Installation Guide for Oracle Self-Service E-Billing

Version 6.1.1

April 2013

ORACLE

Contents

Chapter 4: Configuring Oracle WebLogic

Chapter 6: Migrating to Oracle Self-Service E-Billing Version 6.1.1

1 What's New in This Release

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1.1

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 6.1.1

Topic	Description
"Installing Digital Accessible Information System (DAISY)-Related Software" on page 21	New topic. Describes how to install the software required to provide billing statements in DAISY (Digital Accessible Information System) audio file format.
"Configuring Oracle Services" on page 27 "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29 "Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 37 "Process of Implementing TDE Column Encryption" on page 46 "Specifying the Oracle Wallet Location in the SQLnet.ora File" on page 49 "Setting the Master Encryption Key Using the Ant Encryption Menu" on page 50 "Opening the Oracle Wallet Using the Ant Encryption Menu" on page 51 "Process of Implementing TDE Tablespace Encryption" on page 52 "Configuring the Oracle Self-Service E-Billing Database on Oracle Exadata" on page 56	Modified topics. Updated the database installation and encryption procedures using a single database instance with both the OLAP and OLTP schemas.

Table 1. What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1.1

Topic	Description
"Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74 "Creating the Oracle WebLogic Domain and Managed Servers for the Billing and Payment Application on a Cluster" on page 74 "Defining the Oracle WebLogic Cluster Environment for the Billing and Payment Application" on page 77 "Configuring JDBC Resources for the Billing and Payment Application on an Oracle WebLogic Cluster" on page 78 "Starting the Oracle WebLogic Cluster" on page 115	New topics. Describe how to configure the Billing and Payment application on an Oracle WebLogic cluster.
"Creating the Oracle WebLogic Domain for the Billing and Payment Application" on page 63 "Creating the Oracle WebLogic Domain for Command Center" on page 82 "Creating the Oracle WebLogic Domain for the Customer Service Representative Application" on page 90 "Configuring and Starting Scheduler on Oracle WebLogic and UNIX" on page 117 "Configuring and Starting Scheduler on Oracle WebLogic on Windows" on page 118	Modified topics. Updated for Oracle WebLogic version 12.1.
"Creating the Oracle Workflow Manager" on page 122 "Tuning the OLAP Database" on page 123 "Creating the Oracle Warehouse Builder Repository Owner" on page 124 "Creating the Oracle Warehouse Builder Repository User" on page 125 "Registering Users in Oracle Warehouse Builder 11g" on page 126 "Creating Database Modules" on page 127 "Creating a Process Flow Module" on page 128 "Registering Locations" on page 129 "Installing the ETL Module" on page 130	Modified topics. Updated the ETL installation procedures for using a single database instance with both the OLAP and OLTP schemas.

Table 1. What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1.1

Topic	Description
"Creating a Bootstrap Administrator User for Oracle Self-Service E-Billing" on page 56	New topic. Describes how to create a bootstrap administrator user for Oracle Self-Service E-Billing during the installation process. It is no longer preconfigured with Oracle Self-Service E-Billing.
"Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1" on page 137 "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 196	Modified topics. Replaced the incremental processes for migrating Oracle Self-Service E-Billing version 6.0.2 to 6.1.1 with a direct migration path.
"Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1" on page 153	New topic. Describes how to migrate from Oracle Self-Service E-Billing version 6.0.2 directly to 6.1.1.
"Process of Migrating Oracle Self-Service E-Billing from 6.1 to 6.1.1" on page 173	New topic. Describes how to migrate Oracle Self-Service E-Billing from version 6.1 to 6.1.1.
"Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181 "Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 192	Modified topics. Added the steps for migrating legacy products to Oracle Self-Service E-Billing version 6.1.1.
Migrating from Oracle Self-Service E-Billing 6.0.2 to 6.0.3	References to migration from version 6.0.2 to 6.0.3 were removed from the book because the process has been replaced with a direct migration path to version 6.1.1.

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1, Rev. A

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 6.1, Rev. A

Topic	Description
"Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61	Modified topic. Added the process of configuring the Web services applications.

Table 2. What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1, Rev. A

Topic	Description
"Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications" on page 97	New topic. Describes the process of configuring the Web services applications.
"Process of Repackaging the GNU Lesser General Public License" on page 108	Modified topic. Added the requirement for the Web services applications.

What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1

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 6.1

Topic	Description
"Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 17	Modified topic. Added the option to install the Utility application.
"Roadmap for Configuring the Oracle Self-Service E-Billing Database" on page 25 Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node) on page 29 "Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 37	Modified topics. Updated the procedures for configuring the Oracle Self-Service E-Billing database for Oracle 11*g*, version 11.2.0.3.
"Process of Implementing TDE Column Encryption" on page 46 "Process of Implementing TDE Tablespace Encryption" on page 52 "Specifying the Oracle Wallet Location in the SQLnet.ora File" on page 49 "Setting the Master Encryption Key Using the Ant Encryption Menu" on page 50 "Opening the Oracle Wallet Using the Ant Encryption Menu" on page 51	Modified topics. Updated the encryption procedures for Oracle 11*g*, version 11.2.0.3.
"Configuring JDBC Resources for the Billing and Payment Application" on page 66	Modified topic. Updated the data source configuration properties.
"Setting the Global Configuration Properties for the Billing and Payment Application" on page 70	Modified topic. Added the email formatting properties emailImgSrc and emailHtmlCSS for Oracle WebLogic.
"Setting Up Prenote Functionality" on page 73	New topic. Describes how to set up prenote functionality in Oracle WebLogic.

Table 3. What's New in Installation Guide for Oracle Self-Service E-Billing, Version 6.1

Topic	Description
"Configuring a JMS Persistence Store for the Command Center" on page 88 "Defining the Oracle WebLogic Environment for the Customer Service Representative Domain" on page 91	Modified topics. Updated the procedures for configuring Oracle WebLogic and Apache.
"Verifying ETL Module System Requirements" on page 121 "Creating Database Modules" on page 127 "Installing the ETL Module" on page 130 "Running the ETL Loader Job Using Sample Data" on page 133	Modified topics. Updated the ETL procedures for the new database partitioning feature and changes to the Oracle Self-Service E-Billing database.
"Process of Migrating Oracle Self-Service E-Billing from 6.0.4 to 6.1" on page 168 "Migrating Oracle Self-Service E-Billing Version 6.0.4 OLTP to Version 6.1" on page 169 "Migrating Oracle Self-Service E-Billing Version 6.0.4 OLAP to Version 6.1" on page 170 "Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.0.4 to 6.1" on page 171	New topics. Added the procedures for migrating Oracle Self-Service E-Billing version 6.0.4 to version 6.1.
"Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181 "Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 192 "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 196	Modified topics. Added the steps for migrating legacy products to Oracle Self-Service E-Billing version 6.1.

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 6.1.1

This topic describes the tasks necessary to install a new implementation of Oracle Self-Service E-Billing, version 6.1.1.

CAUTION: If you are migrating to Oracle Self-Service E-Billing 6.1.1, then do not use this roadmap. Follow the particular roadmap appropriate for migrating your current product and version. If you perform a full installation on an existing implementation instead of the proper migration, then the migration will fail and the database will become unstable from deployment of the target schema.

For information about migrating, see "Migrating to Oracle Self-Service E-Billing Version 6.1.1" on page 137 or "Migrating to Oracle Self-Service E-Billing Version 6.1 from Other Products" on page 181.

To install a new implementation of Oracle Self-Service E-Billing 6.1.1, perform the following processes and tasks:

1. "Preparing Your Platform" on page 16
2. "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 16
3. "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 17

   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 19
5. "Adding Foreign Language Fonts to Your Application Server" on page 21

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

7 "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61

8 "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121

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 6.1.1" on page 15.

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 62.

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

   For details, see your server documentation.

4 For distributed database 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 Software Delivery Cloud, 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, then either the download was interrupted or data was corrupted, and you must start the download again.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15.

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
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`

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

- **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 directory:

- **UNIX.** *EDX_HOME*/J2EEApps
- **Windows.** *EDX_HOME*\J2EEApps

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15.

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

1. Start InstallAnywhere in UI mode:
   - **UNIX.** Log in using the user and group name of the Oracle Self-Service E-Billing application server owner, such as edxadmin:edxadmin. Make sure DISPLAY is set, then type the following command:

     ```
     ./eBilling.bin
     ```

     To start InstallAnywhere in Console Mode, type the following command then follow the on-screen instructions:

     ```
     ./eBilling.bin -i console
     ```

   - **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, then 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, then 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. Select the vertical application to install, either Telco or Utility.
9. On the Choose Product Features screen, choose one of the following:
   - If you are installing Oracle Self-Service E-Billing for the first time, then select Option 1, Oracle E-Billing, and click Next.

- If you are migrating to version, then 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 6.1.1" on page 137.

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

11 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, then a congratulatory message appears with the directory that contains the Oracle Self-Service E-Billing components. Click Next.

12 Click Done to exit the installer.

If the installation fails, then 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, then make sure your values are consistent in all servers for your installation, or Oracle Self-Service E-Billing will not function properly.

Configuring Log File Paths for Log4j

By default, Oracle Self-Service E-Billing writes log files to the following directory:

- **UNIX.** `WL_HOME/user_projects/domains/domain_name`
- **Windows.** `WL_HOME\user_projects\domains\domain_name`

It is recommended that you specify the log output paths to the following directory:

- **UNIX.** `/opt/Oracle/eBilling/logs`
- **Windows.** `D:\Oracle\eBilling\logs`

Oracle Self-Service E-Billing applications generate log files for the different functionality in each application. To change log output paths, update the XML log configuration files associated with each application.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15.

To change the output path of log files

- Edit the output path name in the File parameter for each log4j configuration files, located in the following directory:
  - **UNIX.** *EDX_HOME*`/config`

- **Windows.** *EDX_HOME*\config

The log4j configuration files are listed in the following table.

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_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 in the following example:

```
<appender name="ebilling-log" class="org.apache.log4j.RollingFileAppender">
<param name="File" value="/opt/Oracle/eBilling/logs/eBilling.log"/>
...
</ appender>
```

Adding Foreign Language Fonts to Your Application Server

If you plan to localize Oracle Self-Service E-Billing, then you must copy the foreign language fonts in the TrueType Font (TTF) files to your application server directories.

This task is a step in "Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15.

To copy foreign language fonts to your application server

- The TTF files are located in the following directory:
  - **UNIX.** *EDX_HOME*/config/fonts
  - **Windows.** *EDX_HOME*\config\fonts

  Copy all of the TTF files to the following folders in Oracle WebLogic:
  - **UNIX:**
    - WL_HOME\jdk150_11\jre\lib\fonts\
    - WL_HOME\jrockit_150_11\jre\lib\fonts\
  - **Windows:**
    - WL_HOME/jdk150_11/jre/lib/fonts/
    - WL_HOME/jrockit_150_11/jre/lib/fonts/

Installing Digital Accessible Information System (DAISY)-Related Software

If you want to offer end users the option to receive billing statements in Digital Accessible Information System (DAISY) audio format, then follow these steps to install the required DAISY-related software and configure your implementation. This feature is available to users in the Consumer Edition of Oracle Self-Service E-Billing only.

To install DAISY-related software

1. Follow the instructions contained in the following document to install and set up the Odt2daisy toolkit on the same computer where you installed Oracle Self-Service E-Billing. Do not use spaces in the save path name.

   http://odt2daisy.sourceforge.net/doc/Dev_Documentation.pdf

2. Click the following link to download the DAISY Pipeline Core binaries to the same computer where you installed Oracle Self-Service E-Billing:

   http://sourceforge.net/projects/daisymfc/files/pipeline/pipeline-20111215/pipeline-20111215.zip/download

3 Install and configure the DAISY Pipeline LAME MP3 encoder on the same computer where you installed Oracle Self-Service E-Billing. Download the software:

- **Windows.** Go to the following site and click the Softmedia Mirror link appropriate for your implementation:

  http://www.softpedia.com/progDownload/LAME-MP3-Encoder-Download-6952.html

  Create a subdirectory called ext under the pipeline-20111215 folder. Copy the LAME.exe file, found in the downloaded LAME.zip file, to this folder.

- **Linux and Oracle Solaris.** Go to the following Web site:

  http://lame.sourceforge.net/download.php

  By default. LAME installs in the path /usr/local/bin/lame directory. You can optionally specify a different installation directory.

4 Edit the pipeline.user.properties file, found in the pipeline-20111215 directory. Add the absolute path of the LAME.exe file in the Path to LAME section, for example:

- **Windows:**

```
<!-- Path to LAME -->

<entry key="pipeline.lame.path">d:\devTools\pipeline-20111215\ext\lame.exe</
entry>
```

a **Linux and Oracle Solaris**:

```
<!-- Path to LAME -->

<entry key="pipeline.lame.path">/usr/local/bin/lame</entry>
```

5 (Linux only) Install and configure the Software Speech synthesizer (Text-To-Speech) plug-in, eSpeak:

a From the main Linux menu, select System, Application, and then Add/Remove Software.

b Search for espeak. From the resulting list, select version espeak-1.40.02-4.el6 (x86_64).

By default, eSpeak installs in the /usr/bin/espeak directory.

c Edit the pipeline.user.properties file, found in the pipeline-20111215 folder. Add the absolute installed path of eSpeak to Path to SOX section:

```
<!-- Path to sox (Sound eXchange - used on Linux) -->

<entry key="pipeline.espeak.path">/usr/bin/espeak</entry>
```

6 Open the reporting configuration file, globalConfig.properties, located in the *EDX_HOME*/config/rpt directory (or the *EDX_HOME*\config\rpt directory on Windows). Specify the following parameters.

Property	Description
odt2daisy.path	The location of the odt2daisy jar file, for example: odt2daisy.path=C:\\NetBeansProjects\\odt2daisy\\dist\\odt2daisy.jar
pipeline.command	The pipeline classpath. On Windows, the library name in the classpath must be separated by a semicolon, for example: pipeline.command=java -classpath C:\\pipeline-20111215\\pipeline.jar;C:\\pipeline-20111215 org.daisy.pipeline.ui.CommandLineUI On Linux or Unix, the library in the classpath must be separated by a colon, for example: java -classpath /user/pipeline-20111215/pipeline.jar:/user/pipeline-20111215 org.daisy.pipeline.ui.CommandLineUI
pipeline.script	The location of the Pipeline software, for example: pipeline.script=D:\\pipeline-20111215\\scripts\\create_distribute\\dtb\\DTBookToDaisy3TextOnlyDTB.taskScript

7 To enable the log file for the StatementSummaryDaisyProvider job, open the log4j_cc.xml file, located in the *EDX_HOME*/config/ directory (or the *EDX_HOME*\config directory on Windows) and add the following content before the first category element:

```
<category name="report.daisy.log" additivity="false">
                <priority value="DEBUG"/>
            <!-- <appender-ref ref="FILE_SCHEDULER"/> -->
            <!-- <appender-ref ref="CONSOLE"/>  -->
                <appender-ref ref="CONSOLE"/>
        </category>
```

8 Restart the Command Center server.

9 To enable a user to select an alternative bill-ready notification method such as DAISY audio book, you must set the ISACCESSIBLE column in the EDX_BSL_UMF_USER OLTP table for the individual user to a value of 1 (the default value is 0, or disabled).

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 6.1.1" on page 15.

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.
   - **Console**. Enter the following commands:

     ```
     export LAX_DEBUG=true
     ```

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

   - **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 while 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

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 27
3. If you do not plan to use database encryption, then choose one of following methods of using Ant to create the Oracle Self-Service E-Billing database (with or without the automated Ant target):
   - To create the Oracle Self-Service E-Billing database using Ant, choose the database installation process appropriate for your implementation:
     - **Standalone Installation (Single Node, or Self-Sustaining).** See "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29.
     - **Enterprise Installation (RAC or Multinode).** Multinode or RAC systems must be preconfigured before installing Oracle Self-Service E-Billing. If you are installing on an Oracle Exadata computer, then see "Configuring the Oracle Self-Service E-Billing Database on Oracle Exadata" on page 56.

- To create the Oracle Self-Service E-Billing database using the automated Ant target, see "Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 37.

4 If you plan to use database encryption, then see "Choosing a Database Encryption Method" on page 45 and one of the following database encryption processes:

- "Process of Implementing TDE Column Encryption" on page 46
- "Process of Implementing TDE Tablespace Encryption" on page 52

5 (Optional) Loading Sample Data on page 54

6 (Optional) Enabling Oracle Auditing on page 55

7 "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121

8 "Creating a Bootstrap Administrator User for Oracle Self-Service E-Billing" on page 56

Related Topic

"Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15

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 the 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:

- **UNIX.** `$ORACLE_HOME/network/admin`
- **Windows.** `%ORACLE_HOME%\network\admin`:

