonts for Localization" on page 23

5   Perform the following migration tasks:

a   "Migrating the OLTP Database from Oracle Self-Service E-Billing 6.0.3 to a New Character Set (Optional)" on page 165

b   "Migrating Oracle Self-Service E-Billing Version 6.0.3 OLTP to 6.0.4" on page 166

c   "Migrating Oracle Self-Service E-Billing Version 6.0.3 OLAP to 6.0.4" on page 167

d   "Compiling the Schema for the Oracle Self-Service E-Billing 6.0.4 OLTP and OLAP Databases" on page 168

e   "Migrating Batch Reports from Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 168

6   Configure your application server for Oracle Self-Service E-Billing 6.0.4:

■   **Oracle WebLogic**. Follow "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 45.

■   **IBM WebSphere.** Follow "Roadmap for Configuring IBM WebSphere for Oracle Self-Service E-Billing" on page 75.

7   "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 119

8   Uninstall Oracle Self-Service E-Billing version 6.0.3. For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 23.

This process is a step in the following roadmaps:

■   "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.0.4" on page 137

■   "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4" on page 171

■   "Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 182

■   "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 186.

Migrating the OLTP Database from Oracle Self-Service E-Billing 6.0.3 to a New Character Set (Optional)

The OLTP character set changed from WE8ISO8859P1 in Oracle Self-Service E-Billing 6.0.2 to AL32UTL8 in Oracle Self-Service E-Billing 6.0.3. Character set AL32UTL8 can store languages that are not available in character set WE8ISO8859P1.

If you have not already migrated your OLTP database to the new character set, and you want to support a language that is available only in the new character set, you must migrate your Oracle Self-Service E-Billing 6.0.3 OLTP database to the new character set. For more information about which languages can be stored in most common character sets, see Article ID 62421.1 on My Oracle Support.

To use the new character set, you must perform a full export and import of the OLTP database.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163.

To migrate the Oracle Self-Service E-Billing 6.0.3 OLTP database to the new character set

1 It is strongly recommended to create a full backup of your OLTP database and check your source OLTP database with Csscan to avoid data loss.

2 It is recommended to review the following notes on My Oracle Support:

■ **Article ID 745809.1**. Installing and configuring CSSCAN in Oracle Database 10*g* and Oracle Database 11*g*.

■ **Article ID 260192.1.** Changing the NLS_CHARACTERSET to AL32UTF8 / UTF8 (Unicode).

3 Verify the NLS_CHARACTERSET on the source database. Log on to the OLTP instance as SYSDBA, and run the following command:

```
SQL>select * from nls_database_parameters where parameter = 'NLS_CHARACTERSET';
```

4 Export the NLS_LANG set to the new character set:

■ UNIX:

```
export NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

exp system/manager@oltp_tnsname file=oltp_export.dmp FULL=Y
log=oltp_export.log
```

■ Windows:

```
set NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

exp system/manager@oltp_tnsname file=oltp_export.dmp FULL=Y
log=oltp_export.log
```

5 Create a new OLTP database with the new character set.

a In "Creating the Oracle Self-Service E-Billing Database Using Ant," follow Step 1 through Step 5 on page 35.

b Follow Step 12 through Step 15 on page 35 in the same procedure.

    c Select Option 4, Install Application Database - Create tablespace and user.

6 Create the database link, TAM_LINK, in the OLTP schema. Log on to the OLTP instance as SYSDBA, and run the following command, where *OLTP_Schema* is the name of the OLTP schema:

    SQL> GRANT CREATE DATABASE LINK TO *OLTP_Schema*;

7 Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows). Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following SQL script:

    SQL> DROP DATABASE LINK TAM_LINK;

    SQL>@ crt_db_link.sql *OLAP_User OLAP_Password OLAP_TNS_Name*

    SQL>exit

where:

■ *OLAP_User* is the name of the OLAP schema user.

■ *OLAP_Password* is the OLAP schema user's password.

■ *OLAP_TNS_Name* is the name of the OLAP instance.

8 Import the exported data into the new AL32UTF8 character set. Note that the NLS_LANG setting is uses the source character set:

■ UNIX:

    export NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

    imp system/manager@oltp_sid fromuser=oltp touser=oltp file=oltp_export.dmp log=oltp_import.log

■ Windows

    set NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

    imp system/manager@oltp_sid fromuser=oltp touser=oltp file=oltp_export.dmp log=oltp_import.log

Migrating Oracle Self-Service E-Billing Version 6.0.3 OLTP to 6.0.4

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.3 OLTP to 6.0.4 for both UNIX and Windows.

This task is a step in *"Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163*.

To migrate Oracle Self-Service E-Billing 6.0.3 OLTP to 6.0.4

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/ oracle/oltp/migration/ebilling 6.0.3_to_6.0.4 directory (or the *EDX_HOME*\db\ebilling\oracle\oltp\migration\ebilling6.0.3_to_6.0.4 directory on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Open the migrate_oltp_6.0.3_to_6.0.4.properties file, and set the correct value for each property in the file.

Property	Value
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password
OLTP_SID	OLTP instance name
SYS_PASSWD	System password

NOTE: Use a backslash (\) as the path separation character on Windows.

3 Save and close the migrate_oltp_6.0.3_to_6.0.4.properties file.

4 Run following command to start migration:

    ant -f migrate_oltp_6.0.3_to_6.0.4.xml

5 Select Option 1, Migrate OLTP Database.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

6 Review all log files for possible errors.

Migrating Oracle Self-Service E-Billing Version 6.0.3 OLAP to 6.0.4

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.3 OLAP to 6.0.4 for both UNIX and Windows.

This task is a step in .

To migrate Oracle Self-Service E-Billing 6.0.3 OLAP to 6.0.4

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/ oracle/olap/migration/ebilling6.0.3_to_6.0.4 folder (or the *EDX_HOME*\db\ebilling\oracle\olap\migration\ebilling6.0.3_to_6.0.4 folder on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Open the migrate_olap_6.0.3_to_6.0.4.properties file, and set the correct value for each property in this file:

■ **OLAP_SID.** OLAP instance name.

- ■ **OLAP_USER.** OLAP schema name.

- ■ **OLAP_PASSWD.** OLAP schema password.

3 Save and close the migrate_olap_6.0.3_to_6.0.4.properties file.

4 Run following command to start migration.

ant -f migrate_olap_6.0.3_to_6.0.4.xml

5 Select Option 1, Migrate OLAP db.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

6 If the build was successful, select Option 2, Repopulate Summary Tables. Query the edx_rpt_etl_log table to see log information on the data repopulation.