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

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 = EBILL)
 (ORACLE_HOME = /opt/oracle/product/11.2.0.3)
))
```

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 name EBILL, installed on the database server localhost. Your service name might be different:

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

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...
```

```
EBILL 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, EBILL.

Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)

Oracle Self-Service E-Billing provides an ant script for creating the Oracle Self-Service E-Billing database. This topic describes how to create a single-node installation of the Oracle Self-Service E-Billing database. The single-node database is also referred to as a standalone database.

You must configure the ebilling_olap.properties and ebilling_oltp.properties files with the same EBILL database SID and tnsnames, and OLTP and OLAP user names and passwords. 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 Oracle Self-Service E-Billing Ant script.

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 Ant (single node)

1. Open the ebilling_olap.properties file, located in the following directory:
   - **UNIX.** *EDX_HOME*/db/oracle
   - **Windows.** *EDX_HOME*\db\oracle

   In the directory, *EDX_HOME* is the location where you installed Oracle Self-Service E-Billing.
2. Specify the following values in the ebilling_olap.properties file for the current installation:

Parameter	Description
ORACLE_HOME	The root installation path for any Oracle product installed through an installer that is based on Oracle Universal Installer.
ORACLE_ADMIN	Location of the Oracle admin folder.
OLAP_USER	The OLAP schema user name.
OLAP_PASSWD	The password for the OLAP schema.
OLTP_USER	The OLTP schema user name.
OLTP_PASSWD	The password for the OLTP schema.
EBILL_SID	The Oracle SID of the Oracle Self-Service E-Billing database.
EBILL_TNS_NAME	The Oracle Self-Service E-Billing database TNS (Transparent Network Substrate) name.

Parameter	Description
SYS_PASSWD	The password of the SYS user.
META_USER	Not used.
META_PASSWD	Not used.
L_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
L_DB_LOAD_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_STG_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_STG_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
DB_CONTROL_FILE_LOCN1	An instance control file folder location. The installation script automatically adds the Oracle SID to the path, as in: `/home/oracle/oradata/ORACLE_SID`.
DB_CONTROL_FILE_LOCN2	An instance control file folder location. The installation script automatically adds the Oracle SID to the path, as in: `/home/oracle/oradata/ORACLE_SID`.
DB_CONTROL_FILE_LOCN3	An instance control file folder location. The installation script automatically adds the Oracle SID to the path, as in: `/home/oracle/oradata/ORACLE_SID`.

Parameter	Description
REDO_LOG_FILE_LOCN1	An instance redo file folder location. The installation script automatically adds the Oracle SID and /redo to the path, as in: /home/oracle/oradata/ORACLE_SID/redo.
REDO_LOG_FILE_LOCN2	An instance redo file folder location. The installation script automatically adds the Oracle SID and /redo to the path, as in: /home/oracle/oradata/ORACLE_SID/redo.
SYSTEM_FILE_LOCN	The instance system file folder location. The installation script automatically adds the Oracle SID and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/data.
TRACE_FILE_LOCN	The folder location of the instance trace file, such as /home/oracle/oradata.
L_BACKUP_FILE	Not used.
L_LOG_FILE	Not used.
L_OLAP_BACKUP_FILE	Not used.
L_OLAP_LOG_FILE	Not used.
ENCRYPTION_WALLET_LOCN	(Optional) If you are implementing database encryption, specify the location of the Oracle wallet folder location. Use the same location that you set in the sqlnet.ora file.
WALLET_AUTO_OPEN	(Optional) If you are implementing database encryption, specify whether the Oracle Wallet opens automatically when the database instance is restarted, or whether you must open it manually using the Ant Encryption menu. The value can be Y or N. It is recommended to set the value to Y.
TDE_ENCRYPT_OLAPCOLUMN	(Optional) When you are using encryption, specify whether to use column-level encryption in the OLAP schema. The value can be Y or N.
TDE_ENCRYPT_OLAPTSPACE	(Optional) When you are using encryption, specify whether to use tablespace-level encryption in the OLAP schema. The value can be TRUE or FALSE.

3 Open the ebilling_oltp.properties file, located in the following directory:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

4 Specify the following values in the ebilling_oltp.properties file for the current installation:

Parameter	Description
ORACLE_HOME	The root installation path for any Oracle product installed through an installer that is based on Oracle Universal Installer.
ORACLE_BASE	Location of the Oracle base folder.
EBILL_LISTEN_PORT	The Oracle Self-Service E-Billing database listening port for creating the bootstrap user.
EBILL_HOSTNAME	The Oracle Self-Service E-Billing database host name for creating the bootstrap user.
OLTP_USER	The OLTP schema user name.
OLTP_PASSWD	The password for the OLTP schema.
OLAP_USER	The OLAP schema user name.
OLAP_PASSWD	The password for the OLAP schema.
EBILL_SID	The ORACLE SID for the Oracle Self-Service E-Billing database.
EBILL_TNS_NAME	The TNS (Transparent Network Substrate) name for the Oracle Self-Service E-Billing database.
SYS_PASSWD	The OLTP SYS user password.
L_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
L_DB_APP_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_STG_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
L_DB_STG_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
LARGE_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
LARGE_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
MEDIUM_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
MEDIUM_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
TDE_ENCRYPT_OLTPCOLUMN	(Optional) When you are using encryption, specify whether to use column-level encryption in the OLTP schema. The value can be Y or N.
TDE_ENCRYPT_OLTPTSPACE	(Optional) When you are using encryption, specify whether to use tablespace-level encryption in the OLTP schema. The value can be TRUE or FALSE.

5 Go to the directory with the Oracle Database installation files:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

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

- **UNIX.** Run the following commands, where JDK160_24 is your JDK version:

```
export ANT_HOME=/opt/apache-ant-1.6.5

export JAVA_HOME= $WEBLOGIC_HOME/JDK160_24

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

- **Windows.** Run the following commands, where JDK160_24 is your JDK version:

```
set ANT_HOME=C:\apache-ant-1.6.5

set JAVA_HOME= %WEBLOGIC_HOME%\JDK160_24

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

7 Enter Ant to run the build script.

By default, the Ant command runs the build.xml file in the current directory.

8 From the Main Menu, select Option 2, Standalone Install. This option is for a self-sustaining or single-node instance.

9 You have the option to create the database instance, tablespaces, and schemas, including the users and objects, in one step or in separate steps:

- To create the instance, tablespaces, and schemas in one step, select Option 6, Run Steps 1, 2, and 3, from the single-node Installation Main Menu.

  This step can take anywhere from 20 minutes to 2 hours to complete, depending on the speed of your platform. The script creates a new database instance for Oracle Self-Service E-Billing, defines the data dictionaries and all system database objects, and creates the tablespace and schemas. If you have set properties for TDE encryption, then the Ant script prompts you to create an Oracle Wallet password when it is ready to create the Master Encryption Key. Specify a password for the Oracle Wallet, then enter it again. Ant exits when finished. Review all log files for possible errors. You do not have to follow any more steps.

- To create the instance, tablespaces, and schemas separately, continue with Step 10.

10 Select Option 1, Create Database Instance.

This step can take anywhere from 20 minutes to 2 hours to complete, depending on the speed of your platform. The script creates a new instance for the Oracle Self-Service E-Billing database, defines the data dictionaries and all system database objects.

Ant returns to the current menu when finished. Review all log files for possible errors.

11 Select Option 2, Create eBilling Tablespaces.

This step creates a tablespace for the Oracle Self-Service E-Billing database.

12 Select Option 3, Create eBilling Schemas.

If you have set properties for TDE encryption, then the Ant script prompts you to create an Oracle Wallet password when it is ready to create the Master Encryption Key. Specify a password for the Oracle Wallet, then enter it again.

13 You have the option to create database users and objects in one step or separately:

- To create both users and objects in one step, select Option 4, Run Steps 1 and 2. Review all log files for possible errors.
- To create users and objects separately, select Option 1, Create Database Users. Select Option 2, Create Database Objects. Review all log files for possible errors.

The script creates new database users, grants the necessary permission and privileges, and creates all the database objects for the Oracle Self-Service E-Billing OLTP and OLAP schemas. Ant returns to the current menu when finished.

14 Select Option 6, Return to the Main Menu.

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 EBILL database SID and tnsnames, and OLTP and OLAP user names and passwords. 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.

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 Open the ebilling_olap.properties file, located in the following directory:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

In the directory, *EDX_HOME* is the location where you installed Oracle Self-Service E-Billing.

2 Specify the following values in the ebilling_olap.properties file for the current installation:

3 Open the ebilling_oltp.properties file, located in the following directory:

Parameter	Description
ORACLE_HOME	The root installation path for any Oracle product installed through an installer that is based on Oracle Universal Installer.
ORACLE_ADMIN	Location of the Oracle admin folder.
OLAP_USER	The OLAP schema user name.
OLAP_PASSWD	The password for the OLAP schema.
OLTP_USER	The OLTP schema user name.
OLTP_PASSWD	The password for the OLTP schema.
EBILL_SID	The Oracle SID of the Oracle Self-Service E-Billing database.
EBILL_TNS_NAME	The Oracle Self-Service E-Billing database TNS (Transparent Network Substrate) name.
SYS_PASSWD	The password of the SYS user.
META_USER	Not used.
META_PASSWD	Not used.
L_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
L_DB_APP_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_STG_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
L_DB_STG_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
DB_CONTROL_FILE_LOCN1	An instance control file folder location. The installation script automatically adds the Oracle SID to the path, as in: `/home/oracle/oradata/ORACLE_SID`.
DB_CONTROL_FILE_LOCN2	An instance control file folder location. The installation script automatically adds the Oracle SID to the path, as in: `/home/oracle/oradata/ORACLE_SID`.
DB_CONTROL_FILE_LOCN3	An instance control file folder location. The installation script automatically adds the Oracle SID to the path, as in: `/home/oracle/oradata/ORACLE_SID`.
REDO_LOG_FILE_LOCN1	An instance redo file folder location. The installation script automatically adds the Oracle SID and /redo to the path, as in: `/home/oracle/oradata/ORACLE_SID/redo`.
REDO_LOG_FILE_LOCN2	An instance redo file folder location. The installation script automatically adds the Oracle SID and /redo to the path, as in: `/home/oracle/oradata/ORACLE_SID/redo`.
SYSTEM_FILE_LOCN	The instance system file folder location. The installation script automatically adds the Oracle SID and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/data`.
TRACE_FILE_LOCN	The folder location of the instance trace file, such as `/home/oracle/oradata`.
L_BACKUP_FILE	Not used.
L_LOG_FILE	Not used.
L_OLAP_BACKUP_FILE	Not used.
L_OLAP_LOG_FILE	Not used.
ENCRYPTION_WALLET_LOCN	(Optional) If you are implementing database encryption, specify the location of the Oracle wallet folder location. Use the same location that you set in the sqlnet.ora file.

Parameter	Description
WALLET_AUTO_OPEN	(Optional) If you are implementing database encryption, specify whether the Oracle Wallet opens automatically when the database instance is restarted, or whether you must open it manually using the Ant Encryption menu. The value can be Y or N. It is recommended to set the value to Y.
TDE_ENCRYPT_OLAPCOLUMN	(Optional) When you are using encryption, specify whether to use column-level encryption in the OLAP schema. The value can be Y or N.
TDE_ENCRYPT_OLAPTSPACE	(Optional) When you are using encryption, specify whether to use tablespace-level encryption in the OLAP schema. The value can be TRUE or FALSE.

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

4 Specify the following values in the ebilling_oltp.properties file for the current installation:

Parameter	Description
ORACLE_HOME	The root installation path for any Oracle product installed through an installer that is based on Oracle Universal Installer.
ORACLE_BASE	Location of the Oracle base folder.
EBILL_LISTEN_PORT	The Oracle Self-Service E-Billing database listening port for creating the bootstrap user.
EBILL_HOSTNAME	The Oracle Self-Service E-Billing database host name for creating the bootstrap user.
OLTP_USER	The name of the OLTP schema.
OLTP_PASSWD	The password for the OLTP schema.
OLAP_USER	The name of the OLAP schema.
OLAP_PASSWD	The password for the OLAP schema.
EBILL_SID	The ORACLE SID for the Oracle Self-Service E-Billing database.
EBILL_TNS_NAME	The TNS (Transparent Network Substrate) name for the Oracle Self-Service E-Billing database.
SYS_PASSWD	The password for the user SYS.

Parameter	Description
L_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
L_DB_FS_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_FS_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_STG_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_STG_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
LARGE_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
LARGE_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Parameter	Description
MEDIUM_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_DATA_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
MEDIUM_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_INDX_TB_FILE_LOC schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
TDE_ENCRYPT_OLTPCOLUMN	(Optional) When you are using encryption, specify whether to use column-level encryption in the OLTP schema. The value can be Y or N.
TDE_ENCRYPT_OLTPTSPACE	(Optional) When you are using encryption, specify whether to use tablespace-level encryption in the OLTP schema. The value can be TRUE or FALSE.

5 Change directory to the location of the Oracle Self-Service E-Billing Database installation files in your software installation:

6 Open the ebilling_olap.properties file, located in the following directory:

- **UNIX.** *EDX_HOME*`/db/oracle`
- **Windows.** *EDX_HOME*`\db\oracle`

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

```
ant install-new
```

Or, if the EBILL instance and tablespaces are already created and you want to create schemas on them, then 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
```

These commands create the Oracle Self-Service E-Billing seed data for OLTP and OLAP.

If you want to install sample data for both OLAP and OLTP, then use the following commands:

- **Telco Sample Data.** `ant install-existing -DloadSampleDataTelco=true`
- **Utility Sample Data.** `ant install-existing -DloadSampleDataUtility=true`

8 If you have set properties for TDE encryption, then the Ant script prompts you to create an Oracle Wallet password when it is ready to create the Master Encryption Key. Specify a password for the Oracle Wallet, then enter it again.

9 Install the ETL module and the Oracle Warehouse Builder Repository.

For instructions on installing the ETL module, see "Installing the ETL Module" on page 130.

For instructions on installing and setting up the Oracle Warehouse Builder Repository, see "Process of Installing the Oracle Warehouse Builder Repository" on page 123.

Choosing a Database Encryption Method

Oracle Self-Service E-Billing supports Transparent Data Encryption (TDE), which ships with the Oracle database. Encryption is an optional feature. If you decide to use encryption, then you must choose your security strategy.

There are two types of Transparent Data Encryption:

- TDE Column Encryption
- TDE Tablespace Encryption

Both types of TDE provide secure storage and management of encryption keys in Oracle Wallet.

TDE Column Encryption enables encryption of sensitive data in select table columns. Use TDE Column Encryption if you only have small amount of data to encrypt.

TDE Tablespace Encryption is an alternative to TDE Column Encryption, and enables encryption of an entire tablespace. Encrypting an entire tablespace eliminates the need to identify which columns contain sensitive data.

NOTE: You must 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:
  - **TDE Column Encryption.** Identify which columns contain sensitive data.

    For details on implementing TDE Column Encryption, see "Process of Implementing TDE Column Encryption" on page 46.

- **TDE Tablespace Encryption.** Identify which tablespaces contain sensitive data.

  For details on implementing TDE Tablespace Encryption, see "Process of Implementing TDE Tablespace Encryption" on page 52. Consider the potential for a reduction in performance of about 9% when database-level encryption is enabled.

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

Process of Implementing TDE Column Encryption

TDE Column 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 schemas already known to contain sensitive data are listed in the CSV files:

- **tde_olap_columns.csv.** Columns in the OLAP schema that contain sensitive data; 0 columns included by default.
- **tde_oltp_columns.csv.** Columns in the OLTP schema that contain sensitive data; 95 columns included by default.

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

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

To implement TDE Column Encryption, perform the following tasks:

1. Follow the steps in "Specifying the Oracle Wallet Location in the SQLnet.ora File" on page 49.
2. (Optional) To encrypt additional columns, follow these steps:
   a. Open the tde_olap_columns.csv and tde_oltp_columns.csv files, located in the following directory, where *EDX_HOME* is the location where you installed Oracle Self-Service E-Billing:
      - **UNIX.** *EDX_HOME*/db/oracle/encrypt
      - **Windows.** *EDX_HOME*\db\oracle\encrypt

b 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

...

3 Follow the steps in "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29. Set the following encryption properties when configuring the OLAP and OLTP billing property files in those steps.

Property File	Encryption Property	Description
ebilling_olap.properties	ENCRYPTION_WALLET_LOCN	Specify the location of the Oracle Wallet folder. Use the same location that you set in the sqlnet.ora file.
ebilling_olap.properties	WALLET_AUTO_OPEN	Specify whether to automatically open the Oracle Wallet when the database instance restarts. Valid values are Y or N. It is recommended to set the value to Y.
ebilling_olap.properties	TDE_ENCRYPT_OLAPCOLUMN	Set the value to Y to set the Master Encryption Key and enable column-level encryption in the OLAP schema.
ebilling_olap.properties	TDE_ENCRYPT_OLAPTSPACE	Set the value to FALSE to suppress tablespace-level encryption in the OLAP schema.
ebilling_oltp.properties	TDE_ENCRYPT_OLTPCOLUMN	Set the value to Y to enable column-level encryption in the OLTP schema.
ebilling_oltp.properties	TDE_ENCRYPT_OLTPTSPACE	Set the value to FALSE to suppress tablespace-level encryption in the OLTP schema.

NOTE: If you do not set the encryption properties while setting the other properties in the ebilling_olap.properties and ebilling_oltp.properties files, then you will have to open these files again to set the encryption properties before performing column encryption. You will also have to use the Ant Encryption menu to set the Master Encryption Key and open the Oracle Wallet, which require you to shut down and restart the database again.

4 The Ant script prompts you to create an Oracle Wallet password when it is ready to create the Master Encryption Key. Specify a password for the Oracle Wallet, then enter it again.

The installation script sets the Master Encryption Key for TDE column encryption after creating the schema objects.

5 Go to the directory where the Oracle Self-Service E-Billing database installation files are located:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

6 Enter Ant.

By default, the Ant command runs the build.xml file in the current directory.

7 From the Main Menu, select Option 2, Standalone Install. Select Option 4, Encrypt Sensitive Data.

8 (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 log files for details. The log files are located in the following directory:

- **UNIX.** *EDX_HOME*/db/oracle/encrypt
- **Windows.** *EDX_HOME*\db\oracle\encrypt

9 If you did not set the TDE_ENCRYPT_OLAPCOLUMN and TDE_ENCRYPT_OLTPCOLUMN properties in the ebilling_olap.properties and ebilling_oltp.properties files to Y before running the Ant installation script, then do the following:

a Open the ebilling_olap.properties and ebilling_oltp.properties files, located in the following directory, and specify the values described in Step 3 on page 48 for the current installation:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

b Follow the steps in "Setting the Master Encryption Key Using the Ant Encryption Menu" on page 50.

10 If you did not set the WALLET_AUTO_OPEN property in the ebilling_olap.properties file to Y (to automatically open the Oracle Wallet when starting a database instance, then follow the steps in "Opening the Oracle Wallet Using the Ant Encryption Menu" on page 51.

11 Select Option 4, Encrypt OLAP sensitive data. Review the encrypt_olap.log file, located in the following directory, for detailed information:

- **UNIX.** *EDX_HOME*/db/oracle/encrypt
- **Windows.** *EDX_HOME*\db\oracle\encrypt

12 Select Option 5, Encrypt OLTP Sensitive Data. Review the encrypt_oltp.log file, located in the following directory, for detailed information:

- **UNIX.** *EDX_HOME*/db/oracle/encrypt
- **Windows.** *EDX_HOME*\db\oracle\encrypt

13 Select Q, Quit.

Specifying the Oracle Wallet Location in the SQLnet.ora File

To implement Transparent Data Encryption on either columns or tablespaces, you must specify an Oracle Wallet location in the sqlnet.ora configuration file before creating the database instance.

After creating the database instance, the database installation script creates an Oracle Wallet in the location specified in sqlnet.ora. An external security module adds a Master Encryption Key to the Wallet.

This task is a step in "Process of Implementing TDE Column Encryption" on page 46 and in "Process of Implementing TDE Tablespace Encryption" on page 52.

To specify the Oracle Wallet location in the SQLnet.ora files

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 and end with a right parenthesis. Verify that there are no invisible characters at the end of the directory path. These characters can cause Oracle Database not to recognize the directory.

3 Save and close the sqlnet.ora file.

Setting the Master Encryption Key Using the Ant Encryption Menu

When implementing TDE column encryption, you can optionally set the Master Encryption Key, using the Ant Encryption menu after creating the Oracle Self-Service E-Billing database, but before encrypting the columns. Setting the Master Encryption Key using the Ant Encryption menu is only necessary if you did not set the TDE_ENCRYPT_OLAPCOLUMN and TDE_ENCRYPT_OLTPCOLUMN property values to Y in the ebilling_olap.properties and ebilling_oltp.properties files before running the Oracle Self-Service E-Billing database installation using Ant.

This task is a step in "Process of Implementing TDE Column Encryption" on page 46.

To set master encryption key using the Ant Encryption menu

1. Change to the directory of the Oracle Self-Service E-Billing database installation files. In the directory, *EDX_HOME* is the location where you installed Oracle Self-Service E-Billing.
   - **UNIX.** *EDX_HOME*/db/oracle
   - **Windows.** *EDX_HOME*\db\oracle
2. Enter Ant to run the build script.

   By default, the Ant command runs the build.xml file in the current directory.
3. From the Main Menu, select Option 2, Standalone Install.
4. (Optional) Select Option 3, Encrypt Sensitive Data.
5. 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.

   Ant prompts you to create an Oracle Wallet password when it is ready to create the Master Encryption Key.
6. Specify a password for the Oracle Wallet, then enter it again.
7. Specify whether you want to open the Oracle Wallet automatically when the database instance restarts.

   It is recommended to enter Y to keep the Oracle Wallet open when the database restarts. If you enter N, then you will have to open the Wallet each time you restart the database.

Opening the Oracle Wallet Using the Ant Encryption Menu

The external security module stores the Master Encryption Key in an Oracle Wallet. The Oracle Self-Service E-Billing 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.

When the Wallet is opened, 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. The Oracle Self-Service E-Billing database is configured for manual open mode when the WALLET_AUTO_OPEN property in the eBilling_OLAP_properties file is set to N. To configure the Wallet to open automatically upon restarting the instance, change the property value to Y.

This task is a step in "Process of Implementing TDE Column Encryption" on page 46 and in "Process of Implementing TDE Tablespace Encryption" on page 52.

To open the Oracle Wallet using the Ant Encryption menu

1. Change to the directory of the Oracle Self-Service E-Billing database installation files. In the directory, *EDX_HOME* is the location where you installed Oracle Self-Service E-Billing.
   - **UNIX.** *EDX_HOME*/db/oracle

- **Windows.** *EDX_HOME*\db\oracle

2 Enter Ant.

By default, the Ant command runs the build.xml file in the current directory.

3 From the Main Menu, select Option 2, Standalone Install.

4 Select Option 3, Encrypt Sensitive Data.

5 Select Option 3, Open Oracle Wallet.

NOTE: If you shut down an encrypted database (a shutdown abort), then 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;

Process of Implementing TDE Tablespace Encryption

By default, TDE Tablespace Encryption is disabled in Oracle Self-Service E-Billing. To enable TDE Tablespace Encryption, you must specify the encryption properties in ebilling_olap.properties and ebilling_oltp.properties files when setting the configuration values for use by Ant script that creates the Oracle Self-Service E-Billing database.

When you enable TDE Tablespace Encryption, the Oracle Self-Service E-Billing database installation script encrypts the following OLTP tablespaces by default:

- EDX_DATA
- EDX_FS_DATA
- EDX_PWC_DATA
- EDX_APP_DATA

You can optionally identify any additional tablespaces you want to encrypt in two SQL files. No OLAP tablespaces are encrypted by default.

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

To set up TDE Tablespace Encryption, perform the following tasks:

1 Follow the steps in "Specifying the Oracle Wallet Location in the SQLnet.ora File" on page 49.

2 For other tablespaces you want to encrypt, in the setuptablespaces.sql file for the OLTP and OLAP databases, change the encrypted field value from FALSE to TRUE in the corresponding insert statements. The setuptablespaces.sql files are located in the following directories:

- **OLTP Database**:
  - **UNIX.** *EDX_HOME*/db/oracle/oltp/setuptablespaces.sql
  - **Windows.** *EDX_HOME*\db\oracle\oltp\setuptablespaces.sql
- **OLAP Database**:
  - **UNIX.** *EDX_HOME*/db/oracle/olap/setuptablespaces.sql
  - **Windows.** *EDX_HOME%*\db\oracle\olap\setuptablespaces.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 "ebill_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');
```

3 Follow the steps in "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29 to create the Oracle Self-Service E-Billing database. You must set the encryption properties as follows when configuring the OLAP and OLTP billing property files in those steps.

Property File	Encryption Property	Description
ebilling_olap.properties	ENCRYPTION_WALLET_LOCN	Specify the location of the Oracle Wallet folder. Use the same location that you set in the sqlnet.ora file.
ebilling_olap.properties	WALLET_AUTO_OPEN	Specify whether the Oracle wallet opens automatically when the database instance is restarted, or whether you must open it manually. Valid values are Y or N. It is recommended to set the value to Y.
ebilling_olap.properties	TDE_ENCRYPT_OLAPCOLUMN	Set the value to N to suppress column-level encryption in the OLAP schema.

Property File	Encryption Property	Description
ebilling_olap.properties	TDE_ENCRYPT_OLAPTSPACE	Set the value to TRUE to set the Master Encryption Key and encrypt tablespaces in the OLAP schema.
ebilling_oltp.properties	TDE_ENCRYPT_OLTPCOLUMN	Set the value to N to suppress column-level encryption in the OLTP schema.
ebilling_oltp.properties	TDE_ENCRYPT_OLTPTSPACE	Set the value to TRUE to encrypt tablespaces in the OLTP schema.

4 The Ant script prompts you to create an Oracle Wallet password when it is ready to create the Master Encryption Key. Specify a password for the Oracle Wallet, then enter it again.

The installation script sets the Master Encryption Key for TDE tablespace encryption and creates the encrypted tablespaces as you specified.

Loading Sample Data

You can optionally load sample data into the Oracle Self-Service E-Billing database.

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

To load sample data into your Oracle Self-Service E-Billing database

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

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

2 Enter Ant to run the build script.

By default, the Ant command runs the build.xml file in the current directory.

3 From the Main Menu, select Option 2, Standalone Install.

4 Select Option 3, Create eBilling Schemas

5 Select Option 3, Load Sample Data.

6 Choose the sample data for your line of business only:

- Option 1, Load the Telco Sample Data
- Option 2, Load the Utility Sample Data

7 Review all log files for possible errors.

8 Select Option 3, Return to Main Menu, or select Option 4, Quit.

Enabling Oracle Auditing

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

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

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

Enabling Standard Auditing

Follow these steps to enable standard database auditing.

To enable standard auditing

1. Open the init.ora initialization file for the Oracle Self-Service E-Billing EBILL database, located in the following directory:
   - **UNIX.** *$ORACLE_HOME/dbs*
   - **Windows.** *%ORACLE_HOME%*
2. Set the Oracle static initialization parameter audit_trail to db, extended.
3. Restart the database instance.
4. Using SQL*Plus, connect to the EBILL database as the OLTP schema user.
5. For each table you want to audit, run the following command. In the command, *Tablename* is the database tablename.

   ```
   audit all on Tablename by access;
   ```

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

Enabling Auditing by a System Administrator

Follow these steps to enable auditing by a system administrator.

To enable auditing by a system administrator

1. Open the init.ora initialization file for the Oracle Self-Service E-Billing EBILL database, located in the following directory:
   - **UNIX.** *$ORACLE_HOME/dbs*
   - **Windows.** *%ORACLE_HOME%*
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.

Creating a Bootstrap Administrator User for Oracle Self-Service E-Billing

You must create a default bootstrap administrator user for Oracle Self-Service E-Billing. The bootstrap administrator is the only user who can create other administrator users for the Command Center and CSR applications.

To create a bootstrap administrator user for Oracle Self-Service E-Billing

1. Go to the directory with the Oracle Database installation files:
   - **UNIX.** *EDX_HOME*/db/oracle
   - **Windows.** *EDX_HOME*\db\oracle
2. Enter Ant to run the build script.

   By default, the Ant command runs the build.xml file in the current directory.
3. From the Main Menu, select Option 3, Create Bootstrap User.
4. Enter a user name for the bootstrap administrator.
5. Enter a password for the bootstrap administrator. Enter the password again when prompted to confirm.

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 with Database Configuration Assistant (DBCA). You must also modify the Ant script before creating the schemas.

You do not have to configure Oracle Services.

If you are using Oracle RAC One Node and creating a single instance of a database on Oracle Exadata, then you do not have to follow these steps. Use the procedures for UNIX described "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29.

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 instance manually, using DBCA.

   NOTE: DBCA automatically creates a TNS name for a new instance. You do not have to manually configure the tnsnames.ora file.
3. Use SQL*Plus to log on the EBILL instance as sysdba. Run the following scripts individually as the EBILL 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. Make the following modifications to the buildinstance.xml file, found in the *EDX_HOME/db/oracle* directory:

   - Remove the following code from the CheckdirOLAP 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"/>
     ```

   - Remove the following code from the CheckdirOLTP 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 “SetupDatabaseOLAP"->replace->replacefilter, remove "${EBILL_SID}/${OLAP_USER}/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="${L_DB_EDX_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLAP_USER}/data"/>
     ```

   - In the target "SetupDatabaseOLTP "->replace->replacefilter, remove "${EBILL_SID}/${OLTP_USER}/data" in 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}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$L_DB_APP_DATA_TB_FILE_LOC"
value="${L_DB_APP_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$L_DB_EDX_INDX_TB_FILE_LOC"
value="${L_DB_EDX_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data">
token="$L_DB_APP_INDX_TB_FILE_LOC"
value="${L_DB_APP_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data">
token="$L_DB_LOAD_DATA_TB_FILE_LOC"
value="${L_DB_LOAD_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data">
token="$L_DB_LOAD_INDX_TB_FILE_LOC"
value="${L_DB_LOAD_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$L_DB_FS_DATA_TB_FILE_LOC"
value="${L_DB_FS_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$L_DB_FS_INDX_TB_FILE_LOC"
value="${L_DB_FS_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$L_DB_STG_DATA_TB_FILE_LOC"
value="${L_DB_STG_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$L_DB_STG_INDX_TB_FILE_LOC"
value="${L_DB_STG_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$LARGE_DB_EDX_DATA_TB_FILE_LOC"
value="${LARGE_DB_EDX_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$LARGE_DB_EDX_INDX_TB_FILE_LOC"
value="${LARGE_DB_EDX_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$MEDIUM_DB_EDX_DATA_TB_FILE_LOC"
value="${MEDIUM_DB_EDX_DATA_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
token="$MEDIUM_DB_EDX_INDX_TB_FILE_LOC"
value="${MEDIUM_DB_EDX_INDX_TB_FILE_LOC}/${EBILL_SID}/${OLTP_USER}/data"/>
```

5 Make the following modifications to the Ant script property files:

- **Modifications to OLAP Properties.** In the ebilling_olap.properties file, for the EBILL_SID, set the SID on your software database server, such as EBILL_SID=ebill1. Set all TB_FILE_LOC properties to this format: `DISK_GROUP/EBILL_TNS/datafile`. For example: `L_DB_EDX_DATA_TB_FILE_LOC=+DATA/ebill/olap/datafile`. Specify a local directory value for the property TRACE_FILE_LOCN, for example: `TRACE_FILE_LOCN=/export/oracle/ebill`. 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
- **Modifications to OLTP Properties.** In the ebilling_oltp.properties file, for the EBILL_SID, set the SID on your software database server, such as EBILL_SID=ebill1. Set all TB_FILE_LOC properties to this format: `DISK_GROUP/EBILL_TNS/datafile`, for example: `L_DB_EDX_DATA_TB_FILE_LOC=+DATA/ebill/oltp/datafile`.

6 Run the modified Ant script, using the install-existing option. Use either of the following procedures:

- “Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)” on page 29
- “Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target” on page 37

7 Follow the procedures in “Installing the ETL Module for Oracle Self-Service E-Billing” on page 121.

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

You must configure Oracle WebLogic to use each Oracle Self-Service E-Billing application.

This roadmap is part of "Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15.

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

1. "Preparing to Configure Oracle WebLogic" on page 62
2. Follow the process of configuring Oracle WebLogic for the Billing and Payment application appropriate for your implementation:
   - **On a single computer.** "Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 63

- **On multiple computers with Oracle WebLogic Clustering.** "Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74

3 "Process of Configuring Oracle WebLogic for the Command Center Application" on page 81

4 "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 90

5 Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications on page 97

6 "Process of Repackaging the GNU Lesser General Public License" on page 108

7 "Deploying Oracle Self-Service E-Billing Applications on Oracle WebLogic" on page 112

8 If you are configuring the Billing and payment application to run on an Oracle WebLogic cluster, follow the steps in "Starting the Oracle WebLogic Cluster" on page 115.

9 "Configuring and Starting Scheduler on Oracle WebLogic" on page 116

10 "Running the Sample Oracle Self-Service E-Billing Applications on Oracle WebLogic" on page 118

Preparing to Configure Oracle WebLogic

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

This task is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

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, then 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, then make sure your values are consistent in all servers for your installation of Oracle Self-Service E-Billing.

Process of Configuring Oracle WebLogic for the Billing and Payment Application

This topic describes the process of configuring Oracle WebLogic for the Billing and Payment application.

If you are configuring the Billing and Payment application on an Oracle WebLogic cluster, do not follow this process. Instead, follow "Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74.

This process is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

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

1. "Creating the Oracle WebLogic Domain for the Billing and Payment Application" on page 63
2. "Defining the Oracle WebLogic Environment for the Billing and Payment Application Domain" on page 64
3. "Enabling HTTPS on Your Server for the Billing and Payment Application" on page 65
4. "Configuring JDBC Resources for the Billing and Payment Application" on page 66
5. "Setting the Mail Server Properties for the Billing and Payment Application" on page 68
6. "Setting the Global Configuration Properties for the Billing and Payment Application" on page 70
7. "Setting Up Prenote Functionality" on page 73

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 63.

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

1. Go to the following directory:
   - **UNIX.** `WL_HOME/wlserver_12.1/common/bin`
   - **Windows.** `WL_HOME\wlserver_12.1\common\bin`
2. Run the following command:
   - **UNIX.** `config.sh`
   - **Windows.** `config.cmd`
3. On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain, then click Next.

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

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

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

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

8 Enter the name of the domain to create, such as ebilling_domain, then enter a location for the domain, such as port 7001, and the 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 63.

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.** `WL_HOME/user_projects/domains/ebilling_domain/bin/setDomainEnv.sh`
- **Windows.** `%WL_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:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Linux.** `CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Windows.** set `CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\config\resourcebundle;%EDX_HOME%\lib\xercesImpl-2.7.1.jar;%EDX_HOME%\lib\xalan-2.7.1.jar;%EDX_HOME%\lib\serializer-2.7.1.jar`

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

- **Oracle Solaris**. Add the Java as shown:

```
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**. Add the Java as shown:

```
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.** Add the Java as shown. (The backslashes (\) 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 Specify the memory settings as follows:

```
MEM_MAX_PERM_SIZE_64BIT=”-XX:MaxPermSize=1024m
```

6 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 63.

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 your domain name, Environment, and Servers.
3. In the Servers table, click the server where you want to deploy your application.
4. Select Configuration, General tab, and then click SSL Listen Port Enabled. Enter an SSL port number, then click Save.
5. Click Activate Changes to save 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 63.

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 the Data Sources link under JDBC in the services section.

4 Click New to create a new data source.

5 Enter the data source name and the JNDI name.

6 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 and Driver	Settings	Database Details
edxAdminDataSource	edx.databasePool	Oracle Driver (Thin)	Emulate Two-Phase Commit	OLTP/ Hostname: ListenPort:SID
edxXMADataSource	edx.xma.databasePool	Oracle Driver (Thin XA)	None	OLTP/ Hostname: ListenPort:SID
reportDataSource	edx.report.databasePool	Oracle Driver (Thin XA)	None	OLAP/ Hostname: ListenPort:SID

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

8 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

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

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

11 Click Activate Changes to save the changes.

12 Repeat from Step 4 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
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.cfg.xma.xml, with the mail host, message transport protocol, and mail account authentication properties for your organization.

This task is a step in "Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 63.

To set the notification mail server properties

1 Open the notification.cfg.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/com/edocs/common/notification
- **Windows.** *EDX_HOME*\xma\config\com\edocs\common\notification

2 Find the bean ID in the MessagePropertiesBean. 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, then set the following properties. If not, then set the mail.smtp.auth property to false, or remove the three properties:

- **mail.smtp.auth**. If the value is true, then 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, then set the user name to use when connecting to the mail server.
- **mail.password.** If mail.smtp.auth is true, then set the user password to use when connecting to the mail server.

4 If your company mail server requires SSL connection, then 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 MessagePropertiesBean:

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

    <property name="mailProperties">

      <props>

         <!-- For non authentication mail server

         <prop key="mail.host">mail.example.com</prop>

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

          -->

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

         <prop key="mail.password">eBilling603</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>

    <property name="messageFrom"><value>admin@example.com</value></property>

    </bean>

   </bean>
```

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

Setting the Global Configuration Properties for the Billing and Payment Application

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 63.

To modify the global configuration properties for the Billing and Payment application

1 Open the globalConfig.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/modules
- **Windows.** *EDX_HOME*\xma\config\modules

2 Find the bean ID called globalConfig. Modify the following 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.
ebillingPortalName	The mobile portal application name, if any.
emailImgSrc	Email notifications use HTML templates. The property value is an HTML URL referencing an image located on the Billing and Payment application server, such as https://www.XXX.com/ebilling/_assets/swan/headerBg.jpg.
emailHtmlCSS	The location of the CSS file for formatting email HTML content.
unMaskedLength	The string length, which is not masked in the notification template.
maskSymbol	The symbol used to mask the string in the notification template.
ebillingHomePageURL	The correct URL of the Billing and Payment application home page.
csrHomePageURL	The correct URL of the Customer Service Representative application home page.

The content of the globalConfig bean is:

```
<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>localhost</value></property>
```

```
    <property name="csrHostName"><value>localhost</value></property>

    <property name="rsWebServiceHostName"><value>localhost</value></property>


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

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

    <property name="rsApplicationName"><value>ebillingrs</value></property>


    <property name="ebillingHttpPort"><value>7001</value></property>

    <property name="ebillingSSLPort"><value>7002</value></property>

    <property name="csrhttpPort"><value>7005</value></property>

    <property name="csrSSLPort"><value>7006</value></property>

    <property name="rsWebServicePort"><value>7017</value></property>

    <property name="rsWebServiceSSLPort"><value>7018</value></property>


    <property name="rsSSLHostEnabled"><value>false</value></property>

    <property name="ebillingPortalName"><value>ebillingPortal.portal</value></
property>

    <property name="emailImgSrc"><value>http://localhost:7001/ebilling/_assets/
swan/headerBg.jpg</value></property>

    <property name="emailHtmlCSS"><value>notification/util/email.css</value></
property>

    <property name="unMaskedLength"><value>4</value></property>

<property name="maskSymbol"><value>x</value></property>


    <!-- for sso setting -->

    <property name="ebillingSingleSignOnEnabled"><value>false</value></
property>

    <property name="csrSingleSignOnEnabled"><value>false</value></property>

    <property name="ebillingWebServiceSingleSignOnEnabled"><value>false</
value></property>

    <!--
```

```
    <property name="singleSignOutUrl"><value>j_spring_cas_security_logout</
value></property>

    -->

    <!-- The link of Home in the page header

    <property name="ebillingHomePageUrl"><value>http://localhost:7001/portlet</
value></property>

    <property name="csrHomePageUrl"><value>http://localhost:7001/portlet</
value></property>

    -->

    <property name="autoDetectExternalResourcePeriod"><value>5</value></
property>

  </bean>

</beans>
```

Setting Up Prenote Functionality

Use the following instructions to set up prenote functionality.

This task is a step in "Process of Configuring Oracle WebLogic for the Billing and Payment Application" on page 63.

To set up prenote functionality

1. Open the payment.xma.xml file, located in the following directory:
   - **UNIX.** *EDX_HOME*\xma\config\modules\payment\file
   - **Windows.** *EDX_HOME*/xma/config/modules/payment/file
2. Find the paymentConfigurationBean bean definition, and set the enablePreNote property value as follows:

```
<property name="enablePreNote">

            <value>Yes</value>

      </property>
```

Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster

This topic describes the process of configuring the Oracle Self-Service E-Billing Billing and Payment application on an Oracle WebLogic cluster.

If you are configuring the Billing and Payment application on a single computer (nonclustered environment), do not follow these instructions. Instead, see "Process of Configuring Oracle WebLogic for the Command Center Application" on page 81.

This process is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

To configure the Billing and Payment application on an Oracle WebLogic cluster, perform the following tasks:

1. "Creating the Oracle WebLogic Domain and Managed Servers for the Billing and Payment Application on a Cluster" on page 74
2. "Defining the Oracle WebLogic Cluster Environment for the Billing and Payment Application" on page 77
3. "Enabling HTTPS on Your Server for the Billing and Payment Application" on page 65
4. "Configuring JDBC Resources for the Billing and Payment Application on an Oracle WebLogic Cluster" on page 78
5. "Setting the Mail Server Properties for the Billing and Payment Application" on page 68
6. "Setting the Global Configuration Properties for the Billing and Payment Application" on page 70
7. "Setting Up Prenote Functionality" on page 73

Creating the Oracle WebLogic Domain and Managed Servers for the Billing and Payment Application on a Cluster

To implement the Billing and Payment application on an Oracle WebLogic cluster, you must create a domain and admin server on the computer where you installed the Billing and Payment application. You must also create one or more managed servers on this computer and on the other computers in the cluster.

Table 4 shows an example of the entities created for the Billing and Payment application in a clustered environment.

Table 4. Example of a Cluster Environment for the Billing and Payment Application

Server	Associated Computer	Server Name	IP Address	Port
Admin Server	Computer 1	AdminServer	10.240.12.163	7001
Proxy Server	Computer 1	EBillingProxy	10.240.12.163	9007
Managed Server 1	Computer 1	EBillingMS1	10.240.12.163	9001
Managed Server 2	Computer 2	EBillingMS2	10.240.8.244	9002
Managed Server 3	Computer 3	EBillingMS3	10.240.12.157	9003

This task is a step in "Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74.

To create a domain and servers for the Billing and Payment application on a cluster

1. On the computer where you installed the Billing and Payment application, create a domain for the Billing and Payment EAR file:
   a. Go to the following directory:
      - **UNIX.** `WL_HOME/wlserver_12.1/common/bin`
      - **Windows.** `WL_HOME\wlserver_12.1\common\bin`
   b. Run the following command:
      - **UNIX.** `config.sh`
      - **Windows.** `config.cmd`
   c. On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain, then click Next.
   d. Leave the option to automatically configure the domain selected, then click Next.
   e. Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.
   f. Select the SUNJDK to use for this domain, then click Next.
   g. Choose Yes to configure the Oracle WebLogic domain, then click Next.
   h. Enter the name of the domain to create, such as EBilling_Domain, then enter a location for the domain, such as port 7001, and the default Admin Server. Click Create.
2. Start the domain, using the following command:

   ```
   WLS_HOME/common/bin/startWebLogic.sh
   ```

3. Create a managed server:

a. Log in to the Oracle WebLogic administration console, for example:

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

   where:

   - *localhost* is the host name of the server where you installed the Billing and Payment application. The host name can be the Domain Name System (DNS) name or the IP address, such as 10.240.12.163.
   - *7001* is the port number where you installed the Billing and Payment application.

b. Click Environment, Servers, and then New.

c. Enter a name for the managed server, such as EBillingMS1.

d. Specify the server listen address (the IP address of the managed server).

e. Enter the server listen port, such as 9001.

f. Select the Standalone Server option (not the Clustering option).

g. Click Next, and click Finish.

   If successful, the following messages appear:

   All changes have been activated. However, 1 items must be restarted for the changes to take effect.

   Server created successfully.

4. Stop the domain, using the following command:

   `$WLS_HOME/common/bin/stopWebLogic.sh`

5. Configure the Node Manager for the computer:

   a. Make sure you start the Node Manager at least once. The configuration file, nodemanager.properties, is created automatically when you first start Node Manager.

   b. Edit the nodemanager.properties file, located in the `$WLS_HOME/common/nodemanager` directory. Set the following parameter values:

Parameter	Value	Purpose
AuthenticationEnabled	false	Creates a nonauthenticated connection to Node Manager.
SecureListener	false	Creates a plain connection to Node Manager, not a secure connection.

   c. Start the Node Manager, using the following command:

      `$WEBLOGIC_HOME/server/bin/startNodeManager.sh`

6. Edit the required scripts:

   a. Add the following code to the commEnv.sh file, located in the `$WLS_HOME/common/bin` directory to set EDX_HOME, the directory where you installed Oracle Self-Service E-Billing:

```
export EDX_HOME=/home/oracle/eBilling

export CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/
resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-
2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar
```

b. Also in the commEnv.sh file, change the MaxPermSize value to 1024 for each platform.

c. In the startManagedWebLogic.sh file, located in the `$WLS_HOME/common/bin` directory, add the Dlog4j.configuration Java option to the end of the definition in the JAVA option section:

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

7. Repeat Step 3 through Step 6 as necessary to create and configure additional servers on this computer or other computers in the cluster.

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

You must configure the Oracle WebLogic cluster environment.

This task is a step in "Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74.

To configure the Oracle WebLogic cluster environment

1. Log in to the Oracle WebLogic administrative console on the computer where the admin server is located, using the following URL format:

   `http://`*`IP_Address`*`:7001/console`

   where:

   - *IP_Address* is the name of the computer, such as 10.240.12.163.
   - *7001* is the port number where you installed the Billing and Payment application.

2. Create the Oracle WebLogic cluster:

   a. Click Environment, Clusters, then New.

   b. Enter a name for the cluster, such as EBillingCluster.

   c. Select the messaging mode. For best performance, it is recommended to choose Multicast.

   d. Leave the default values for the other configuration parameters.

   e. Click OK.

3 Click the name of the cluster you just created. For each computer in the cluster, follow these steps:

   a Click Environment, Machines, New, and Next.

   b Select Plain as the protocol to use with the node manager on this computer.

   c Enter the listen address (IP address) of the computer and the listen port number, such as 5556.

   d Click Finish.

   e Click the computer that contains the node manager. Click the Monitoring tab, and then click Clusters.

      The Monitoring option continuously verifies that Node Manager is accessible.

   f Click the name of the cluster you created. Click the Servers tab.

4 For each managed server in the cluster, follow these steps:

   a Click Environment, Server, and then New.

   b Enter a name for the managed server.

   c Specify the server listen address (where the server will listen for incoming connections) and the port number.

   d Choose the Stand-Alone cluster.

   e Check the Listen Port Enabled option.

   f Click Next, and then Finish.

5 Create the Proxy Server, which functions as a load balancer:

   a Click Environment, Servers, and then New.

   b Specify a name for the proxy server, such as EBillingProxy.

   c Specify the listen address as the IP address of the computer where the admin server is located, such as 10.240.12.163. Specify the listen port, such as 9007.

   d Choose the Standalone cluster.

6 Add each server to the cluster. For each server, perform the following steps:

   a Click Clusters, click the name of the cluster, and then click the Servers tab.

   b Select the servers to add to the cluster and click Add.

   c Click Finish.

Configuring JDBC Resources for the Billing and Payment Application on an Oracle WebLogic Cluster

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 the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74.

Configuring the Data Sources for the Billing and Payment Application on an Oracle WebLogic Cluster

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://localhost:7001/console/`

   where:

   - *localhost* is the host name of the server where you installed the Billing and Payment application. The host name can be the Domain Name System (DNS) name or the IP address, such as 10.240.12.157.
   - *7001* is the port number where you installed the Billing and Payment application.

2. Log on to the Oracle WebLogic Administration Console, using the user name and password defined when the domain was created.
3. Click the Data Sources link under JDBC in the services section.
4. Click New to create a new data source.
5. Enter the data source name and the JNDI name.

6 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 and Driver	Settings	Database Details
edxAdminDataSource	edx.databasePool	Oracle Driver (Thin)	Emulate Two-Phase Commit	OLTP/ Hostname: ListenPort:SID
edxXMADataSource	edx.xma.databasePool	Oracle Driver (Thin XA)	None	OLTP/ Hostname: ListenPort:SID
reportDataSource	edx.report.databasePool	Oracle Driver (Thin XA)	None	OLAP/ Hostname: ListenPort:SID

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

8 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

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

10 Click Next. Click Finish.

11 Open each datasource and click Target. For each datasource, follow these steps:

- a Click Summary of JDBC DataSource
- b Select the datasource.
- c Select the Target tab.
- d Click the cluster name, and click Save.

12 Click Activate Changes to save the changes.

13 Repeat from Step 4 to create each remaining data source.

Configuring the Connection Pools for the Billing and Payment Application on an Oracle WebLogic Cluster

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
Shrink Frequency	15
Login Delay	1

Process of Configuring Oracle WebLogic for the Command Center Application

This topic describes the process of configuring Oracle WebLogic for the Command Center application.

This process is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

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 82
2. "Defining the Oracle WebLogic Environment for the Command Center Domain" on page 82
3. "Enabling HTTPS for the Command Center Server" on page 84
4. "Configuring JDBC Resources for the Command Center" on page 84
5. "Setting the Service Transaction Time (Linux Only)" on page 87
6. "Configuring JMS Resources for the Command Center on Oracle WebLogic" on page 87

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 81.

To create an Oracle WebLogic domain for the Command Center

1. Go to the following directory:
   - **UNIX.** `WL_HOME/wlserver_12.1/common/bin`
   - **Windows.** `WL_HOME\wlserver_12.1/common\bin`
2. Run the following command:
   - **UNIX.** `config.sh`
   - **Windows.** `config.cmd`
3. On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain, then click Next.
4. Leave the option to automatically configure the domain selected, then click Next.
5. Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.
6. Select the SUNJDK to use for this domain, then click Next.
7. Choose Yes to configure the Oracle WebLogic domain, then click Next.
8. Enter the name of the domain to create, such as cc_domain, and 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 "Process of Configuring Oracle WebLogic for the Command Center Application" on page 81.

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.** `WL_HOME/user_projects/domains/cc_domain/bin/setDomainEnv.sh`

- **Windows.** `WL_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:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Linux.** `export CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Windows.** `set CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\config\resourcebundle;%EDX_HOME%\lib\xercesImpl-2.7.1.jar;%EDX_HOME%\lib\xalan-2.7.1.jar;%EDX_HOME%\lib\serializer-2.7.1.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**. Add the Java as shown:

```
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"

export JAVA_VM
```

- **Linux**. Add the Java as shown:

```
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**. Add the Java as shown:

```
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 "Process of Configuring Oracle WebLogic for the Command Center Application" on page 81.

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/`

where:

- *localhost* is the name of the server where you installed the Command Center application.
- *7003* is the port number where you installed the Command Center application.

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

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

4 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 81.

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 the Data Sources link under JDBC in the services section.
4. Click New to create a new data source.
5. Enter the data source name and the JNDI name.
6. 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 and 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

Data Source Name	JNDI Name	Database and Driver	Settings	DB Details
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

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

8 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

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

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

11 Click Activate Changes to save 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 Service Transaction Time (Linux Only)

Linux users must set the service transaction time.

This task is a step in "Process of Configuring Oracle WebLogic for the Command Center Application" on page 81.

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 module and connection factory
- JMS queues

This task is a step in "Process of Configuring Oracle WebLogic for the Command Center Application" on page 81.

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 `WL_HOME/user_projects/domains/commandcenter_domain/config/FileStore`)
EventFileStore	Select Your Server (AdminServer)	Directory Location to Save Store

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)
EventJMSServer	EventFileStore	Select Your Server (AdminServer)

Creating the JMS Module and Connection Factory for the Command Center

You must create a JMS module and add a JMS connection factory to the module for the Command Center.

To create the JMS module and connection factory

1. Click Services, Messaging, and the JMS Modules link on the Administration Console page for the Command Center domain.
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.
4. Click the JMS Modules link on the Administration Console page for the Command Center domain.
5. Click the link for the module you created in the previous procedure.
6. Click New, select Connection Factory, and then click Next.
7. Add one Connection Factory, using the following parameters. Click Finish.

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

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

Configuring a JMS Queue for the Command Center

You must create two JMS queues for the Command Center application.

To create the JMS queues

1. Click the JMS Modules link on the Administration Console page for the Command Center domain.
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
EventQueue	edx.queue.outbound	EventJMSServer

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 to the subdeployment, then you can rename it. Select the new subdeployment and the corresponding target JMS Server.

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

Process of Configuring Oracle WebLogic for the Customer Service Representative Application

This topic describes the process of configuring Oracle WebLogic for the Customer Service Representative application.

This process is a step in "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

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 90

2 "Defining the Oracle WebLogic Environment for the Customer Service Representative Domain" on page 91

3 "Accessing the Oracle WebLogic Console" on page 93

4 "Enabling HTTPS on Your Server for the Customer Service Representative Application" on page 93

5 "Configuring JDBC Resources for the Customer Service Representative Application" on page 94

6 "Modifying the csr.xma.xml File for Customer Service Representative Properties" on page 96

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 "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 90.

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

1. Go to the following directory:
   - **UNIX.** `WL_HOME/wlserver_12.1/common/bin`
   - **Windows.** `WL_HOME\wlserver_12.1\common\bin`
2. Run the following command:
   - **UNIX.** `config.sh`
   - **Windows.** `config.cmd`
3. On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain. Click Next.
4. Leave the option to automatically configure the domain selected, then click Next.
5. Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.
6. Select the SUNJDK to use for this domain, then click Next.
7. Choose Yes to configure the Oracle WebLogic domain, then click Next.
8. Enter the name of the domain to create, such as CSR_domain, and enter a location for the domain, such as port 7006, and enter the 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 "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 90.

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

1. Go to the following directory:
   - **UNIX.** `WL_HOME/user_projects/domains/csr_domain/bin`
   - **Windows.** `WL_HOME\user_projects\domains\csr_domain\bin`
2. Open the following file in a text editor:
   - **UNIX.** `setDomainEnv.sh`
   - **Windows.** `setDomainEnv.cmd`
3. 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`

4 Add the following entries to the file:

- **Oracle Solaris.** `CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar:$EDX_HOME/lib/xercesImpl-2.7.1.jar`
- **Linux.** `CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar:$EDX_HOME/lib/xercesImpl-2.7.1.jar`
- **Windows.** set `CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\config\resourcebundle;%EDX_HOME%\lib\xalan-2.7.1.jar;%EDX_HOME%\lib\serializer-2.7.1.jar;%EDX_HOME%\lib\xercesImpl-2.7.1.jar`

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

- **Oracle Solaris.** Add the Java as shown:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} ${JAVA_PROFILE} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_csr.xml -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.TransformerFactoryImpl -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl -
Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBuilderFactoryImpl"

export JAVA_VM
```

- **Linux.** Add the Java as shown:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_csr.xml -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.TransformerFactoryImpl -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl -
Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBuilderFactoryImpl"

export JAVA_VM
```

- **Windows.** Add the Java as shown. (The backslashes (\) 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.TransformerFactoryImpl
```

```
-
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl
```

6 Save the changes, and close the file.

Accessing the Oracle WebLogic Console

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

This task is a step in "Process of Configuring Oracle WebLogic for the Customer Service Representative Application" on page 90.

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 90.

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/`

  - *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.

2 Click your domain name, Environment, and Servers.

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

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

Configuring JDBC Resources for the Customer Service Representative 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 Customer Service Representative Application" on page 90.

Configuring the Data Sources for the Customer Service Representative Application

You must create the following data sources for the Customer Service Representative application:

- edxAdminDataSource
- edxXMADataSource
- reportDataSource

To create the data source for the Customer Service Representative 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 Customer Service Representative domain.
- *Server_Port* is the port number of the Customer Service Representative 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 the Data Sources link under JDBC in the services section.

4 Click New to create a new data source.

5 Enter the data source name and the JNDI name.

6 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. For the edxAdminDataSource, select Supports Global Transactions, select the corresponding transaction option, and click Next.

Data Source Name	JNDI Name	Database and Driver	Settings	Database Details
edxAdminDataSource	edx.databasePool	Oracle Driver (Thin)	Emulate Two-Phase Commit	OLTP/ Hostname: ListenPort:SID
edxXMADataSource	edx.xma.databasePool	Oracle Driver (Thin XA)	None	OLTP/ Hostname: ListenPort:SID
reportDataSource	edx.report.databasePool	Oracle Driver (Thin XA)	None	OLAP/ Hostname: ListenPort:SID

7 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	Database Server Listening Port
Database User Name	Your Oracle Database User Name
Password	Your Oracle Database Password

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

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

10 Click Activate Changes to save the changes.

11 Repeat from Step 4 to create each remaining data source.

Configuring the Connection Pools for the Customer Service Representative Application

You must configure a connection pool for each data source.

To configure the connection pools for the Customer Service Representative 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
Shrink Frequency	15
Login Delay	1

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 90.

To modify the csr.xma.xml file

1. Open the csr.xma.xml file, located in the following directory:
   - **UNIX.** *EDX_HOME*/xma/config/modules/application/csr
   - **Windows.** *EDX_HOME*\xma\config\modules\application\csr
2. Modify the custAppURL property to point to the correct HTTPS server name or IP address where the Oracle Self-Service E-Billing Billing and Payment application is deployed.

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

   https://localhost:7002/ebilling/j_acegi_security_check?

3 Modify the custLogoutAppURL property for the HTTPS URL directed to when logging out of the Billing and Payment application.

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

Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications

Oracle Self-Service E-Billing RESTful Web Services run on a separate service domain that accesses the same database as the other Oracle Self-Service E-Billing applications. You must set up and configure a RESTful service domain for use with Oracle Self-Service E-Billing. You can optionally configure the RESTful services client, which is provided as a test harness application for development use only.

To configure Oracle WebLogic to use RESTful Web services with Oracle Self-Service E-Billing, perform the following tasks:

1 Configure the Web Services Server application:

   a “Configuring the Oracle Self-Service E-Billing Web Services Server Domain” on page 97

   b “Defining the Oracle WebLogic Environment for the Web Services Server Domain” on page 98

   c “Enabling HTTPS on Your Server for the Web Services Server Application” on page 100

   d “Configuring JDBC Resources for the Web Services Server Application” on page 100

2 (Optional) Configure the Web Services Client application (for development use only):

   a “Configuring the Oracle Self-Service E-Billing Web Services Client Domain (Development Use Only)” on page 103

   b “Defining the Oracle WebLogic Environment for the Web Services Client Domain” on page 103

   c “Enabling HTTPS on Your Client Server for Web Services” on page 105

   d “Setting the Global Configuration Properties for the Web Services Client Application” on page 106

Configuring the Oracle Self-Service E-Billing Web Services Server Domain

If you intend to use Web services with Oracle Self-Service E-Billing, then you must create an Oracle WebLogic domain for the RESTful Web Services Server application EAR file.

This task is a step in “Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications” on page 97.