7 Review all log files for possible errors.

Compiling the Schema for the Oracle Self-Service E-Billing 6.0.4 OLTP and OLAP Databases

After successfully migrating the OLTP and OLAP databases, you must compile the schema. Follow these steps for both UNIX and Windows.

This task is a step in

To compile the schema for Oracle Self-Service E-Billing 6.0.4

1 After successfully migrating the OLAP and OLTP databases, you must compile the schema. On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

2 Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

3 Log on to the OLAP schema using SQL*Plus (not as SYSDBA), and run the following commands:

SQL>@ compile_schema.sql

SQL>exit;

Migrating Batch Reports from Oracle Self-Service E-Billing 6.0.3 to 6.0.4

Once you have successfully migrated from Oracle Self-Service E-Billing 6.0.3 to 6.0.4, you must copy the batch reports to the directory you want to use for reports in the 6.0.4 installation.

If you copy the reports to a different location than you used in version 6.0.3, you must update the
batch report file location in the OLTP database. Also, if you install Oracle Self-Service E-Billing 6.0.4
in a location other than the default *EDX_HOME*, or if you create a custom subdirectory for storing
batch reports and do not use the default *output/reportapp*, you must update the related properties
in the reporting.batch.xma.xml file.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163.

To migrate your batch reports from Oracle Self-Service E-Billing 6.0.3 to version 6.0.4

1 On the application server, copy your existing batch reports to *EDX_HOME*\output\reportapp,
where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing 6.0.4.
Copying the batch reports to the same location in each version maintains the existing OLTP
database links to the reports.

2 If you copy your batch reports to a different file location in version 6.0.4 than you used in 6.0.3,
you must update the batch report file location in the OLTP database. Log on to the OLTP schema
using SQL*Plus, and run the following script:

```
update edx_rpt_batch_report set file_location = 'NEW_LOCATION' ||
substr(file_location,length('OLD_LOCATION')+1) where file_location is not null;

SQL>commit;
```

where:

■ *NEW_LOCATION* is the new location where you will store batch reports in Oracle Self-Service
E-Billing 6.0.4.

■ *OLD_LOCATION* is the old location where you stored batch reports in Oracle Self-Service E-
Billing 6.0.3

For example:

```
SQL> update edx_rpt_batch_report set file_location = '/export/home/oracle/
eBilling604/output/reportapp' || substr(file_location,length('/export/home/
oracle/eBilling/output/reportapp')+1) where file_location is not null;

SQL>commit;
```

3 If you install Oracle Self-Service E-Billing 6.0.4 in a directory other than the default *EDX_HOME*,
or if you plan to use a customized batch report location (the default is *output/reportapp*), you
must update the properties in the *EDX_HOME*/xma/config/com/edocs/common/reporting/
reporting.batch.xma.xml file. Specify your installation root directory in the rootDir property,
and specify the custom report subdirectory in the path property (report files are stored in
rootDir/path):

```
<!-- use this property to override the default base, the system property for
edx.home

<property name="rootDir"><value>C:/edocs</value></property>

-->

<property name="path">
```

```
<value>/output/reportapp</value>

</property>
```

8 Migrating to Oracle Self-Service E-Billing Version 6.0.4 From Other Products

This chapter describes how to migrate Oracle Self-Service E-Billing to Version 6.0.4 from other electronic billing products. It includes the following topics:

- Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4 on page 171
- Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.0.4 on page 182
- Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.0.4 on page 186
- Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0. on page 186

Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4

This section describes the tasks necessary to migrate the Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing, version 6.0.4. Oracle eStatement Manager can consist of either of the following application sets:

- Oracle Siebel eStatement Manager 4.7 Fix Pack 1
- Oracle Siebel eStatement Manager 4.7 Fix Pack 1 with Oracle Siebel ePayment Manager 4.7

CAUTION: You must run the migration process on the Oracle Self-Service E-Billing 6.0.3 database server only.

To migrate from Oracle eStatement Manager to Oracle Self-Service E-Billing version 6.0.4 database, perform the following processes and tasks:

1 Verify that you have at least one month of billing cycle data loaded in the Oracle Self-Service E-Billing database using the ETL process described in *Administration Guide for Oracle Self-Service E-Billing*.

2 Verify that you have the ebilling-weblogic-10-6.0.3.ear file. This file is required for migrating the OLTP database and is packaged with LGPL libraries. For more information about LGPL libraries, see "Process of Repackaging the GNU Lesser General Public License" on page 67.

3 Use the Oracle Self-Service E-Billing Command Center to create all Oracle eStatement Manager applications, or Data Definition Names (DDNs), in Oracle Self-Service E-Billing 6.0.3. For details on creating application DDNs using Command Center, see *Administration Guide for Oracle Self-Service E-Billing*.

4 Follow these procedures to install the Oracle Self-Service E-Billing 6.0.3 database and create the schemas:

   a "Preparing to Configure the Oracle Self-Service E-Billing Database" on page 26

   b "Configuring Oracle Services" on page 26

   c "Choosing a Database Encryption Method" on page 29

   d "Creating the Oracle Self-Service E-Billing Database Using Ant" on page 33

   e "Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 38

5 Follow these procedures to set up the ETL and load your data:

   a "Verifying ETL Module System Requirements" on page 119

   b "Installing the Temporary Patch for Oracle Warehouse Builder 11g" on page 120

   c "Creating the Oracle Workflow Manager" on page 121

   d "Process of Installing the Oracle Warehouse Builder Repository" on page 122

   e "Installing the ETL Module" on page 131

   f "Running the ETL Loader Job Using Sample Data" on page 135

Determine the number of months of user transactional data you want to reside in the database. Load the summary data for the previous X months into the Oracle Self-Service E-Billing database using the ETL Loader Job.

6 "Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing 6.0.3" on page 173

7 (Optional) After migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing version 6.0.3, new statements appear in the application. If you want to retrieve historical statements for migrated users and display the statements in the original style as Oracle eStatement Manager, perform the following tasks to integrate flat-file functionality in Oracle Self-Service E-Billing:

   a "Repackaging the Flat-File Component" on page 175

   b "Loading Historical Bill Data Using ETL" on page 175

   c (Optional) If you want to display the historical bills using a format other than the view currently published, publish a new view (DDF, ALF, and HTML template files) in the Command Center in Oracle eStatement Manager now. For information about publishing view files, see *Administration Guide for Oracle Siebel eStatement Manager*.

   d "Configuring the Media Retrieval Functionality" on page 176

   e "Customizing the Media Retrieval Functionality" on page 179

   f "Customizing the Oracle Self-Service E-Billing User Interface to Render the Statement" on page 180

8

Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing 6.0.3

Follow these steps to migrate Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing 6.0.3 for both UNIX and Windows.