To configure a server domain for the Oracle Self-Service E-Billing Web Services Server application

1. Go to the following directory:
   - **UNIX.** `WL_HOME/common/bin`
   - **Windows.** `WL_HOME\common\bin`
2. Run the following command:
   - **UNIX.** `config.sh`
   - **Windows.** `config.cmd`
3. On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain, then click Next.
4. Leave the option to automatically configure the domain selected, then click Next.
5. Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.
6. Select the SUNJDK to use for this domain, then click Next.
7. Choose Yes to configure the Oracle WebLogic domain, then click Next.
8. Enter the name of the domain to create, such as ebillingrs, then enter a location for the domain, such as port 7017, and the default Admin Server. Click Create.

Defining the Oracle WebLogic Environment for the Web Services Server Domain

You must set environment variables and other options in the Oracle WebLogic environment to correctly set up the server domain for use with Oracle Web services.

This task is a step in "Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications" on page 97.

To set environment variables for the Web Services Server domain

1. Open the setDomainEnv file in a text editor.

   This file is located in the domain's home directory, for example:
   - **UNIX.** `WL_HOME/user_projects/domains/ebillingrs/bin/setDomainEnv.sh`
   - **Windows.** `WL_HOME\user_projects\domains\ebillingrs\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:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Linux.** `CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Windows.** `set CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\config\resourcebundle;%EDX_HOME%\lib\xercesImpl-2.7.1.jar;%EDX_HOME%\lib\xalan-2.7.1.jar;%EDX_HOME%\lib\serializer-2.7.1.jar`

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

- **Oracle Solaris**. Add the Java as shown:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} ${JAVA_PROFILE} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_rs.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

- **Linux**. Add the Java as shown:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_rs.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

- **Windows.** Add the Java as shown. (The backslashes (\) 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_rs.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 Specify the memory settings as follows:

```
MEM_MAX_PERM_SIZE_64BIT=”-XX:MaxPermSize=1024m
```

6 Save the changes, and close the file.

Enabling HTTPS on Your Server for the Web Services Server Application

Follow these steps to enable HTTPS on your server for the Web Services Server application, which is required for compliance with the Payment Card Industry Data Security Standard.

The following examples show Web Services Server URLs under an HTTPS configuration:

- URL for logging in or getting the token in your application:

  https://host1:ssl_port1/ebilling-rs/rsLogin.action

- URL for accessing RESTful Web services:

  https://host1:ssl_port1/ebilling-rs/rs/{E-Billing_RS_Resource}

This task is a step in "Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications" on page 97.

To enable HTTPS on your server for Web Services Server application

1. Log in to the Web Services Server domain console, for example:

   `http://localhost:7017/console/`

   where:

   - *localhost* is the name of the server where you installed the Web Services Server application EAR.
   - *7017* is the port number where you installed the Web Services Server application.

2. Click your domain name, Environment, and Servers.
3. In the Servers table, click the server where you want to deploy your application.
4. Select Configuration, General tab, and then click SSL Listen Port Enabled. Enter an SSL port number, then click Save.
5. Click Activate Changes to save the changes.

Configuring JDBC Resources for the Web Services Server Application

You must configure the following JDBC resources for the Web Services Server application 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 RESTful Web Services Server and Client Applications" on page 97.

Configuring the Data Sources for the Web Services Server Application

You must create the following data sources for the Web Services Server application:

- edxAdminDataSource
- edxXMADataSource
- reportDataSource

To create the data sources for the Web Services Server application

1. Start the newly created Web Services Server 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 Web Services Server domain.
   - *Server_Port* is the port number of the Web Services Server 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 the Data Sources link under JDBC in the services section.
4. Click New to create a new data source.
5. Enter the data source name and the JNDI name.
6. 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 and Driver	Settings	Database Details
edxAdminDataSource	edx.databasePool	Oracle Driver (Thin)	Emulate Two-Phase Commit	OLTP/ Hostname: ListenPort:SID
edxXMADataSource	edx.xma.databasePool	Oracle Driver (Thin XA)	None	OLTP/ Hostname: ListenPort:SID
reportDataSource	edx.report.databasePool	Oracle Driver (Thin XA)	None	OLAP/ Hostname: ListenPort:SID

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

8 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

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

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

11 Click Activate Changes to save the changes.

12 Repeat from Step 4 to Step 11 to create each remaining data source.

Configuring the Connection Pools for the Web Services Server Application

You must configure a connection pool for each data source.

To configure the connection pools for the Web Services Server 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

Configuring the Oracle Self-Service E-Billing Web Services Client Domain (Development Use Only)

The RESTful client application is provided to help discover and test the preconfigured Oracle Self-Service E-Billing Web services. The RESTful client application is for development use only. Do not use the client Web Services Client application on the production server.

This task is a step in "Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications" on page 97.

To configure a client domain for Oracle Self-Service E-Billing Web services

1. Go to the following directory:
   - **UNIX.** `WL_HOME/common/bin`
   - **Windows.** `WL_HOME\common\bin`
2. Run the following command:
   - **UNIX.** `config.sh`
   - **Windows.** `config.cmd`
3. On the Oracle WebLogic Configuration Wizard, select Create a new WebLogic domain, then click Next.
4. Leave the option to automatically configure the domain selected, then click Next.
5. Enter the user name and password of the user to administer the Oracle WebLogic domain, then click Next.
6. Select the SUNJDK to use for this domain, then click Next.
7. Choose Yes to configure the Oracle WebLogic domain, then click Next.
8. Enter the name of the domain to create, such as ebillingrs, then enter a location for the domain, such as port 7019, and the default Admin Server. Click Create.

Defining the Oracle WebLogic Environment for the Web Services Client Domain

You must set environment variables and other options in the Oracle WebLogic environment to correctly set up the server domain for use with Oracle Web services.

This task is a step in "Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications" on page 97.

To set environment variables for the Web Services Client domain

1 Open the setDomainEnv file in a text editor.

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

- **UNIX.** `WL_HOME/user_projects/domains/rsclient/bin/setDomainEnv.sh`
- **Windows.** `WL_HOME\user_projects\domains\rsclient\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:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Linux.** `CLASSPATH=$CLASSPATH:$EDX_HOME/config:$EDX_HOME/config/resourcebundle:$EDX_HOME/lib/xercesImpl-2.7.1.jar:$EDX_HOME/lib/xalan-2.7.1.jar:$EDX_HOME/lib/serializer-2.7.1.jar`
- **Windows.** set `CLASSPATH=%CLASSPATH%;%EDX_HOME%\config;%EDX_HOME%\config\resourcebundle;%EDX_HOME%\lib\xercesImpl-2.7.1.jar;%EDX_HOME%\lib\xalan-2.7.1.jar;%EDX_HOME%\lib\serializer-2.7.1.jar`

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

- **Oracle Solaris**. Add the Java as shown:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} ${JAVA_PROFILE} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_rs.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

- **Linux**. Add the Java as shown:

```
JAVA_VM="${JAVA_VM} ${JAVA_DEBUG} -Dedx.home=${EDX_HOME} -
Dlog4j.configuration=file:${EDX_HOME}/config/log4j_rs.xml -
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl -
Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBui
lderFactoryImpl -Dorg.owasp.esapi.resources=${EDX_HOME}/config -
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl"

export JAVA_VM
```

- **Windows.** Add the Java as shown. (The backslashes (\) 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_rs.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 Specify the memory settings as follows:

```
MEM_MAX_PERM_SIZE_64BIT=″-XX:MaxPermSize=1024m
```

6 Save the changes, and close the file.

Enabling HTTPS on Your Client Server for Web Services

Follow these steps to enable HTTPS on your server for the Web Services Client application, which is required for compliance with the Payment Card Industry Data Security Standard.

The following examples show Web Services Client application URLs under an HTTPS configuration:

- For logging in or getting the token in your application:

  https://host1:ssl_port1/ebilling-rs/rsLogin.action

- For accessing RESTful Web services:

  https://host1:ssl_port1/ebilling-rs/rs/{E-Billing_RS_Resource}

This task is a step in “Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications” on page 97.

To enable HTTPS on your server for the Web Services Client application

1 Log in to the Web Services Client application domain console, for example:

  http://*localhost:7019*/console/

  where:

  - *localhost* is the name of the server where you installed the Web Services Client application EAR.
  - *7019* is the port number where you installed the Web Services Client application.

2 Click your domain name, Environment, and Servers.

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

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

5 Click Activate Changes to save the changes.

Setting the Global Configuration Properties for the Web Services Client Application

You must modify various properties in the global configuration file, globalConfig.xma.xml, for the Web Services Client application.

This task is a step in "Process of Configuring Oracle WebLogic for RESTful Web Services Server and Client Applications" on page 97.

To modify the global configuration properties for the Web Services Client application

1 Open the globalConfig.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/modules
- **Windows.** *EDX_HOME*\xma\config\modules

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

Property	Value
rsWebServiceHostName	The server name or IP address where the Web Services Server application is deployed.
rsApplicationName	The Web Services Server application name.
rsWebServicePort	The application port of the Web Services Server application. The Web Services Client uses this port to communicate with the Web Services Server application.
rsWebServiceSSLPort	The application SSL port of the Web Services Server application. The Web Services Client uses this SSL port to communicate with the Web Services Server application.
rsSSLHostEnabled	Either true or false, indicating whether the Web Services Server application port is enabled.

The content of the globalConfig bean is:

```
<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>localhost</value></property>
    <property name="csrHostName"><value>localhost</value></property>
    <property name="rsWebServiceHostName"><value>localhost</value></property>

    <property name="ebillingApplicationName"><value>ebilling</value></property>
    <property name="csrApplicationName"><value>ebillingcsr</value></property>
    <property name="rsApplicationName"><value>ebillingrs</value></property>

    <property name="ebillingHttpPort"><value>7001</value></property>
    <property name="ebillingSSLPort"><value>7002</value></property>
    <property name="csrhttpPort"><value>7005</value></property>
    <property name="csrSSLPort"><value>7006</value></property>
    <property name="rsWebServicePort"><value>7017</value></property>
    <property name="rsWebServiceSSLPort"><value>7018</value></property>

    <property name="rsSSLHostEnabled"><value>false</value></property>
    <property name="ebillingPortalName"><value>ebillingPortal.portal</value></
property>
    <property name="emailImgSrc"><value>http://localhost:7001/ebilling/_assets/
swan/headerBg.jpg</value></property>
    <property name="emailHtmlCSS"><value>notification/util/email.css</value></
property>
    <property name="unMaskedLength"><value>4</value></property>
<property name="maskSymbol"><value>x</value></property>

    <!-- for sso setting -->
    <property name="ebillingSingleSignOnEnabled"><value>false</value></
property>
    <property name="csrSingleSignOnEnabled"><value>false</value></property>
    <property name="ebillingWebServiceSingleSignOnEnabled"><value>false</
value></property>
```

```
    <!--

    <property name="singleSignOutUrl"><value>j_spring_cas_security_logout</
value></property>

    -->

    <!-- The link of Home in the page header

    <property name="ebillingHomePageUrl"><value>http://localhost:7001/portlet</
value></property>

    <property name="csrHomePageUrl"><value>http://localhost:7001/portlet</
value></property>

    -->

    <property name="autoDetectExternalResourcePeriod"><value>5</value></
property>

  </bean>

</beans>
```

Process of Repackaging the GNU Lesser General Public License

You must repackage the GNU Lesser General Public License (LGPL) on each server where you have an Oracle Self-Service E-Billing application installed (all platforms), including the servers where you have installed the Web Services Server and Client applications.

This process is a step in the following roadmaps:

- “Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing” on page 61
- “Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1” on page 153

To complete repackaging the GNU LGPL, perform the following tasks on each server:

1. “Setting Up Maven” on page 108
2. “Setting Up a Proxy Maven Configuration” on page 109
3. “Repackaging LGPL” on page 110

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 108.

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, then set up a proxy in the Maven configuration.

This task is a step in "Process of Repackaging the GNU Lesser General Public License" on page 108

To set up a proxy Maven configuration

1. Edit the `settings.xml` file, found in the following file:
2. Go to the following directory:
   - **UNIX.** `%M2_HOME%/conf`
   - **Windows.** `%M2_HOME%\conf`
3. 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

You must download and repackage LGPL libraries for Oracle Self-Service E-Billing.

This task is a step in "Process of Repackaging the GNU Lesser General Public License" on page 108.

To repackage LGPL

1 Download the proxool-0.9.1.jar file from the following location:

http://maven.cloudhopper.com/repos/third-party/proxool/proxool/0.9.1/

Save the file to the following directory:

- **UNIX.** *EDX_HOME*/repackage
- **Windows.** *EDX_HOME*\repackage

2 From the same directory, run the following Maven installation command:

```
mvn install:install-file -DgroupId=proxool -DartifactId=proxool -Dversion=0.9.1
-Dpackaging=jar -Dfile=proxool.0.9.1.jar
```

3 Run the following Maven installation command:

```
mvn install
```

This command places all EAR files in the packaged LGPL libraries in the following directory:

- **UNIX.** EDX_HOME/J2EEApps

- **Windows.** `EDX_HOME\J2EEApps`

If you receive an error similar to the following, or any error indicating JAR files are missing, then the command was not able to access and download LGPL components from the Internet. Follow the instructions in "Downloading LGPL Jar Files Manually" on page 111 to manually download and install the components.

```
[INFO] ------------------------------------------------------------------------
---
[INFO] Building eBilling application
[INFO]    task-segment: [install]
[INFO] ------------------------------------------------------------------------
---
[INFO] ------------------------------------------------------------------------
[ERROR] BUILD ERROR
[INFO] ------------------------------------------------------------------------
[INFO] Failed to resolve artifact.
```

Downloading LGPL Jar Files Manually

If you received an error while repackaging LGPL libraries that indicated JAR files were missing, then follow these steps to manually download and repackage the required components.

To manually download and repackage the required LGPL JAR files

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=1.0RC2 -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 Download the proxool-0.9.1.jar file from the following location and save it to the *EDX_HOME*/repackage directory (or the *EDX_HOME*\repackage directory in Windows):

http://maven.cloudhopper.com/repos/third-party/proxool/proxool/0.9.1/

8 Go to the following directory:

- **UNIX.** *EDX_HOME*/repackage
- **Windows.** *EDX_HOME*\repackage

9 Run the following Maven installation command:

```
mvn install:install-file -DgroupId=proxool -DartifactId=proxool -Dversion=0.9.1
-Dpackaging=jar -Dfile=proxool.0.9.1.jar
```

10 Run the following Maven installation command:

```
mvn install
```

This command places all EAR files in the packaged LGPL libraries in the following directory:

- **UNIX.** EDX_HOME/J2EEApps
- **Windows.** EDX_HOME\J2EEApps

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

For each of the following Oracle Self-Service E-Billing applications you have configured for your implementation, you must deploy the corresponding EAR files, using your application server:

- Billing and Payment application
- Command Center application
- Customer Service Representative application
- Web Services Server and Client applications

This task is a step in “Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing” on page 61.

To deploy an Oracle Self-Service E-Billing application on Oracle WebLogic

1. If you are deploying the Billing and Payment application on an Oracle WebLogic cluster, edit the weblogic.xml file, located in the edx_home\J2EEApps\ebilling\weblogic\ebilling-weblogic-10.3-6.1\ebilling-web-1.0-SNAPSHOT\WEB-INF\ directory. Add the following code to enable a session on one cluster computer member to be replicated to the other computer members:

```
<session-descriptor>

<session-param>

<param-name>PersistentStoreType</param-name>

<param-value>replicated</param-value>

</session-param>

</session-descriptor>
```

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

5 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-10.3-6.1.1.ear Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/ebilling/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\ebilling\weblogic\
Command Center	File Name: command-center-weblogic-10.3-6.1.1.ear Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/commandcenter/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\commandcenter\weblogic
Customer Service Representative	File Name: csr-app-6.1.1.ear Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/csr/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\csr\weblogic
Web Services Server	File Name: ebilling-rs-weblogic-10.3-6.1.1.ear Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/rs/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\rs\weblogic
Web Services Client (for development use only)	File Name: ebilling-rsclient-weblogic-10.3-6.1.1.ear Location: ■ **UNIX.** *EDX_HOME*/J2EEApps/rsclient/weblogic ■ **Windows.** *EDX_HOME*\J2EEApps\rsclient\weblogic

Select the deployed EAR file, and start all services located at the Start, Servicing all requests. If you are using the Billing and Payment application on a cluster, then deploy the Billing and Payment application to the cluster, not the Admin server.

6 Restart the server.

7 Repeat this procedure for each Oracle Self-Service E-Billing application that your implementation requires.

8 If you are using the Billing and Payment application on a cluster, you must also deploy the Cluster Proxy application to the proxy server. Edit the web.xml file found in the eBillingProxy WAR file. In the following section, add the IP address and port number of each managed server in the cluster, for example:

```
<init-param>
      <param-name>WebLogicCluster</param-name>
      <param-value>
        10.240.12.163:9001|10.240.8.244:9002|10.240.12.157:9003
      </param-value>
    </init-param>
```

9 If you are deploying the Billing and Payment application on a cluster, then start the cluster.

For details, see "Starting the Oracle WebLogic Cluster" on page 115.

Starting the Oracle WebLogic Cluster

If you are configuring the Billing and Payment application on an Oracle WebLogic cluster, you must start the cluster after deployment.

This task is a step in "Process of Configuring the Billing and Payment Application on an Oracle WebLogic Cluster" on page 74.

To start the Oracle WebLogic cluster

1 On the computer where the Admin server is located, start the admin server:

```
$eBilling_domain/bin/startWeblogic.sh
```

2 On each computer, start the node manager:

```
$weblogic_home/server/bin/startNodeManager.sh
```

3 Start each of the managed servers that is on the same computer as the Admin server:

```
$weblogic_home/common/bin/startManagedWeblogic.sh EBillingMS1 http://
10.240.12.163:7001
```

where:

- *EBillingMS1* is the name you gave the managed server.
- *10.240.12.163* is the IP address of the computer where the Admin server is located.
- *7001* is the port number of the Admin server.

4 Start the proxy server. In this command, *EBillingProxy* is the name you gave the Proxy server:

```
$weblogic_home/common/bin/startManagedWebLogic.sh EBillingProxy http://
10.240.12.163:7001
```

where:

- *EBillingProxy* is the name you gave the Proxy server.
- *10.240.12.163* is the IP address of the computer where the Admin server is located.

- *7001* is the port number of the Admin server.

5 On each of the other computers in the cluster, follow these steps:

a Start the node manager:

```
$weblogic_home/server/bin/startNodeManager.sh
```

b Start each managed server, using the following command:

```
$weblogic_home /common/bin/startManagedWeblogic.sh EBillingMS2 http://
10.240.12.163:7001
```

where:

- *EBillingMS2* is the name you gave the managed server.
- *10.240.12.163* is the IP address of the computer where the Admin server is located.
- *7001*is the port number of the Admin server.

Stopping the Oracle WebLogic Cluster

Follow these instructions if you want to stop the Oracle WebLogic cluster where the Billing and Payment application is implemented.

To stop the Oracle WebLogic cluster

- Do one of the following:
  - Stop the process in the SSH window.
  - Go to the Admin console, using the following command, and then stop the managed server separately:

    http://*AdminServerIP:Port*/console

    where:

    - *AdminServerIP* is the IP address of the computer where the Admin server and the managed server are located.
    - *7001* is the port number of the managed server.