CAUTION: You must complete the steps outlined in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4" on page 171 before migrating Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing 6.0.3.

This task is a step in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4" on page 171.

To migrate Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing 6.0.3
1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/db/ebilling/ oracle/oltp/migration/eaSuite47Fix Pack 1_to_603 directory. You must run the migration process on the Oracle Self-Service E-Billing database server.

2 Open the migrate_easuite47_to_603.properties file, and set the correct value for each property in the file.

Property	Value
ORACLE_HOME	ORACLE HOME directory
ebilling_oltp_tnsname	OLTP tnsname
ebilling_oltp_user	OLTP schema user name
ebilling_oltp_paswd	OLTP schema password
ebilling_oltp_sid	OLTP instance name
ebilling_oltp_LISTEN_PORT	OLTP database listen port
ebilling_oltp_hostname	OLTP database host name or IP address
EBILLING_EAR_DIR	Directory containing the ebilling-weblogic-10-6.0.3.ear file after packaging LGPL libraries
easuite_hostname	Host name or IP address of the Oracle eStatement Manager database
easuite_db_port	Oracle eStatement Manager database listen port
easuite_sid	Oracle eStatement Manager instance name
easuite_user	Oracle eStatement Manager database schema name
easuite_paswd	Oracle eStatement Manager database schema password

NOTE: Use a backslash (\) as the path separation character on Windows.

3 Save and close the migrate_easuite47_to_603.properties file.

4 Set the Java environment variable (JRE 1.5 is required), where *JDK150_11* is your JDK version. For example:

■ UNIX:

export JAVA_HOME=/usr/local/bea10/*jdk150_11*

export   PATH=$JAVA_HOME/bin:$PATH

■ WINDOWS:

set JAVA_HOME=d:\beak*jdk150_11*

set PATH=%JAVA_HOME%\bin;%PATH%

5 Run the following command to start migration:

ant

6 Select Option 1 to install the migration-related objects.

7 Select Option 2 to migrate Oracle eStatement Manager data to the Oracle Self-Service E-Billing 6.0.3 database.

8   Review all log files for possible errors. If there were no errors, select Option 3, Post Migration Cleanup.

9   After migrating to Oracle Self-Service E-Billing 6.0.3, you must create a new Command Center administrator user; the old user ID and password will not work.

For details on creating a new administrator user in the Command Center, see *Administration Guide for Oracle Self-Service E-Billing*.

Repackaging the Flat-File Component

This section describes how to install, or repackage, the flat-file component into the Oracle eStatement Manager 4.7 Command Center EAR file.

This task is a step in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4" on page 171.

To repackage the flat-file component

1   Verify that the following requirements are met:

   ■   Oracle Self-Service E-Billing and Oracle eStatement Manager 4.7 are installed properly.

   ■   Oracle Warehouse Builder is installed.

   ■   The flat file integration component is available, installed as an OCS add-on component.

   ■   The Oracle eStatement Manager 4.7 database is installed properly.

2   Uncompress the ear-eStatement.ear file, found in the *EDX_HOME*/J2EEApps/WebLogic directory (or the *EDX_HOME*/J2EEApps/Websphere directory) to a folder such as ear-eStatement. In this path *EDX_HOME* is the Oracle eStatement Manager 4.7 installation directory.

3   Edit the application.xml file in the uncompressed folder to include the following line at the end of the following text:

   `<module><ejb>ejb-esmmediaretriever.jar</ejb></module>`

4   Copy the ejb-esmmediaretriever.jar to the uncompressed folder, and compress all contents under it into a new EAR file called ear-eStatement.ear.

5   Delete the previously deployed ear-eStatement.ear file.

6   Deploy the newly updated ear-eStatement.ear at the Oracle eStatement Manager 4.7 domain.

7   Restart the application server.

Loading Historical Bill Data Using ETL

You use Oracle Warehouse Builder to load the historical billing data from the Oracle eStatement Manager database into the Oracle Self-Service E-Billing database.

The historical billing data files must be processed by the ETL for Oracle Self-Service E-Billing as well as by the Indexer job in Oracle eStatement Manager.

ETL is required to load basic summary information (1000 - 2000 records) only. To generate basic reports off the historical bills, load 1000 - 3000 records; this does not include Call Detail Record level detail.

The input billing data file for ETL has required records only. Other information to load depends on your requirements:

■ To use the Payment feature, load the total_amount_due, statement_due_date, and statement_date fields into the edx_rpt_etl_statement_fact table in the OLAP database.

■ To use self-enrollment, load the company, account dimensions, and fact tables.

This task is a step in

To load historical bill data using ETL in Oracle Warehouse Builder

1 Convert the formatted bill data file from Oracle eStatement Manager to a file that is compliant with Oracle Self-Service E-Billing.

2 In record 1000 of the generated (compliant) bill data file, add ESM to the media_type field.

3 Run the ETL Loader job with the updated bill data file to load the statement summary data. For information on running the ETL Loader job, see *Administration Guide for Oracle Self-Service E-Billing.*

Configuring the Media Retrieval Functionality

You must modify the following Oracle Self-Service E-Billing configuration files as part of integrating flat-file functionality:

■ mediaretrieval.xma.xml

■ mediaretrieval.config.xma.xml

Follow these steps to modify the necessary JNDI parameters in the mediaretrieval.xma.xml configuration file.

This task is a step in

To modify the mediaretrieval.xma.xml configuration file

1 In the *EDX_HOME*/xma/config/modules/statement directory (or the *EDX_HOME*\xma\config\modules\statement directory on Windows), open the mediaretrieval.xma.xml file. In the path, *EDX_HOME* is the Oracle Self-Service E-Billing installation path.

2 Replace the following JNDI properties under the bean id="mediaRetrievalJndiB2C" section for the
B2C edition or bean id="mediaRetrievalJndiB2B" for the B2B edition with the value appropriate
for your installation.

Property	Value
java.naming.factory.initial	Class name of the initial context factory to use: ■ **Oracle WebLogic.** weblogic.jndi.WLInitialContextFactory ■ **IBM WebSphere.** com.ibm.websphere.naming.WsnInitialContextFactory
java.naming.provider.url	Location of the registry when the registry is being used as the initial context. Its format is protocol:// yourAppServerIP:Port. ■ **Oracle WebLogic**. The protocol must be t3; the port is your application server's port. ■ **IBM WebSphere.** The protocol is iiop; the port is your application server's bootstrap port.
java.naming.security.principal	Identity of the principal user for the authentication scheme; must be the user name defined in the application server.
java.naming.security.credentials	Principal's credentials for the authentication scheme; must be the user's password defined by java.naming.security.principal.

For example:

```
<bean id="mediaRetrievalJndiB2C" class="org.springframework.jndi.JndiTemplate">

    <property name="environment">

        <props>

            <prop
key="java.naming.factory.initial">weblogic.jndi.WLInitialContextFactory</prop>

            <prop key="java.naming.provider.url">t3://yourserverIP:Port</prop>

            <prop key="java.naming.security.principal">weblogic</prop>

            <prop key="java.naming.security.credentials">weblogic</prop>

        </props>

    </property>

</bean>
```

Follow these steps to configure the mediaretrieval.config.xma.xml configuration file with the name of the DDN defined in Oracle eStatement Manager as well as the report name. The MediaRetrieval service in Oracle Self-Service E-Billing reads this configuration to request a particular report for the specified DDN.

To modify the mediaretrieval.config.xma.xml configuration file

1   In the *EDX_HOME*/xma/config/modules/statement directory (or the *EDX_HOME*\xma\config\modules\statement directory on Windows), open the mediaretrieval.config.xma.xml file.

2   Modify the following properties:

■   **reportIdMap.** Sets the view names published in Oracle eStatement Manager Command Center. The key is the reportId defined in Oracle Self-Service E-Billing application. This reportId passes to the ImediaRetrieverService class to get the statement view.

■   **billerIdMap**. Defines the name of the DDN created in the Oracle eStatement Manager Command Center. Set the key according to the record 0000, BILLING SYSTEM field in the ETL billing file. For example, if the record in the billing file is defined as "0000|B2B |BS1 |, then, you must set the billerIdMap key as BS1.

For example:

```
<bean id="mediaRetrievalConfiguration" scope="singleton"
class="com.edocs.common.statement.mediaretrieval.MediaRetrievalConfig">

     <property name="reportIdMap">

       <map>

         <entry key="StatementSummary" value="HtmlDetail" />

         <entry key="ServiceSummary" value="SASummary" />

         <entry key="ServiceDetail" value="SADetail"/>

       </map>

     </property>

     <property name="billerIdMap">

       <map>

         <entry key="BS2" value="IndexerDDN" />

       </map>

     </property>

   </bean>
```

Customizing the Media Retrieval Functionality

The media retrieval feature gives the Oracle Self-Service E-Billing user remote access to Oracle eStatement Manager 4.7 to fetch a particular statement or history list.

Oracle eStatement Manager extracts and formats the statement information then sends the HTML stream to Oracle Self-Service E-Billing, which renders the statement.

Oracle Self-Service E-Billing provides the following public APIs for customizing the media retrieval functionality:

■ Media retrieval APIs

■ APIs to access statement data

■ Media retrieval configurations

This task is a step in

Using Media Retrieval APIs

The Media Retrieval module exposes the following APIs to get statement summary information and statement data:

■ **ImediaRetrieval**. Allows Oracle Self-Service E-Billing to work with Oracle eStatement Manager. ImediaRetrieval has two methods to retrieve the history list for an account number and retrieve the statement for an account number and bill period. The remote MediaRetrieval EJB is defined in Oracle Self-Service E-Billing, which implements this interface. The statement repository in Oracle eStatement Manager provides the actual implementation of the MediaRetrieval EJB (com.edocs.common.api.statement.mediaretrieval.ImediaRetrieval).

■ **IMediaRetrieverService**. Retrieves statement summary data and the statement view from any media type. Full path of the API is com.edocs.common.api.statement.mediaretrieval.IMediaRetrieverService.

Major implementation class: The main entry class that provides the media retrieval is servicecom.edocs.common.statement.mediaretrieval.ESMMediaRetrievalService.

APIs for Accessing Statement Data

All client classes can access the statement summary data and statement information retrieved from Oracle eStatement Manager. Two interfaces are exposed to store that data:

■ **IStatementSummary**. Provides methods to retrieve statement summary information from the Indexer table, such as account number, amount due, due date, and statement dates. The full path of this API is com.edocs.common.api.statement.mediaretrieval.IStatementSummary.

■ **IStatementInfo**. Provides methods to retrieve the statement stream for a particular DDN, account, view type, and view name. The full path of this API is com.edocs.common.api.statement.mediaretrieval.IStatementInfo.

Customizing the Oracle Self-Service E-Billing User Interface to Render the Statement

You must create a JSP file to invoke the flat file API and render the retrieved billing statement.

For reference, see the test JSP file found in the *EDX_HOME*/J2EEApps/ebilling/weblogic/ebilling-web-1.0-SNAPSHOT.war/mediaretrieval/test directory (or the *EDX_HOME*\J2EEApps\ebilling\weblogic\ebilling-web-1.0-SNAPSHOT.war\mediaretrieval\test directory on Windows). In the path, *EDX_HOME* is the Oracle Self-Service E-Billing installation path.

This task is a step in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.0.4" on page 171.

Examples of How to Get Account Summary Information

The following examples show the code you would add to the JSP file to display various types of account summary information on a bill:

■ Get the ImediaRetrievalService class:

```
………………………

LookupService lookup = LookupServiceFactory.getInstance();

 …………………………

IMediaRetrievalService service  = (IMediaRetrievalService)
lookup.getBean("edx:platform://modules/statement?id=iMediaRetrievalService");
```

■ Get account summary list by invoking ImediaRetrievalService class

```
…………………………

List<IStatementSummary> list = null;

int cols = 1;

int rows = 0;

Map<String,String> summaryMap = null;

IStatementSummary statementSummary = null;

try {

   list = service.getStatementSummary(user, account, accountType null);

   statementSummary = list.get(0);

   summaryMap = statementSummary.getAttribute();

} catch (Exception e) {

   // TODO Auto-generated catch block

   e.printStackTrace();
```

```
     }
```

....

■ You can access the account summary data by retrieving the properties of the
  IstatementSummary class.

Examples of How to Get Statement Data

The following examples show the code you would add to the JSP file to display various types of
statement data on a bill:

■ Get ImediaRetrievalService class

....

```
    LookupService lookup = LookupServiceFactory.getInstance();

    IMediaRetrievalService service  = (IMediaRetrievalService)
    lookup.getBean("edx:platform://modules/statement?id=iMediaRetrievalService");
```

....