Configuring and Starting Scheduler on Oracle WebLogic

Scheduler is a program that administrators use to schedule Command Center jobs. Scheduler must be 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 Oracle WebLogic and UNIX” on page 117

■ "Configuring and Starting Scheduler on Oracle WebLogic on Windows" on page 118

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/oracle11gR2/11.2.0.3
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_12.1

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. In the lines, *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_12.1

@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 "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

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 In a clustered environment, use the Proxy server host and port, such as: like http://10.240.12.163:9007/ebilling
Command Center	http://localhost:7003/eBilling
Customer Service Representative	http://localhost:7006/ebillingcsr
Web Services Server	http://localhost:7017/ebillingrs
Web Services Client (for development use only)	http://localhost:7019/rsclient

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 Administrator
jsmith	jsmith	Administrator
twalsh	twalsh	B2C User
tbrown	tbrown	B2C User
ccadmin	Oracle1234	Command Center Administrator
csradmin	Oracle1234	Customer Service Representative Administrator

5 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

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 121
2. Creating the Oracle Workflow Manager on page 122
3. Process of Installing the Oracle Warehouse Builder Repository on page 123
4. Installing the ETL Module on page 130
5. Running the ETL Loader Job Using Sample Data on page 133

Related Topic

"Roadmap for Installing Oracle Self-Service E-Billing 6.1.1" on page 15

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 121.

To verify that you have the correct ETL Module system requirements

- Verify that your system has the necessary software installed:
  - Oracle Database 11*g* 11.2.0.3.
  - Oracle Warehouse Builder 11g Release 2 and Oracle Workflow; both products are automatically included with the Oracle Database installation.

Creating the Oracle Workflow Manager

You must create and configure an Oracle Workflow Manager.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121.

To create the Oracle Workflow Manager

1. Go to the Workflow directory.
2. Run the Oracle Workflow Configuration Assistant:
   - **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`.

   NOTE: The UNIX Repository Assistant requires a correctly configured X Window environment to run.
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:ebill_sid For example, if the host name is sdc50010qe.us.oracle.com, Port number of the EBILL database is 1521, and ebill_sid is ebill, then the format is: Sdc50010qe.us.oracle.com:1521:ebill.

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

This topic describes the process of installing the Oracle Warehouse Builder Repository.

This process is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121.

To install the Oracle Warehouse Builder Repository you must perform the following procedures:

1 "Tuning the OLAP Database" on page 123

2 "Creating the Oracle Warehouse Builder Repository Owner" on page 124

3 "Creating the Oracle Warehouse Builder Repository User" on page 125

4 "Creating an Oracle Warehouse Builder Project" on page 126

5 "Registering Users in Oracle Warehouse Builder 11g" on page 126

6 "Creating Database Modules" on page 127

7 "Creating Database Modules" on page 127

8 "Creating a Process Flow Module" on page 128

9 "Creating File System Locations" on page 129

10 "Registering Locations" on page 129

11 "Installing the ETL Module" on page 130 or "Installing the ETL Module" on page 130

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 123.

To tune the database parameters

1 Open the init*SID*.ora initialization file for the Oracle Self-Service E-Billing database, located in the following directory:

- **UNIX.** `$ORACLE_HOME/dbs`
- **Windows.** `%ORACLE_HOME%\database`

In the file name, *SID* is your EBILL database SID.

2 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

3 Restart the EBILL 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 "Process of Installing the Oracle Warehouse Builder Repository" on page 123.

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 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 If you are creating the first workspace for the repository, then 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 Select the language required for the repository, if necessary. The default is American English. Click Next.

14 Click Next to skip selecting a user.

The Repository Assistant Summary displays all the information related to repository.

15 Click Finish.

The progress bar shows the installation status.

16 When the progress bar is complete, click OK on the Installation Successful dialog box.

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 "Process of Installing the Oracle Warehouse Builder Repository" on page 123.

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, then 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 EBILL 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 123.

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, OLTP, 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 123.

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/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, then 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 and OLTP 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 and OLTP 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 reregister the OLAP or OLTP schema whenever the corresponding the schema refresh or reimage is done. If the OLAP or OLTP schema is not registered, then 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

You must 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 "Process of Installing the Oracle Warehouse Builder Repository" on page 123.

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 For each module, enter the module name, module location, EBILL service name, and schema user name as shown in the following table. (Be careful to specify the correct service name and schema user name for the MAPPINGS and the OLTP modules). Enter the schema password. The Oracle database parameters for each module must be the ones used for the Oracle Self-Service E-Billing installation. Use the default values for other fields. For example, the default module type is Warehouse Target. Click Next.

Module Name	Module Location	Service Name	Schema User Name
STAGE_TABLES	STAGE_TABLES_LOCATION	EBILL	OLAP
TARGET_TABLES	TARGET_TABLES_LOCATION	EBILL	OLAP
MAPPINGS	MAPPINGS_LOCATION	EBILL	MAP_USER
OLTP	OLTP_LOCATION	EBILL	OLTP

4 Click Edit next to the location field. Enter the location name and database connection details of the Oracle Self-Service E-Billing 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 123.

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 Process Flow Module to start the Create Module Wizard.

3 Create the module as shown in the following table. Enter the name of the module and click Next.

Module Name	Module Location Name	Service Name	Schema
ETL_PF_MODULE	EBILLING_ETL_LOCATION	EBILL	OWF_MGR

4 Click Edit next to the location field. Enter the location name. 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.

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. If the Create Process Flow Package window appears, then click Cancel.

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 123.

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 Navigator 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 123.

To register a location

1. In the Design Center, select the Tools menu, and click Control Center Manager.

   If you receive error RTC-5301, then the Runtime Platform Service cannot be started (Not Available), verify that you have added the parameter job_queue_processes to the EBILL database configuration.

   If the Control Center Service is still not available, then 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 the script runs successfully, then you receive the message Available. If the message Not Available displays, then shut down and restart the EBILL instance, and try the script again.
   - If Oracle Warehouse Builder is installed locally, then 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.

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 123.

For Windows, change the slashes (/ or \) and root directory as necessary.

This task is a step in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121.

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 following directory:

   - **UNIX.** *EDX_HOME*`/db/oracle`
   - **Windows.** *EDX_HOME*`\db\oracle`

   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, then OWB_ORACLE_HOME is the same as ORACLE_HOME.

2 Modify the following properties to reflect your Oracle Self-Service E-Billing environment. 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	The location of the Oracle Warehouse Builder home folder. If you are installing on Oracle Database 11*g*, this directory is the same as the ORACLE_HOME path, such as `export/home/oracle/11.2.0.3`
TCL_SCRIPTS_DIRECTORY	The directory where the Oracle Self-Service E-Billing TCL scripts are located, such as: *EDX_HOME*`/db/oracle/olap/etl`.
MDL_LOCATION	The folder location where the Oracle Self-Service E-Billing MDL file is located, such as: *EDX_HOME*`/db/oracle/olap/etl`
MDL_NAME	ebilling_etl_v6.1.1.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 Oracle Self-Service E-Billing database. Specify the HOST name of the Oracle Self-Service E-Billing database where the Oracle Warehouse Builder repository has been created.
OWB_PORT	The port number of the Oracle Self-Service E-Billing database, such as 1521.
OWB_SERVICE	The ORACLE_SID of the Oracle Self-Service E-Billing database.
SYS_PASSWD	The SYS password of the Oracle Self-Service E-Billing database.
OLAP_USER	The user name for the OLAP schema.
OLAP_PASSWD	The password for the OLAP schema.
EBILL_SID	The ORACLE_SID of the Oracle Self-Service E-Billing database.
OLTP_USER	The user name for the OLTP schema.
INPUT_DIR	Specify the input directory path, such as: `/export/home/oracle/testdata/OWF_INDATA`

Property	Value
OUTPUT_DIR	Specify the output directory path, such as: /export/home/oracle/testdata/OWF_OUTDATA
REJECT_DIR	Specify the reject directory path, such as: /export/home/oracle/testdata/OWF_REJDATA
GRP_MOVE_INDIR	Specify the group move input directory path, such as: /export/home/oracle/testdata/GRPMV_INDATA
GRP_MOVE_OUTDIR	Specify the group move output directory path, such as: /export/home/oracle/testdata/GRPMV_OUTDATA

3 Specify the following mail server parameters.

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.

4 Open a command prompt, and go to the directory containing the ebilling_etl.properties file.

5 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 ANT_HOME=/opt/apache-ant-1.6.5

set PATH=%PATH%;%ANT_HOME%/bin
```

6 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

7 Enter Ant to run the build script.

By default, the Ant command runs the build.xml file in the current directory.

8 From the Main Menu, select Option 2, Standalone Install.

9 Select Option 5, ETL Setup.

10 You have the option to create the ETL packages, import the MDL file, and deploy the OWB mappings in one step or in separate steps:

- To create the ETL packages, import the MDL file, and deploy the OWB mappings in one step, select Option 3, Run Steps 1 and 2, from the ETL Setup menu. Ant returns to the current menu when finished. When these steps are complete, go to Step 16.

- To create the ETL packages, import the MDL file, and deploy the OWB mappings in separate steps, continue with Step 11.

11 Select Option 1, Import MDL.

This step can be used to create the ETL packages and import the MDL file.

12 Check the MDL*FileName*.log file, found in the *EDX_HOME*/db/oracle/olap/etl directory, for errors. Log in to the Design Center, and make sure all modules are connected to the database locations. For the ETL_PF_MODULE module, check the LOCATION and Evaluation Location.

13 Review all log files for possible errors

14 Select Option 2, OWB Mappings Deployment.

Ant returns to the current menu when finished.

15 Review all log files for possible errors

16 Review all log files found in the *EDX_HOME*/db/oracle/olap/etl and *EDX_HOME*/db/oracle directories. You can ignore the following error message shown on screen and in the ETLPackages.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
```

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 121.

To run the ETL Loader job using sample data

1 If you are using the Telco version of Oracle Self-Service E-Billing, then copy the sample data files from the following directory to the ETL input directory. In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

- **UNIX.** *EDX_HOME*\db\oracle\olap\etl\sample_data\Telco
- **Windows.** *EDX_HOME*/db/oracle/olap/etl/sample_data\Telco

Move the following sample Telco data files:

- MASTER-DATA-FILE-20100509.DAT
- EBILLING_B2B-DATA-FILE-20100510.DAT
- EBILLING_B2B-DATA-FILE-20100512.DAT
- EBILLING_B2B-DATA-FILE-20100551.DAT
- EBILLING_B2B-DATA-FILE-20100552.DAT
- EBILLING_B2B-DATA-FILE-20100553.DAT
- EBILLING_B2B-DATA-FILE-20101101.DAT
- EBILLING_B2B-DATA-FILE-20120528.DAT
- EBILLING_B2B-DATA-FILE-20120618.DAT
- EBILLING_B2B-DATA-FILE-20120630.DAT
- EBILLING_B2C-DATA-FILE-20100516.DAT
- EBILLING_B2C-DATA-FILE-20101101.DAT
- EBILLING_B2C-DATA-FILE-20120528.DAT
- EBILLING_B2C-DATA-FILE-20120618.DAT
- EBILLING_B2C-DATA-FILE-20120630.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 If you are using the Utility version of Oracle Self-Service E-Billing, then copy the sample data files from the following directory to the ETL input directory:

- **UNIX.** *EDX_HOME*\db\oracle\olap\etl\sample_data\Utilities
- **Windows.** *EDX_HOME*/db/oracle/olap/etl/sample_data/Utilities

Move the following sample Utility data files:

- MASTER-UTILITIES-20110101.DAT
- EBILLING_B2C-UTILITIES-20110101.DAT
- EBILLING_B2C-UTILITIES-20120528.DAT
- EBILLING_B2C-UTILITIES-20120618.DAT
- EBILLING_B2C-UTILITIES-20120630.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.

3 Log in to the Oracle Warehouse Builder Design Center as the repository owner. Start the Control Center from the Tools menu.

4 Select Tools, Preferences, OWB, and then Deployment Menu. Make sure the Prompt for execution parameters option is enabled.

5 Expand the EBILLING_ETL_LOCATION, ETL_PF_MODULE, and ETL_SUP nodes.

6 In the Command Center, run the HierarchyCopy job to the period specified in the data file. Also run the Hierarchy Copy job if the new load file, sample or real data, is for a new period.

For details on running the HierarchyCopy job, see *Administration Guide for Oracle Self-Service E-Billing*.

7 Right click the EBILL_DATALOAD 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, then click OK. Or you can run the ant command again from the following directory:

- **UNIX.** *EDX_HOME*/db/oracle
- **Windows.** *EDX_HOME*\db\oracle

8 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 comply with the Payment Card Industry Data Security Standard.

9 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 comply with the Payment Card Industry Data Security Standard.

6 Migrating to Oracle Self-Service E-Billing Version 6.1.1

This chapter describes how to migrate to Oracle Self-Service E-Billing Version 6.1.1. It includes the following topics:

Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1

This topic describes the tasks necessary to migrate from Oracle Self-Service E-Billing 6.0.x to Oracle Self-Service E-Billing 6.1.1.

To migrate to Oracle Self-Service E-Billing version 6.1.1, 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 11*g* 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, then contact the database administrator to correct this problem. For additional information, see Oracle Database 11*g* documentation on Oracle Technology Network.

3. Have any existing Oracle Self-Service E-Billing database passwords available. Check with your database administrator.

4. From the following list, follow only the steps necessary to incrementally migrate your implementation to Oracle Self-Service E-Billing version 6.1.1:

   a. “Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1” on page 138

   b. “Process of Migrating Oracle Self-Service E-Billing 6.0.1 to 6.0.2” on page 146

   c. “Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1” on page 153

   d. “Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4” on page 161

   e. “Process of Migrating Oracle Self-Service E-Billing from 6.0.4 to 6.1” on page 168

f "Process of Migrating Oracle Self-Service E-Billing from 6.1 to 6.1.1" on page 173

Process of Migrating Oracle Self-Service E-Billing 6.0 to 6.0.1

This topic describes the process required to migrate Oracle Self-Service E-Billing version 6.0 to 6.0.1.

This process is a step in the following roadmaps:

- "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1" on page 137
- "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 196

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 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 17.
3. "Migrating Oracle Self-Service E-Billing 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 24.

Migrating Oracle Self-Service E-Billing to Oracle Database 11*g*

If you are migrating Oracle Self-Service E-Billing from Oracle Database 10*g* to Oracle Database 11*g*, then you must follow these procedures. These steps are for both UNIX and Windows environments.

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. Edit the edxadmin_unix.properties file, located in the following directory:
   - **UNIX.** *EDX_HOME*/db/ebilling/oracle
   - **Windows.** *EDX_HOME*\db\ebilling\oracle
5. Verify that the following values are correct for the current installation:
   - 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
6. Create the Oracle Self-Service E-Billing database:
   a. Go to the location of the Oracle Database installation files in your software installation:
      - **UNIX.** *EDX_HOME*/db/ebilling/oracle
      - **Windows**. *EDX_HOME*\db\ebilling\oracle
   b. If you have not configured the Apache Ant environment, then do so now:
      - **UNIX.** Run the following commands, where *JDK150_11* is your JDK version:

        ```
        export ANT_HOME=/opt/apache-ant-1.6.5

        export PATH=$ANT_HOME/bin:$PATH

        export JAVA_HOME= $WEBLOGIC_HOME/JDK150_11

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

      - **Windows**. Run the following commands, where *JDK150_11* is your JDK version:

        ```
        set ANT_HOME=C:\apache-ant-1.6.5

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

        set JAVA_HOME= %WEBLOGIC_HOME%\JDK150_11
        ```

   c. Enter Ant to run the build script.

      By default, the Ant command runs the build.xml file in the current directory.

   d. 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).

7 Create a new Oracle Database 11*g* database instance, tablespace, and user for OLTP (eStatement):

a Edit the edxadmin.properties file, located in the following directory:

- **UNIX.** *EDX_HOME*/db/eStatement/oracle
- **Windows.** *EDX_HOME*\db\eStatement\oracle

b Verify that the following values are correct for the current installation:

- 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

c 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

d Run the commands to configure the Apache Ant environment.

- **UNIX.** Run the following commands, where *JDK150_11* is your JDK version:

```
export ANT_HOME=/opt/apache-ant-1.6.5
export JAVA_HOME=$WEBLOGIC_HOME/JDK150_11
export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$PATH
```

- **Windows.** Run the following commands, where *JDK150_11* is your JDK version:

```
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
```

In the set JAVA_HOME command, *JDK150_11* is your JDK version.

e Enter Ant to run the build script.

By default, the Ant command runs the build.xml file in the current directory.

f From the top level Main Menu, select Option 1, Install eStatement Database to start.

The Install eStatement Database menu appears.

g 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).

8 Create the database link, TAM_LINK, in the OLTP schema.

9 Log on to the OLTP instance as SYSDBA, and execute the following command. In the command, *OLTP_Schema* is the name of the OLTP schema.

```
SQL> GRANT CREATE DATABASE LINK TO OLTP_Schema;
```

10 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle
- **Windows**. *EDX_HOME*\db\ebilling\oracle

11 Log on to the OLTP schema, using SQL*Plus, not as SYSDBA.

12 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.

13 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, then 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, then follow these steps to migrate your OLTP database.

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 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.

2 Go to the *EDX_HOME*/eBilling/db/ebilling/oracle/oltp/migration/Ebilling_6.0_to_6.0.1 directory.

3 Verify that the following files exist:

- `migrate_oltp_6.0_to_6.0.1.sh`
- `migrate_oltp_6.0_to_6.0.1.sql`

4 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
```

5 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

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 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, then follow these steps to migrate your OLTP database.

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 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. In the directory, *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,. In the script, *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 script as shown. In this script, *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, then follow these steps to migrate your OLAP database.

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 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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle/olap/migration/ebilling_6.0_to_6.0.1
- **Windows**. *EDX_HOME*\db\ebilling\oracle\olap\migration\ebilling_6.0_to_6.0.1

3 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

4 Create the database link (OLTP_LINK) in the OLAP schema. Log on to the OLAP instance as SYSDBA, and run the following command. In the command, *OLAP_Schema* is the name of the OLAP schema.

```
GRANT CREATE DATABASE LINK TO OLAP_Schema;
```

5 Go to the following directory:

- **UNIX.** *EDX_HOME*/eBilling/db/ebilling/oracle

- **Windows**. *EDX_HOME*\eBilling\db\ebilling\oracle

6 Log on to the OLAP schema, using SQL*Plus, not as SYSDBA.

7 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.

8 (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 11*g* database, then 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.

9 (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*, then 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.

2 Go to the following directory:

- **UNIX.** /opt/Oracle/eBilling/db/ebilling/oracle

- **Windows**. `\Oracle\eBilling\db\ebilling\oracle`

3. Log on to the OLTP schema, using SQL*Plus, not as SYSDBA.
4. Run the following commands:

```
SQL>@ compile_schema.sql

SQL>exit;
```

5. Log on to the OLAP schema, using SQL*Plus, not as SYSDBA.
6. 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

This topic describes the process required to migrate Oracle Self-Service E-Billing version 6.0.1 to 6.0.2.

This process is a step in the following roadmaps:

- "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1" on page 137
- "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 196

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 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 17.
3. "Migrating Oracle Self-Service E-Billing 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 153

8 Uninstall Oracle Self-Service E-Billing version 6.0.1. For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 24.

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*, then you must follow these procedures. These steps are for both UNIX and Windows environments.

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 (Single Node)" on page 29. 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. In this command, *OLTP_Schema* is the name of the OLTP schema.