■ Get a statement list by invoking ImediaRetrievalService class:

....

```
    IStatementInfo iStatementInfo = null;
```

....

```
    try {

      iStatementInfo = service.getStatement(user, account, accountType,
    statement_date, reportId,map);

        } catch (Exception e) {

            e.printStackTrace();

        }
```

....

■ Get the statement view by invoking the iStatementInfo class:

....

```
    response.getOutputStream().write(iStatementInfo.getStatementData());
```

....

Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.0.4

To migrate from Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing version 6.0.4, perform the following steps or processes.

CAUTION: You must run the migration process on the Oracle Self-Service E-Billing 6.0.3 database server only.

1 Verify that you have at least one month of billing cycle data loaded in the Oracle Self-Service E-Billing database using the Extract Transform Loading (ETL) process described in *Administration Guide for Oracle Self-Service E-Billing*.

2 Back up your existing Oracle Communications Billing Manager database.

3 Verify that you have the ebilling-weblogic-10-6.0.3.ear file. This file is required for migrating the OLTP database and is packaged with LGPL libraries.

For more information about LGPL libraries, see "Process of Repackaging the GNU Lesser General Public License" on page 67.

4 Use the Oracle Self-Service E-Billing Command Center to create all Oracle Communications Billing Manager application DDNs.

For details on creating application DDNs using Command Center, see *Administration Guide for Oracle Self-Service E-Billing*.

5 Follow these procedures to install the Oracle Self-Service E-Billing 6.0.3 database and create the schemas:

   a "Preparing to Configure the Oracle Self-Service E-Billing Database" on page 26

   b "Configuring Oracle Services" on page 26

   c "Choosing a Database Encryption Method" on page 29

   d "Creating the Oracle Self-Service E-Billing Database Using Ant" on page 33

   e "Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 38

6 Follow these procedures to set up the ETL and load your data:

   a "Verifying ETL Module System Requirements" on page 119

   b "Installing the Temporary Patch for Oracle Warehouse Builder 11g" on page 120

   c "Creating the Oracle Workflow Manager" on page 121

   d "Process of Installing the Oracle Warehouse Builder Repository" on page 122

   e "Installing the ETL Module" on page 131

f Determine the number of months of user transactional data you want to reside in the database. Load the summary data for the previous X months into the Oracle Self-Service E-Billing database using the ETL Loader Job.

For details on using the ETL Loader Job, see *Administration Guide for Oracle Self-Service E-Billing*.

7 Follow "Migrating Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3" on page 183.

8 Follow "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163.

Migrating Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3

Follow these steps to migrate Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3 for both UNIX and Windows.

CAUTION: You must complete the steps outlined in "Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 182 before migrating Communications Billing Manager 5.1.1 to Oracle Self-Service E-Billing 6.0.3.

This task is a step in "Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 182.

To migrate Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*\db\ebilling\oracle\oltp\migration\cbm511qf3_to_603 directory (or the *EDX_HOME*/db/ebilling/oracle/oltp/migration/cbm511qf3_to_603 directory on Windows). In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed. You must run the migration process on the Oracle Self-Service E-Billing database server.

2 Open the migrate_cbm_to_ebilling603.properties file, and set the correct value for each property in the file.

Property	Value
migration_type	Migration type: B2B, B2C, or B2B&B2C
cbmb2b_hostname	Host name of the Oracle Communications Billing Manager Business Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2b_db_port	Listener port of the Oracle Communications Billing Manager Business Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2b_sid	ORACLE_SID of the Oracle Communications Billing Manager Business Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2b_user	User name of the Oracle Communications Billing Manager Business Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2b_paswd	Password of the Oracle Communications Billing Manager Business Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2c_hostname	Host name of the Oracle Communications Billing Manager Consumer Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2c_db_port	Listener port of the Oracle Communications Billing Manager Consumer Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2c_sid	ORACLE_SID of the Oracle Communications Billing Manager Consumer Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2c_user	User name of the Oracle Communications Billing Manager Consumer Edition database. Specify only if migration_type is B2B, or B2B and B2C.
cbmb2c_paswd	Password of the Oracle Communications Billing Manager Consumer Edition database. Specify only if migration_type is B2B, or B2B and B2C.
ORACLE_HOME	Oracle home on the target (OLTP) database server.
ebilling_oltp_tnsname	Oracle Self-Service E-Billing 6.0.3 OLTP database tnsname
ebilling_oltp_user	Oracle Self-Service E-Billing 6.0.3 OLTP user
ebilling_oltp_paswd	Oracle Self-Service E-Billing 6.0.3 OLTP password
ebilling_oltp_sid	Oracle Self-Service E-Billing 6.0.3 OLTP ORACLE_SID

Property	Value
ebilling_oltp_LISTEN_PORT	Oracle Self-Service E-Billing 6.0.3 OLTP listener port
ebilling_oltp_hostname	Host name of the Oracle Self-Service E-Billing 6.0.3 OLTP database
EBILLING_EAR_DIR	Directory that contains the ebilling-weblogic-10-6.0.3.ear file after packaging the LGPL libraries

NOTE: Use a backslash (\) as the path separation character on Windows.

3 Save and close migrate_cbm_to_ebilling603.properties file.

4 Set the java environment variable (JRE 1.5 is required), where *JDK150_11* is your JDK version. For example:

■ UNIX:

export   JAVA_HOME=/usr/local/bea10/*jdk150_11*

export   PATH=$JAVA_HOME/bin:$PATH

■ WINDOWS:

set   JAVA_HOME=d:\bea*jdk150_11*

set PATH=%JAVA_HOME%\bin;%PATH%

5 Run the following command to start migration:

ant

6 Select Option 1 to install the migration-related objects.

7 Choose the option appropriate for your product. To merge both the Oracle Communications Billing Manager Business and Consumer Editions into one database schema, complete both of the following steps:

■ Business Edition: Select Option 2 if you are migrating data for the Business Edition of Oracle Communications Billing Manager into Oracle Self-Service E-Billing 6.0.3. Then from the submenu for B2B data migration, select Option 1 to migrate user profile and payment data; select Option 2 to migrate hierarchy data. Choose Option 3 to return to the Main Menu.

■ Consumer Edition: Select Option 3 if you are migrating data for the Consumer Edition of Oracle Communications Billing Manager into Oracle Self-Service E-Billing 6.0.3. Choose Option 3 to return to the Main Menu.

NOTE: If you merge both Business and Consumer Editions into one schema, the ant script checks for data conflicts; you must resolve any conflicts before continuing with the merge.

8 Select Option 4 to run the java process and perform post-migration operations.

9 Review all log files for possible errors. If there were no errors, select Option 5, Post Migration Cleanup.

10 After migrating to Oracle Self-Service E-Billing 6.0.3, you must create a new Command Center administrator user; the old user ID and password will not work.

For details on creating a new administrator user in the Command Center, see *Administration Guide for Oracle Self-Service E-Billing*.

Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.0.4

To migrate from Oracle Communications Billing Analytics 5.1.1 Quick Fix 3 to Oracle Self-Service E-Billing version 6.0.4, perform the following processes:

1 "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 186

2 "Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1" on page 138

3 "Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2" on page 146

4 "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.2 to 6.0.3" on page 156

5 "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 163

Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0.

To migrate from Oracle Communications Billing Analytics 5.1.1 or 5.1.1 QF2 database to Oracle Self-Service E-Billing 6.0, perform the following tasks:

1 Back up your OLTP and OLAP Oracle Self-Service E-Billing databases.

2 "Migrating the Oracle Communications Billing Analytics 5.1.1 Database from Oracle9i to Oracle Database 10g (UNIX and Windows)" on page 187

3 "Migrating the Oracle Communications Billing Analytics 5.1.1 eStatement Component to Oracle Self-Service E-Billing 6.0 on UNIX and Windows" on page 188

4 "Installing the Payment Database on an Existing Oracle Communications Billing Analytics 5.1.1 OLTP Database" on page 189

5 "Migrating the Oracle Communications Billing Analytics 5.1.1 OLTP Database to Oracle Self-Service E-Billing 6.0 OLTP on UNIX" on page 191 or Migrating the Oracle Communications Billing Analytics 5.1.1 OLTP Database to Oracle Self-Service E-Billing 6.0 OLTP on Windows on page 192

6 "Migrating the Oracle Communications Billing Analytics 5.1.1 OLAP Schema to Oracle Self-Service E-Billing 6.0 on UNIX" on page 194 or Migrating the Oracle Communications Billing Analytics 5.1.1 OLAP Schema to Oracle Self-Service E-Billing 6.0 on Windows on page 195

7 "Compiling the Schema After Migrating the OLTP and OLAP Oracle Communications Billing Analytics Databases" on page 196

8 "Populating the Role Authorization Database Table" on page 197

This process is a step in "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.0.4" on page 186.

Migrating the Oracle Communications Billing Analytics 5.1.1 Database from Oracle9*i* to Oracle Database 10*g* (UNIX and Windows)

This topic describes how to migrate the Oracle Communications Billing Analytics 5.1.1 database from Oracle9*i* to Oracle Database 10*g* (10.2.0.2) on UNIX or Windows.

This task is a step in "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 186.

To migrate the Oracle Communications Billing Analytics 5.1.1 QF2 database from Oracle9i to Oracle Database 10g (10.2.0.2) on UNIX and Windows

1 On UNIX, log in as the ORACLE user for migration activity. Export your existing Oracle Communications Billing Analytics 5.1.1 OLAP and OLTP databases from Oracle9*i*:

```
exp system/manager@olap_sid file=olap_export.dmp FULL=Y

exp system/manager@oltp_sid file=oltp_export.dmp FULL=Y
```

2 Install Oracle Database 10*g* (10.2.0.2) to upgrade your database software.

3 Create a new Oracle Database 10*g* database instance for OLAP and OLTP to migrate your databases. Follow the procedures in this guide to create a new instance for both OLAP and OLTP.

4 Create tablespaces and a user in the new OLAP and OLTP databases. Follow the steps in this guide to create tablespaces and users in the new OLAP and OLTP databases.

5 Create the database link (TAM_LINK) in the OLTP schema if it was not successfully created during the import:

  a Log on to the OLTP instance as SYSDBA and run the following command, where *OLTP_Schema* is the name of the OLTP schema:

```
SQL> GRANT CREATE DATABASE LINK TO OLTP_Schema;
```

  b Go to the *EDX_HOME*/db/ebilling/oracle directory (or the *EDX_HOME*\db\ebilling\oracle directory on Windows), where *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing.

    c Log on to the OLTP schema using SQL*Plus (not as SYSDBA), and run the following script, providing the three input parameters:

        SQL> DROP DATABASE LINK TAM_LINK;

        SQL>@ crt_db_link.sql *OLAP_User OLAP_Password OLAP_TNS_Name*

    where:

      ❏ *OLAP_User* is the name of the OLAP schema user.

      ❏ *OLAP_Password* is the OLAP schema user's password.

      ❏ *OLAP_TNS_Name* is the name of the OLAP instance.

6 Import your Oracle9*i* database into the new database:

    imp system/manager@olap_sid fromuser=olap touser=olap file=olap_export.dmp

    imp system/manager@oltp_sid fromuser=oltp touser=oltp file=olap_export.dmp

Migrating the Oracle Communications Billing Analytics 5.1.1 eStatement Component to Oracle Self-Service E-Billing 6.0 on UNIX and Windows

This topic describes how to migrate the Oracle Communications Billing Analytics 5.1.1 eStatement component to Oracle Self-Service E-Billing 6.0 on UNIX or Windows.

This task is a step in

To migrate the Oracle Communications Billing Analytics 5.1.1 eStatement Component to Oracle Self-Service E-Billing 6.0 on UNIX and Windows

1 On UNIX, log in as the ORACLE user for migration activity. Go to the following directory, where *EDX_HOME* is the location where Oracle Communications Billing Analytics is installed:

    ■ **UNIX.** *EDX_HOME*/db/eStatement/oracle/migration

    ■ **Windows.** *EDX_HOME*\db\eStatement\oracle\migration

2 Edit the migrate.properties file, and modify the parameters in the following statements for your OLTP database environment.

    This file controls the eStatement database migration.

Property	Value
DB_SID=EDX44	Change the OLTP instance name.
DB_USERNAME=edx_dba	Change the OLTP schema user name.
DB_PASSWORD=edx	Change the OLTP schema password.