```
GRANT CREATE DATABASE LINK TO OLTP_Schema;
```

b Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle
- **Windows**. *EDX_HOME*\db/ebilling\oracle

c Log on to the OLTP schema, using SQL*Plus, not as SYSDBA, and run the following command:

```
DROP DATABASE LINK TAM_LINK;
```

d 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. In this command, *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 "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 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 "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 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. In the directory, *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 "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 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. In the directory, *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*, then 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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle
- **Windows**. *EDX_HOME*\db/ebilling\oracle

3 Log on to the OLTP schema, using SQL*Plus, not as SYSDBA, and run the following commands:

```
SQL>@ compile_schema.sql

SQL>exit;
```

4 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, then specify www-*ExampleProxy.com*, where *ExampleProxy.com* is your proxy setting.
Proxy port	(Optional field) 80. If you are using proxy for network connection, then enter your proxy setting.
Paypal user	eaSuite47
Paypal vender	eaSuite47
Paypal partner	PayPal
Paypal password	eaSuite47
URLStreamHandler class of application server	Oracle WebLogic: sun.net.www.protocol.https.Handler

Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1

This topic describes the process required to migrate Oracle Self-Service E-Billing version 6.0.2 to 6.1.1. Run the migration process on the Oracle Self-Service E-Billing database server only.

This process is a step in the following roadmaps:

- "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1" on page 137
- "Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 196

To migrate from Oracle Self-Service E-Billing version 6.0.2 directly to 6.1.1, perform the following tasks:

1. Make sure that your installation of Oracle Self-Service E-Billing 6.0.2 is at patch 9.
2. Write down your payment gateway and job configuration settings as shown in the Command Center. You must reenter the settings after migrating.
3. Back up your existing OLTP and OLAP Oracle Self-Service E-Billing databases.

   For additional information, see Oracle Database 11*g* documentation on Oracle Technology Network.

   Run the Notifier job to process all pending notifications

   For details on how to run the Notifier job, see *Administration Guide for Oracle Self-Service E-Billing*.
4. 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 6.1.1.

   For details on how to run batch report jobs, see *Administration Guide for Oracle Self-Service E-Billing*.
5. Upgrade your Oracle Database 11*g* to version 11.2.0.3.

   For additional information, see Oracle Database 11*g* documentation on Oracle Technology Network.
6. Update your Oracle Warehouse Builder (OWB) repository to version 11.2.0.3.
7. Perform the following installation tasks:
   a. "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 16
   b. "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 17. Select Option 2, Oracle E-Billing and Migration Tools on the Product Features Screen.
   c. "Configuring Log File Paths for Log4j" on page 19
8. Perform the following migration tasks:
   a. "Migrating Oracle Self-Service E-Billing Version 6.0.2 OLTP to 6.1.1" on page 155
   b. "Migrating Oracle Self-Service E-Billing Version 6.0.2 OLAP to 6.1.1" on page 158
   c. "Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.0.2 to 6.1.1" on page 160

   NOTE: Some errors will occur when migrating the OLTP and OLAP databases because some database objects reference each other. Do not take any actions when the errors occur.
9. Run the following command on both the OLTP and OLAP schemas to verify that all objects are valid. The correct result is zero. Connect as the OLTP and OLAP schema owner.

   ```
   select count(*) from user_objects where status ='INVALID';
   ```

10 Configure your application server for Oracle Self-Service E-Billing 6.1.1.

Follow "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

11 Follow the tasks and processes in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121. (Be sure to register the OLTP user in the Oracle Warehouse Builder, and verify that the OLTP location points to the OLAP instance.)

12 Uninstall Oracle Self-Service E-Billing version 6.0.2.

For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 24.

13 Follow the steps in "Creating a Bootstrap Administrator User for Oracle Self-Service E-Billing" on page 56 to create the bootstrap, or default, Oracle Self-Service E-Billing administrative user.

14 Recreate the payment gateway and jobs, using the Command Center.

For details, see *Administration Guide for Oracle Self-Service E-Billing*.

Migrating Oracle Self-Service E-Billing Version 6.0.2 OLTP to 6.1.1

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.2 OLTP to 6.1.1 for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1" on page 153.

To migrate Oracle Self-Service E-Billing 6.0.2 OLTP to 6.1.1

1 On UNIX, log in as the ORACLE user for migration activity.

2 Download the hibernate-3.1.3.jar file from the following Web site:

http://search.maven.org/remotecontent?filepath=org/hibernate/hibernate/3.1.3/hibernate-3.1.3.jar

3 Place the hibernate-3.1.3.jar file in the following directory:

- **UNIX:** *EDX_HOME*/db/oracle/oltp/migration/ebilling6.0.2_to_6.1.1/lib/lib
- **Windows:** *EDX_HOME*\db\oracle\oltp\migration\ebilling6.0.2_to_6.1.1\lib\lib

4 Open the migrate_oltp_6.0.2_to_6.1.1.properties file found in the following directory:

- **UNIX:** *EDX_HOME*/db/oracle/oltp/migration/ebilling6.0.2_to_6.1.1
- **Windows:** *EDX_HOME*\db\oracle\oltp\migration\ebilling6.0.2_to_6.1.1

5 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
LISTEN_PORT	The listening port for the new EBILL SID
OLTP_HOST	OLTP database host name or IP address (to be used for the new EBILL database)
OLTP_SYS_PASSWD	OLTP sys user password
OLAP_USER	OLAP schema user name
OLAP_PASSWD	OLAP schema password
EBILL_SID	Your existing OLAP SID (instance name), to be used for the new EBILL database.
OLAP_SYS_PASSWD	OLAP sys user password
L_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Property	Value
L_DB_LOAD_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_FS_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_FS_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_STG_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_STG_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
LARGE_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
LARGE_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.

Property	Value
MEDIUM_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
MEDIUM_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.

NOTE: Use a backslash (\) as the path separation character on Windows.

6 Save and close the migrate_oltp_6.0.2_to_6.1.1.properties file.

7 Run the following command:

```
ant -f  migrate_oltp_6.0.2_to_6.1.1.xml
```

8 Select Option 1, Export OLTP schema.

9 Select Option 2, Create OLTP tablespaces and OLTP user.

10 Select Option 3, Import OLTP schema into OLAP instance.

11 Select Option 4, Prepare E-Billing 6.0.2 OLTP db.

12 Select Option 5, Data Re-Encryption.

13 Select Option 6, Post Re-Encryption cleanup.

14 Select Option 7, Migrate OLTP schema.

15 Select Option 8, Migrate OLTP data.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

16 Review all log files for possible errors.

Migrating Oracle Self-Service E-Billing Version 6.0.2 OLAP to 6.1.1

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.2 OLAP to 6.1.1 for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1" on page 153.

To migrate Oracle Self-Service E-Billing 6.0.2 OLAP to 6.1.1

1 On UNIX, log in as the ORACLE user for migration activity.

2 Open the migrate_olap_6.0.2_to_6.1.1.properties file found in the following directory:

- **UNIX:** *EDX_HOME*/db/oracle/olap/migration/ebilling6.0.2_to_6.1.1
- **Windows:** *EDX_HOME*\db\oracle\olap\migration\ebilling6.0.2_to_6.1.1

3 Set the correct value for each property in the file.

Property	Value
OLAP_USER	OLAP schema user name
OLAP_PASSWD	OLAP schema password
EBILL_SID	Your existing OLAP SID (instance name), to be used for the new EBILL database.
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password
SYS_PASSWD	OLAP SYS user password
L_DB_EDX_DATA_TB_FILE_LOC	Data tablespace file location, for example: /export/home/oracle/oradata
GRP_MOVE_INDIR	The input directory path to use when moving ETL groups, for example: /export/home/oracle/testdata/GRPMV_INDATA
GRP_MOVE_OUTDIR	The output directory path to use when moving ETL groups, for example: /export/home/oracle/testdata/GRPMV_OUTDATA

4 Save and close the migrate_oltp_6.0.2_to_6.1.1.properties file.

5 Run the following command:

```
ant -f  migrate_olap_6.0.2_to_6.1.1.xml
```

6 Select Option 1, Create New Staging Tablespaces for OLAP schema.

7 Select Option 2, Migrate OLAP Schema.

8 Select Option 3, Drop redundant object.

9 Select Option 4, Migrate OLAP data.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

10 Review all log files for possible errors.

Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.0.2 to 6.1.1

After you have successfully migrated from Oracle Self-Service E-Billing 6.0.2 to 6.1.1, you must copy the batch reports to the directory you want to use for reports in the Oracle Self-Service E-Billing version 6.1.1.

If you copy the reports to a different location than you used in version 6.0.2, then you must update the batch report file location in the OLTP database. Also, if you installed Oracle Self-Service E-Billing 6.1.1 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* directory or the *output**reportapp* directory on Windows), then 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 6.0.2 to 6.1.1" on page 153

To migrate your batch reports from Oracle Self-Service E-Billing 6.0.2 to version 6.1.1

1. On the application server, copy your existing batch reports to the following directory:
   - **UNIX.** *EDX_HOME*\output\reportapp
   - **Windows.** *EDX_HOME*/output/reportapp

   In the directory, *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing 6.1.1.
2. If you copy your batch reports to a different file location in version 6.1.1 than you used in version 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 6.1.1.
   - *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/
   eBilling611/output/reportapp' || substr(file_location,length('/export/home/
   oracle/eBilling/output/reportapp')+1) where file_location is not null;

   SQL>commit;
   ```

3 If you installed Oracle Self-Service E-Billing 6.1.1 in a directory other than the default *EDX_HOME* directory, or if you plan to use a customized batch report location, then you must update the properties in the reporting.batch.xma.xml file, located in the following directory. The default batch report directory is *output/reportapp*, or the *output\reportapp* directory on Windows.

- **UNIX.** *EDX_HOME*/xma/config/com/edocs/common/reporting/
- **Windows.** *EDX_HOME*\xma\config\com\edocs\common\reporting\

Specify your installation root directory in the rootDir property, and specify the custom report subdirectory in the path property. Report files are stored in the *rootDir/path* directory, or the *rootDir/path* directory on Windows:

```
<!-- 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

This topic describes the process required to migrate Oracle Self-Service E-Billing version 6.0.3 to 6.0.4.

This process is a step in the following roadmaps:

- “Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1” on page 137
- “Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1” on page 181
- “Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.1.1” on page 192
- “Roadmap for Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing Version 6.1.1” on page 196.

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 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 Run the migration process on the Oracle Self-Service E-Billing 6.0.4 database server only. Perform the following installation tasks:

   a "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 16

   b "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 17. select Option 2, Oracle E-Billing and Migration Tools on the Product Features Screen.

   c "Configuring Log File Paths for Log4j" on page 19

   d "Adding Foreign Language Fonts to Your Application Server" on page 21

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 162

   b "Migrating Oracle Self-Service E-Billing Version 6.0.3 OLTP to 6.0.4" on page 164

   c "Migrating Oracle Self-Service E-Billing Version 6.0.3 OLAP to 6.0.4" on page 165

   d "Compiling the Schema for the Oracle Self-Service E-Billing 6.0.4 OLTP and OLAP Databases" on page 166

   e "Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.0.3 to 6.0.4" on page 167

6 Configure your application server for Oracle Self-Service E-Billing 6.0.4. Follow "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

7 "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121

8 Uninstall Oracle Self-Service E-Billing version 6.0.3. For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 24.

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, then 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 62421.1 (Article ID) 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 161.

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 articles:
   - For information on installing and configuring CSSCAN in Oracle Database 11*g*, see 745809.1 (Article ID) on My Oracle Support.
   - For information on changing the NLS_CHARACTERSET to AL32UTF8 or UTF8 (Unicode), see 260192.1 (Article ID) on My Oracle Support.
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.** Run the following commands:

     ```
     export NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

     exp system/manager@oltp_tnsname file=oltp_export.dmp FULL=Y
     log=oltp_export.log
     ```

   - **Windows.** Run the following commands:

     ```
     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. Configure the property file and install the Oracle OLTP database. Follow Step 1 through Step 8 in "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29.
   b. Select Option 1, Create Database Instances, and then select Option 2, Create an Oracle OLTP Oracle Instance.

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

   c. Restart the database, and then select Option 4, Return to Main Menu.
   d. Select Option 2, Standalone Install, select Option 2, Create eBilling Schemas, and then select Option 2, Create Database Users. When Ant returns to the current menu, check all log files for errors.
6. Create the database link, TAM_LINK, in the OLTP schema. Log on to the OLTP instance as SYSDBA, and run the following command. In this command, *OLTP_Schema* is the name of the OLTP schema.

```
SQL> GRANT CREATE DATABASE LINK TO OLTP_Schema;
```

7 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle
- **Windows.** *EDX_HOME*\db\ebilling\oracle

8 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.

9 Import the exported data into the new AL32UTF8 character set. Note that the NLS_LANG setting uses the source character set.

- **UNIX.** Run the following commands:

```
export NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1

imp system/manager@oltp_sid fromuser=oltp touser=oltp file=oltp_export.dmp
log=oltp_import.log
```

- **Windows.** Run the following commands:

```
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 161.

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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle/oltp/migration/ebilling6.0.3_to_6.0.4

- **Windows.** *EDX_HOME*\db\ebilling\oracle\oltp\migration\ebilling6.0.3_to_6.0.4

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 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 separation character for paths on Windows.

4 Save and close the migrate_oltp_6.0.3_to_6.0.4.properties file.

5 Run following command to start migration:

```
ant -f migrate_oltp_6.0.3_to_6.0.4.xml
```

6 Select Option 1, Migrate OLTP Database.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

7 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 "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 161.

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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle/olap/migration/ebilling6.0.3_to_6.0.4
- **Windows.** *EDX_HOME*\db\ebilling\oracle\olap\migration\ebilling6.0.3_to_6.0.4

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 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.

4 Save and close the migrate_olap_6.0.3_to_6.0.4.properties file.

5 Run following command to start migration:

```
ant -f migrate_olap_6.0.3_to_6.0.4.xml
```

6 Select Option 1, Migrate OLAP db.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

7 If the build was successful, then select Option 2, Repopulate Summary Tables. Query the edx_rpt_etl_log table to see log information on the data repopulation.

8 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 "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 161.

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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle
- **Windows.** *EDX_HOME*\db\ebilling\oracle

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Log on to the OLTP schema, using SQL*Plus, not as SYSDBA, and run the following commands:

```
SQL>@ compile_schema.sql

SQL>exit;
```

4 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 Version 6.0.3 to 6.0.4

After 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, then 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* directory or the *output**reportapp* directory on Windows), then 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 161.

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 the following directory:

- **UNIX.** *EDX_HOME*\output\reportapp
- **Windows.** *EEDX_HOME*/output/reportapp

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

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, then you must update the batch report file location in the OLTP database. Log on to the OLTP schema, 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 directory is *output/reportapp*, or the *output**reportapp* directory on Windows), then you must update the properties in the `reporting.batch.xma.xml` file, located in the following directory:

- **UNIX.** *EDX_HOME*`/xma/config/com/edocs/common/reporting/`
- **Windows.** *EDX_HOME*`\xma\config\com\edocs\common\reporting\`

Specify your installation root directory in the `rootDir` property, and specify the custom report subdirectory in the path property (report files are stored in the *rootDir/path* directory, or the *rootDir/path* directory on Windows):

```
<!-- 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 from 6.0.4 to 6.1

You must run the migration process on the Oracle Self-Service E-Billing 6.0.4 database server only.

This process is a step in "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1" on page 137.

To migrate from Oracle Self-Service E-Billing version 6.0.4 to 6.1.1, 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 11*g* documentation on Oracle Technology Network.

2 Process all pending notifications in Oracle Self-Service E-Billing 6.0.4.

For details on how to run the Notifier job, see *Administration Guide for Oracle Self-Service E-Billing*.

3 Process all pending batch reports in Oracle Self-Service E-Billing 6.0.4.

For details on how to run the batch report jobs, see *Administration Guide for Oracle Self-Service E-Billing*.

4 Migrate to Oracle Database 11*g* (Version 11.2.0.3) if needed, using the steps described in "Migrating Oracle Self-Service E-Billing to Oracle Database 11g" on page 138.

5 Perform the following installation tasks:

a "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 16

b "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 17. Select Option 2, Oracle E-Billing and Migration Tools on the Product Features Screen.

c "Configuring Log File Paths for Log4j" on page 19

6 Perform the following migration tasks:

a "Migrating Oracle Self-Service E-Billing Version 6.0.4 OLTP to Version 6.1" on page 169

b "Migrating Oracle Self-Service E-Billing Version 6.0.4 OLAP to Version 6.1" on page 170

c "Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.0.4 to 6.1" on page 171

NOTE: Some errors will occur when migrating the OLTP and OLAP databases because some database objects reference each other. Do not take any actions when the errors occur.

7 Run the following command on both the OLTP and OLAP instances to verify that all objects are valid. The correct result is zero. Connect as the OLTP and OLAP schema owner.