3 Also in the migrate.properties file, edit the following statements to specify the appropriate file location:

■ Large data tablespace file location: LARGE_DB_EDX_DATA_TB_FILE_LOC=/export/home/oracle/oradata

■ Large index tablespace file location: LARGE_DB_EDX_INDX_TB_FILE_LOC=/export/home/oracle/oradata

■ Medium data tablespace file location: MEDIUM_DB_EDX_DATA_TB_FILE_LOC=/export/home/oracle/oradata

■ Medium index tablespace file location: MEDIUM_DB_EDX_INDX_TB_FILE_LOC=/export/home/oracle/oradata

4 Go to your eStatement database home directory, for example:

■ **UNIX.** *EDX_HOME*/db/eStatement/oracle

■ **Windows.** *EDX_HOME*\db\eStatement\oracle

5 Run the following Ant target to migrate.

```
ant migrate
```

The install-migrate target migrates eStatement instances with the SIDs specified in the properties file. The following message appears:

get_backup_confirm:

Warning We strongly advise a full backup of your existing database before applying the migration. Do you have a backup (Y,y,N,n)

6 Enter Y if you have a backup, or N if you do not.

7 Enter Y to continue the migration process.

8 Select Option 1, Migrate from 4.3.0.0 Migrate to 4.7.0.0.

Installing the Payment Database on an Existing Oracle Communications Billing Analytics 5.1.1 OLTP Database

You must create an Payment database on your existing Oracle Communications Billing Analytics 5.1.1 OLTP database before migrating OLTP.

This task is a step in .

To install the Payment database on an existing Oracle Communications Billing Analytics 5.1.1 OLTP database

1 In the payusr.properties file, make sure that you give the same OLTP database SID, user name, and password that you specified in the edxadmin.properties file of the eStatement installation. The file payusr.properties contains configuration parameters specific to each installation and is used by the Ant script that installs the Oracle Self-Service E-Billing Payment database.

The payusr.properties file is located in the *EDX_HOME*/db/ePayment/oracle directory (or the *EDX_HOME*\db\epayment\oracle directory on Windows), where *EDX_HOME* is the location where Oracle Communications Billing Analytics is installed.

2 On UNIX, log in as the ORACLE user for migration activity. Go to the directory location of the Payment component installation files in your software installation:

■ **UNIX.** *EDX_HOME*/db/ePayment/oracle

■ **Windows.** *EDX_HOME*\db\epayment\oracle

3 If you have not configured the Apache Ant environment, do so now:

■ UNIX:

export ANT_HOME=/opt/apache-ant-1.6.5

export PATH=$ANT_HOME/bin:$PATH

If you are on Oracle WebLogic, run this command, where *JDK150_11* is your specific JDK version:

export JAVA_HOME=$WEBLOGIC_HOME/*JDK150_11*

If you are on IBM WebSphere, run this command:

export $JAVA_HOME=$WS_HOME/java

Also, on all application servers, run this command:

export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$PATH

■ Windows:

set ANT_HOME=C:\apache-ant-1.6.5

set PATH=%PATH%;%ANT_HOME%\bin

set JAVA_HOME= %WEBLOGIC_HOME%*JDK150_11*

where *JDK150_11* is your specific JDK version.

4 Run the build script by typing ant.

By default Ant picks up the build.xml file in the current directory.

5 From the top level Main Menu select Option 1, Install Application Database I.

6 Type ant again, and from the top level Main Menu select Option 2, Install Application Database II.

7 Type ant again, and from the top level Main Menu select Option 3, Initial Data Population.

Migrating the Oracle Communications Billing Analytics 5.1.1 OLTP Database to Oracle Self-Service E-Billing 6.0 OLTP on UNIX

This topic describes how to migrate the Oracle Communications Billing Analytics 5.1.1 OLTP database to Oracle Self-Service E-Billing 6.0 OLTP on UNIX.

This task is a step in

To migrate the Oracle Communications Billing Analytics 5.1.1 OLTP database to Oracle Self-Service E-Billing 6.0 on UNIX

1 Log in as the ORACLE user for migration activity on UNIX. Go to the *EDX_HOME*/db/ebilling/ oracle/oltp/migration/511_to_6.0 directory. In the path, *EDX_HOME* is the Oracle Self-Service E-Billing installation directory.

2 Verify that the following files exist:

■ migrate_oltp_511_to_6.0.sh

■ migrate_oltp_511_to_6.0.sql

3 Run the following commands at the shell prompt:

chmod 777 migrate_oltp_511_to_6.0.sh

./migrate_oltp_511_to_6.0.sh

4 Provide the correct values for your environment.

Field	What to Enter
Database ID	Instance name, such as OLTP
Database Username	Schema name, such as OLTP
Database Password	Schema Password, such as OLTP
SYS Password	Password of the SYSDBA, such as change_on_install

5 Check for errors in the following log files:

■ db_oltp_migrate_511_6.0.log

■ migrate_oltp_511_to_6.0.log

Note that after running the OLTP migration script, migrate_oltp_511_to_6.0.sh, the following error messages appears in the log file. Ignore these errors. After compiling the schema, these errors will disappear:

Warning: Package Body created with compilation errors.

Errors for PACKAGE BODY OLTP_PROD_LOADER_PKG:

LINE/COL ERROR

-------- --

400/6 PL/SQL: SQL Statement ignored

437/5 PL/SQL: ORA-00904: "SERVICE_FACT"."ETL_KEY": invalid identifier

Warning: Package Body created with compilation errors.

Errors for PACKAGE BODY PKG_COPY_HIERARCHY:

LINE/COL ERROR

-------- --

316/2 PL/SQL: Statement ignored

316/2 PLS-00905: object OLTP6.PR_COPY_ACCOUNT_WSPACE is invalid

Migrating the Oracle Communications Billing Analytics 5.1.1 OLTP Database to Oracle Self-Service E-Billing 6.0 OLTP on Windows

This topic describes how to migrate the Oracle Communications Billing Analytics 5.1.1 OLTP database to Oracle Self-Service E-Billing 6.0 OLTP on Windows.

This task is a step in .

To migrate the Oracle Communications Billing Analytics 5.1.1 OLTP database to
Oracle Self-Service E-Billing 6.0 OLTP database on Windows

1 Go to the migration directory, where *EDX_HOME* is the Oracle Self-Service E-Billing installation directory:

   EDX_HOME\db\ebilling\oracle\oltp\migration\511_to_6.0

2 Make sure the migrate_oltp_511_to_6.0.sql file exists.

3 Run the userlock.sql command from SQL*Plus, providing the exact path of ORACLE_HOME:

```
C:\set ORACLE_SID=oltp

C:\sqlplus  sys/sys_password as sysdba

SQL> @ ORACLE_HOME\rdbms\admin\userlock.sql

SQL> exit;
```

where:

- *sys_password* is the password of the sys user.

- *ORACLE_HOME* is the directory where the Oracle database software is installed.

4 Run the migration script migrate_oltp_511_to_6.0.sql:

```
C:\set ORACLE_SID=oltp

C:\sqlplus oltp/oltp

SQL> @ migrate_oltp_511_to_6.0.sql

SQL> exit;
```

5 Verify that there were no errors in the migrate_oltp_511_to_6.0.log file.

Note that after running the OLTP migration script, migrate_oltp_511_to_6.0.sql, the following error messages appear in the log file. Ignore these errors. After compiling the schema, these errors will disappear.

Warning: Package Body created with compilation errors.

Errors for PACKAGE BODY OLTP_PROD_LOADER_PKG:

LINE/COL ERROR

-------- --

400/6 PL/SQL: SQL Statement ignored

437/5 PL/SQL: ORA-00904: "SERVICE_FACT"."ETL_KEY": invalid identifier

Warning: Package Body created with compilation errors.

Errors for PACKAGE BODY PKG_COPY_HIERARCHY:

LINE/COL ERROR

-------- --

316/2 PL/SQL: Statement ignored

316/2 PLS-00905: object OLTP6.PR_COPY_ACCOUNT_WSPACE is invalid

Migrating the Oracle Communications Billing Analytics 5.1.1 OLAP Schema to Oracle Self-Service E-Billing 6.0 on UNIX

These instructions describe how to migrate the Oracle Communications Billing Analytics 5.1.1 OLAP schema to Oracle Self-Service E-Billing 6.0 on UNIX.

This task is a step in "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 186.

To migrate the Oracle Communications Billing Analytics 5.1.1 OLAP schema to Oracle Self-Service E-Billing 6.0 on UNIX

1 Log in as the ORACLE user for migration activity on UNIX. Go to the
   EDX_HOME\db\oracle\olap\migration\511_to_6.0 directory. In the path, *EDX_HOME* is the
   Oracle Self-Service E-Billing installation directory.

2 Verify that the following files exist:

   ■ migrate_olap_511_to_6.0.sh

   ■ migrate_olap_511_to_6.0.sql

   ■ Create_Tables.sql

   ■ Load_Data.sql

   ■ Create_Sequences.sql

   ■ Create_Procedure.sql

   ■ Create_Views.sql

   ■ Create_Indexes.sql

   ■ Create_Constraints.sql

   ■ dblink.sql

3 Log in to the database as SYSDBA:

   export ORACLE_SID=myolap

   sqlplus "/as sysdba"

4 Grant the create materialized view privilege to the user you are migrating, where *User Name* is
   the name of the user:

   SQL> grant create materialized view to *User Name*;

5 Replace the following parameters in the dblink.sql file with the values appropriate for your
   installation.

Parameter	Value
OLTP_USERNAME	The name of the OLTP schema user.

Parameter	Value
OLTP_PASSWORD	The password of the OLTP schema user.
OLTP_SID	The name of the OLTPinstance.

6 Log in to the OLAP database as the user who you are migrating, Run the following script to create the database link name OLTP_LINK. Do not run the dblink.sql script as SYSDBA:

dblink.sql

7 Execute the following commands at the shell prompt:

chmod 777 migrate_oltp_511_to_6.0.sh

./migrate_oltp_511_to_6.0.sh

8 Provide the correct values for your environment.

Field	What to Enter
OLAP Database SID	OLAP instance name, such as OLAP
OLAP Database Username	OLAP schema name, such as OLAP
OLAP Database Password	OLAP Schema Password, such as OLAP
OLAP SYS Password	Password of the SYSDBA, such as change_on_install
OLTP TNS Name	Enter the TNS name for OLTP, such as OLTP
OLTP Database Username	OLTP schema name, such as OLTP
OLTP Database Password	OLTP schema password, such as OLTP

9 Check for errors in the following log files:

■ db_olap_migrate_511_6.0.log

■ migrate_olap_511_to_6.0.log

Migrating the Oracle Communications Billing Analytics 5.1.1 OLAP Schema to Oracle Self-Service E-Billing 6.0 on Windows

This topic describes how to migrate the Oracle Communications Billing Analytics 5.1.1 OLAP schema to Oracle Self-Service E-Billing 6.0 on Windows.

This task is a step in *"Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 186*.

To migrate the Oracle Communications Billing Analytics 5.1.1 OLAP schema to Oracle Self-Service E-Billing 6.0 on Windows

1 Go to the *EDX_HOME*\db\oracle\olap\migration\511_6.0 directory, and make sure the migrate_olap_511_to_6.0.sql file exists. In the path, *EDX_HOME* is the Oracle Self-Service E-Billing installation directory.

2 Log in to the database as SYSDBA:

export ORACLE_SID=myolap

sqlplus "/as sysdba"

3 Grant the create materialized view privilege to the user you are migrating, where *User Name* is the name of the user:

SQL> grant create materialized view to *User Name*;

4 Replace the following parameters in the dblink.sql file with the values appropriate for your installation.

Parameter	Value
OLTP_USERNAME	The name of the OLTP schema user
OLTP_PASSWORD	The password of the OLTP schema user
OLTP_TNS_NAME	The TNS name of the OLTP instance

5 Log in to the OLAP database as the user you are migrating. Run the following script to create the database link name OLTP_LINK. Do not run the dblink.sql script as SYSDBA:

dblink.sql

6 Run the migrate_olap_511_to_6.0.sql script from SQL*Plus:

C:\set ORACLE_SID=olap

C:\sqlplus oltp/oltp

SQL> @ migrate_olap_511_to_6.0.sql

SQL> exit;

7 Make sure there are no errors in the migrate_olap_511_to_6.0.log file.

Compiling the Schema After Migrating the OLTP and OLAP Oracle Communications Billing Analytics Databases

After successfully migrating the OLTP and OLAP Oracle Communications Billing Analytics databases, you must compile the schema.

This task is a step in "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 186.

To compile the schema after migrating the OLTP and OLAP Oracle Communications Billing Analytics databases

1 On UNIX, log in as the ORACLE user for migration activity. Go to the *EDX_HOME*/ebilling/db/ebilling/oracle directory.

2 Log in to the OLTP schema using SQL*Plus (not as SYSDBA).

3 Run the following commands:

    SQL>@ compile_schema.sql

    SQL>exit;

4 Log in to the OLAP schema using SQL*Plus (not as SYSDBA).

5 Run the following commands:

    SQL>@ compile_schema.sql

    SQL>exit;

Populating the Role Authorization Database Table

After migrating Oracle Communications Billing Analytics 5.1, you must populate the Privilege Order column in the Role Authorization database table with the role values you have defined for your organization.

By default, the values in the OLTP role authorization table are set to null. This is currently causing an error when you log in to the Billing and Payment end-user application.

This task is a step in "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 186.

Index