```
select count(*) from user_objects where status ='INVALID';
```

8 Configure your application server for Oracle Self-Service E-Billing 6.1.1.

Follow "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

9 Follow the tasks and processes in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121.

10 Uninstall Oracle Self-Service E-Billing version 6.0.4.

For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 24.

11 Recreate the payment gateway and Command Center jobs after migration.

Migrating Oracle Self-Service E-Billing Version 6.0.4 OLTP to Version 6.1

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.4 OLTP to 6.1 for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing from 6.0.4 to 6.1" on page 168.

To migrate Oracle Self-Service E-Billing 6.0.4 OLTP to version 6.1

1 On UNIX, log in as the ORACLE user for migration activity.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle/oltp/migration/ebilling6.0.4_to_6.1
- **Windows.** *EDX_HOME*\db\ebilling\oracle\oltp\migration\ebilling6.0.4_to_6.1

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Open the migrate_oltp_6.0.4_to_6.1.properties file, and set the correct value for each property in this file.

Property	Value
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password
OLTP_SID	OLTP instance name
SYS_PASSWD	OLTP SYS user password

4 Save and close the migrate_oltp_6.0.4_to_6.1.properties file.

5 Run the following command to start the migration:

```
ant -f migrate_oltp_6.0.4_to_6.1.xml
```

6 Select Option 1, Migrate OLTP schema.

7 Select Option 2, Migrate OLTP data.

BUILD SUCCESSFUL displays on the console screen if the migration is successful.

8 Review all log files for possible errors.

Migrating Oracle Self-Service E-Billing Version 6.0.4 OLAP to Version 6.1

Follow these steps to migrate Oracle Self-Service E-Billing 6.0.4 OLAP to 6.1 for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing from 6.0.4 to 6.1" on page 168.

To migrate Oracle Self-Service E-Billing 6.0.4 OLAP to Version 6.1

1 On UNIX, log in as the ORACLE user for migration activity.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/oracle/olap/migration/ebilling6.0.4_to_6.1
- **Windows.** *EDX_HOME*\db\oracle\olap\migration\ebilling6.0.4_to_6.1

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Open the migrate_olap_6.0.4_to_6.1.properties file, and set the correct value for each property in this file.

Property	Value
OLAP_USER	OLAP schema user name
OLAP_PASSWD	OLAP schema password
OLAP_SID	OLAP instance name
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password
OLTP_SID	OLTP instance name
SYS_PASSWD	OLAP SYS user password
L_DB_EDX_DATA_TB_FILE_LOC	Data tablespace file location, for example: `/export/home/oracle/oradata`
GRP_MOVE_INDIR	The input directory path to use when moving ETL groups, for example: `/export/home/oracle/testdata/GRPMV_INDATA`
GRP_MOVE_OUTDIR	The output directory path to use when moving ETL groups, for example: `/export/home/oracle/testdata/GRPMV_OUTDATA`

4 Save and close the migrate_olap_6.0.4_to_6.1.properties file.

5 Run the following command to start the migration:

```
ant -f migrate_olap_6.0.4_to_6.1.xml
```

6 Select Option 1, Create New Staging Tablespaces for OLAP schema.

7 Select Option 2, Migrate OLAP Schema.

8 Select Option 3, Drop redundant object.

9 Select Option 4, Migrate OLAP data.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

10 Review all log files for possible errors.

Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.0.4 to 6.1

After you have successfully migrated from Oracle Self-Service E-Billing 6.0.4 to 6.1, you must copy the batch reports to the directory you want to use for reports in the Oracle Self-Service E-Billing version 6.1.

If you copy the reports to a different location than you used in version 6.0.4, then you must update the batch report file location in the OLTP database. Also, if you installed Oracle Self-Service E-Billing 6.1 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* directory or the *output\reportapp* directory on Windows), then 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 from 6.0.4 to 6.1" on page 168.

To migrate your batch reports from Oracle Self-Service E-Billing 6.0.4 to version 6.1

1 On the application server, copy your existing batch reports to the following directory:

- **UNIX.** *EDX_HOME*\output\reportapp
- **Windows.** *EDX_HOME*/output/reportapp

In the directory, *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing 6.1.

2 If you copy your batch reports to a different file location in version 6.1 than you used in version 6.0.4, 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 6.1.
- *OLD_LOCATION* is the old location where you stored batch reports in Oracle Self-Service E-Billing 6.0.4.

For example:

```
SQL> update edx_rpt_batch_report set file_location = '/export/home/oracle/
eBilling61/output/reportapp' || substr(file_location,length('/export/home/
oracle/eBilling/output/reportapp')+1) where file_location is not null;

SQL>commit;
```

3 If you installed Oracle Self-Service E-Billing 6.1 in a directory other than the default *EDX_HOME* directory, or if you plan to use a customized batch report location (the default directory is *output/reportapp*, or the *output\reportapp* directory on Windows), then you must update the properties in the reporting.batch.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/com/edocs/common/reporting/
- **Windows.** *EDX_HOME*\xma\config\com\edocs\common\reporting\

Specify your installation root directory in the rootDir property, and specify the custom report subdirectory in the path property (report files are stored in the *rootDir/path* directory, or the *rootDir/path* directory on Windows):

```
<!-- 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 from 6.1 to 6.1.1

You must run the migration process on the Oracle Self-Service E-Billing 6.1 database server only.

This process is a step in "Roadmap for Migrating Oracle Self-Service E-Billing 6.0.x to Version 6.1.1" on page 137.

To migrate from Oracle Self-Service E-Billing version 6.1 to 6.1.1, 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 11*g* documentation on Oracle Technology Network.

2. Process all pending notifications in Oracle Self-Service E-Billing 6.1.

   For details on how to run the Notifier job, see *Administration Guide for Oracle Self-Service E-Billing*.

3. Process all pending batch reports in Oracle Self-Service E-Billing 6.1.

   For details on how to run the batch report jobs, see *Administration Guide for Oracle Self-Service E-Billing*.

4. Write down your payment gateway and job configuration settings as shown in the Command Center. You must reenter the settings after migrating.

5. Upgrade your Oracle Database 11*g* to version 11.2.0.3, if needed.

   For additional information, see Oracle Database 11*g* documentation on Oracle Technology Network.

6. If you implemented Transparent Data Encryption with the 6.1 OLTP database, make sure the Oracle Wallet is open.

7. Perform the following installation tasks:

   a. "Checking the Integrity of the Oracle Self-Service E-Billing Installer Package" on page 16

   b. "Installing the Oracle Self-Service E-Billing Software Using InstallAnywhere" on page 17. Select Option 2, Oracle E-Billing and Migration Tools on the Product Features Screen.

c "Configuring Log File Paths for Log4j" on page 19

8 Perform the following migration tasks:

a "Migrating Oracle Self-Service E-Billing Version 6.1 OLTP to Version 6.1.1" on page 174

b "Migrating Oracle Self-Service E-Billing Version 6.1 OLAP to Version 6.1.1" on page 177

c "Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.1 to 6.1.1" on page 178

9 Run the following command on both the OLTP and OLAP schemas to verify that all objects are valid. The correct result is zero. Connect as the OLTP and OLAP schema owner.

```
select count(*) from user_objects where status ='INVALID';
```

10 If you had previously implemented Transparent Data Encryption on the 6.1 database, then you must reimplement it on the 6.1.1 database. To implement Transparent Data Encryption on the EBILL (OLAP) instance, see "Choosing a Database Encryption Method" on page 45.

11 Configure your application server for Oracle Self-Service E-Billing 6.1.1.

Follow "Roadmap for Configuring Oracle WebLogic for Oracle Self-Service E-Billing" on page 61.

12 Follow the tasks and processes in "Roadmap for Installing the ETL Module for Oracle Self-Service E-Billing" on page 121. (Be sure to register the OLTP user in the Oracle Warehouse Builder, and verify that the OLTP location points to the OLAP instance.)

13 Uninstall Oracle Self-Service E-Billing version 6.1.

For information about uninstalling, see "Uninstalling Oracle Self-Service E-Billing" on page 24.

14 Recreate the payment gateway and jobs, using the Command Center.

For details, see *Administration Guide for Oracle Self-Service E-Billing*.

Migrating Oracle Self-Service E-Billing Version 6.1 OLTP to Version 6.1.1

Follow these steps to migrate Oracle Self-Service E-Billing 6.1 OLTP to 6.1.1 for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing from 6.1 to 6.1.1" on page 173.

To migrate Oracle Self-Service E-Billing 6.1 OLTP to version 6.1.1

1 On UNIX, log in as the ORACLE user for migration activity.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle/oltp/migration/ebilling6.1_to_6.1.1
- **Windows.** *EDX_HOME*\db\ebilling\oracle\oltp\migration\ebilling6.1_to_6.1.1

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Open the migrate_oltp_6.1_to_6.1.1.properties file, and set the correct value for each property in this file.

Property	Value
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password
OLTP_SID	OLTP instance name
OLTP_SYS_PASSWD	OLTP sys user password
OLAP_USER	OLAP schema user name
OLAP_PASSWD	OLAP schema password
EBILL_SID	Your existing OLAP SID (instance name), to be used for the new EBILL database.
OLAP_SYS_PASSWD	OLAP sys user password
L_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_EDX_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_APP_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_APP_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.
L_DB_LOAD_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: `/home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data`.

Property	Value
L_DB_LOAD_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_LOAD_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_FS_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_FS_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_FS_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_STG_DATA_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
L_DB_STG_INDX_TB_FILE_LOC	The folder location where the installation script creates the L_DB_STG_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
LARGE_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
LARGE_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the LARGE_DB_EDX_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.

Property	Value
MEDIUM_DB_EDX_DATA_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_DATA_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.
MEDIUM_DB_EDX_INDX_TB_FILE_LOC	The folder location where the installation script creates the MEDIUM_DB_EDX_INDX_TB_FILE_LOC OLTP schema tablespace. The installation script automatically adds the Oracle SID, schema name, and /data to the path, as in: /home/oracle/oradata/ORACLE_SID/SCHEMA_NAME/data.

4 Save and close the migrate_oltp_6.1_to_6.1.1.properties file.

5 Run the following command to start the migration:

```
ant -f migrate_oltp_6.1_to_6.1.1.xml
```

6 Select Option 1, Export OLTP schema.

7 Select Option 2, Create OLTP tablespaces and OLTP user.

8 Select Option 3, Import OLTP schema into OLAP instance.

9 Select Option 4, Migrate OLTP schema.

10 Select Option 5, Migrate OLTP data.

BUILD SUCCESSFUL displays on the console screen if the migration is successful.

11 Review all log files for possible errors.

Migrating Oracle Self-Service E-Billing Version 6.1 OLAP to Version 6.1.1

Follow these steps to migrate Oracle Self-Service E-Billing 6.1 OLAP to 6.1.1 for both UNIX and Windows.

This task is a step in "Process of Migrating Oracle Self-Service E-Billing from 6.1 to 6.1.1" on page 173.

To migrate Oracle Self-Service E-Billing 6.1 OLAP to Version 6.1.1

1 On UNIX, log in as the ORACLE user for migration activity.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/oracle/olap/migration/ebilling6.1_to_6.1.1
- **Windows.** *EDX_HOME*\db\oracle\olap\migration\ebilling6.1_to_6.1.1

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

3 Open the migrate_olap_6.1_to_6.1.1.properties file, and set the correct value for each property in this file.

Property	Value
OLAP_USER	OLAP schema user name
OLAP_PASSWD	OLAP schema password
EBILL_SID	Your existing OLAP SID (instance name), to be used for the new EBILL database.
OLTP_USER	OLTP schema user name
OLTP_PASSWD	OLTP schema password

4 Save and close the migrate_olap_6.1_to_6.1.1.properties file.

5 Run the following command to start the migration:

```
ant -f migrate_olap_6.1_to_6.1.1.xml
```

6 Select Option 1, Migrate OLAP Schema.

7 Select Option 2, Migrate OLAP data.

BUILD SUCCESSFUL displays on the console screen if migration is successful.

8 Review all log files for possible errors.

Migrating Batch Reports from Oracle Self-Service E-Billing Version 6.1 to 6.1.1

After you have successfully migrated from Oracle Self-Service E-Billing 6.1 to 6.1.1, you must copy the batch reports to the directory you want to use for reports in the Oracle Self-Service E-Billing version 6.1.1.

If you copy the reports to a different location than you used in version 6.1, then you must update the batch report file location in the OLTP database. Also, if you installed Oracle Self-Service E-Billing 6.1.1 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* directory or the *output**reportapp* directory on Windows), then 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 from 6.1 to 6.1.1" on page 173

To migrate your batch reports from Oracle Self-Service E-Billing 6.1 to version 6.1.1

1 On the application server, copy your existing batch reports to the following directory:

- **UNIX.** *EDX_HOME*\output\reportapp

- **Windows.** *EDX_HOME*/output/reportapp

In the directory, *EDX_HOME* is the directory where you installed Oracle Self-Service E-Billing 6.1.1.

2 If you copy your batch reports to a different file location in version 6.1.1 than you used in version 6.1, 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 6.1.1.
- *OLD_LOCATION* is the old location where you stored batch reports in Oracle Self-Service E-Billing 6.1.

For example:

```
SQL> update edx_rpt_batch_report set file_location = '/export/home/oracle/
eBilling611/output/reportapp' || substr(file_location,length('/export/home/
oracle/eBilling/output/reportapp')+1) where file_location is not null;

SQL>commit;
```

3 If you installed Oracle Self-Service E-Billing 6.1.1 in a directory other than the default *EDX_HOME* directory, or if you plan to use a customized batch report location (the default directory is *output/reportapp*, or the *output**reportapp* directory on Windows), then you must update the properties in the reporting.batch.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/com/edocs/common/reporting/
- **Windows.** *EDX_HOME*\xma\config\com\edocs\common\reporting\

Specify your installation root directory in the rootDir property, and specify the custom report subdirectory in the path property (report files are stored in the *rootDir/path* directory, or the *rootDir/path* directory on Windows):

```
<!-- 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>
```

7 Migrating to Oracle Self-Service E-Billing Version 6.1 from Other Products

This chapter describes how to migrate to Oracle Self-Service E-Billing Version 6.1 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 6.1.1

This topic describes the tasks necessary to migrate the Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing, version 6.1.1. 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

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 6.1 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 EBilling 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 108.

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 27

   c "Choosing a Database Encryption Method" on page 45

   d "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29

   e "Open the ebilling_oltp.properties file, located in the following directory:" on page 38

5 Follow these procedures to set up the ETL and load your data:

   a "Verifying ETL Module System Requirements" on page 121

   b "Creating the Oracle Workflow Manager" on page 122

   c "Process of Installing the Oracle Warehouse Builder Repository" on page 123

   d "Installing the ETL Module" on page 130

   e "Running the ETL Loader Job Using Sample Data" on page 133

   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.

6 "Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing 6.0.3" on page 183

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, then perform the following tasks to integrate flat-file functionality in Oracle Self-Service E-Billing:

   a "Repackaging the Flat-File Component" on page 185

   b "Loading Historical Bill Data Using ETL" on page 186

   c (Optional) If you want to display the historical bills using a format other than the view currently published, then 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 186

   e "Customizing the Media Retrieval Functionality" on page 189

   f "Customizing the Oracle Self-Service E-Billing User Interface to Render the Statement" on page 190

8 "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 161

9 "Process of Migrating Oracle Self-Service E-Billing from 6.0.4 to 6.1" on page 168

10 "Process of Migrating Oracle Self-Service E-Billing from 6.1 to 6.1.1" on page 173

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 previous steps outlined in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181 before migrating Oracle eStatement Manager 4.7 to Oracle Self-Service E-Billing 6.0.3, or the migration will fail.

This task is a step in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181.

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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*/db/ebilling/oracle/oltp/migration/eaSuite47Fix Pack 1_to_603
- **Windows.** *EDX_HOME*\db\ebilling\oracle\oltp\migration\eaSuite47Fix Pack 1_to_603

You must run the migration process on the Oracle Self-Service E-Billing server.

3 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.

4 Save and close the migrate_easuite47_to_603.properties file.

5 Set the Java environment variable (JRE 1.6 is required), as shown in the following examples. In the variable, *JDK150_11* is your JDK version.

- **UNIX.** Run the following commands:

```
export JAVA_HOME=/usr/local/bea10/jdk150_11
export  PATH=$JAVA_HOME/bin:$PATH
```

- **Windows.** Run the following commands:

```
set JAVA_HOME=d:\beak\jdk150_11
set PATH=%JAVA_HOME%\bin;%PATH%
```

6 Run the following command to start migration:

```
ant
```

7 Select Option 1 to install the migration-related objects.

8 Select Option 2 to prepare the Oracle Self-Service E-Billing 6.0.3 database.

9 Review all log files for possible errors. If there were no errors, then select Option 3, 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*.

Repackaging the Flat-File Component

This topic 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 6.1.1" on page 181.

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, located in the following directory, to a folder such as ear-eStatement:

- **UNIX.** *EDX_HOME*/J2EEApps/Weblogic
- **Windows.** *EDX_HOME*\J2EEApps\Weblogic

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, not including 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 "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181.

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 complies 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

As part of integrating flat-file functionality, you must configure the following media retrieval configuration files:

- mediaretrieval.xma.xml
- mediaretrieval.config.xma.xml

This task is a step in "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181.

Configuring the mediaretrieval.xma.xml File

Follow these steps to modify the necessary JNDI parameters in the mediaretrieval.xma.xml configuration file.

To modify the mediaretrieval.xma.xml configuration file

1 Open the mediaretrieval.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/modules/statement
- **Windows.** *EDX_HOME*\xma\config\modules\statement

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: weblogic.jndi.WLInitialContextFactory
java.naming.provider.url	Location of the registry when the registry is being used as the initial context. The format is protocol:// yourAppServerIP:Port. For Oracle WebLogic, the protocol must be t3. The port is your application server's 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>
```

Configuring the mediaretrieval.config.xma.xml File

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 Open the mediaretrieval.config.xma.xml file, located in the following directory:

- **UNIX.** *EDX_HOME*/xma/config/modules/statement
- **Windows.** *EDX_HOME*\xma\config\modules\statement

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 follows, then you must set the billerIdMap key as:

```
BS1"0000|B2B                |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 "Roadmap for Migrating Oracle eStatement Manager Version 4.7 to Oracle Self-Service E-Billing Version 6.1.1" on page 181.

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 located in the following directory:

- **UNIX.** *EDX_HOME*/J2EEApps/ebilling/weblogic/ebilling-web-1.0-SNAPSHOT.war/mediaretrieval/test
- **Windows.** *EDX_HOME*\J2EEApps\ebilling\weblogic\ebilling-web-1.0-SNAPSHOT.war\mediaretrieval\test

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 6.1.1" on page 181.

Examples of How to Get Account Summary Information

The following examples show the code to 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 to 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 6.1.1

To migrate from Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing version 6.1.1, perform the following steps or processes.

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 from the Extract Transform Loading (ETL) process.

   For information about using the ETL jobs to load data, see *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 108.

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 27
   c. "Choosing a Database Encryption Method" on page 45
   d. Choose the ant installation process appropriate for your implementation:
      - "Creating the Oracle Self-Service E-Billing Database Using Ant (Single Node)" on page 29
      - "Creating the Oracle Self-Service E-Billing Database Using the Automated Ant Target" on page 37

6. Follow these procedures to set up the ETL and load your data:

   a. "Verifying ETL Module System Requirements" on page 121
   b. "Creating the Oracle Workflow Manager" on page 122
   c. "Process of Installing the Oracle Warehouse Builder Repository" on page 123
   d. "Installing the ETL Module" on page 130

e 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 "Migrating Oracle Communications Billing Manager 5.1.1 QF3 to Oracle Self-Service E-Billing 6.0.3" on page 193

8 "Process of Migrating Oracle Self-Service E-Billing 6.0.3 to 6.0.4" on page 161

9 "Process of Migrating Oracle Self-Service E-Billing from 6.0.4 to 6.1" on page 168

10 "Process of Migrating Oracle Self-Service E-Billing from 6.1 to 6.1.1" on page 173

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 previous steps outlined in "Roadmap for Migrating the Oracle Communications Billing Manager 5.1.1 QF3 Database to Oracle Self-Service E-Billing Version 6.1.1" on page 192 before migrating Communications Billing Manager 5.1.1 to Oracle Self-Service E-Billing 6.0.3, or the migration will fail.

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 6.1.1" on page 192.

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.

2 Go to the following directory:

- **UNIX.** *EDX_HOME*\db\ebilling\oracle\oltp\migration\cbm511qf3_to_603
- **Windows.** *EDX_HOME*/db/ebilling/oracle/oltp/migration/cbm511qf3_to_603

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.

3 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

4 Save and close migrate_cbm_to_ebilling603.properties file.

5 Set the Java environment variable (JRE 1.5 is required), as shown in the following example. In the variable, *JDK150_11* is your JDK version.

- **UNIX.** Run the following commands:

```
export  JAVA_HOME=/usr/local/bea10/jdk150_11

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

- **WINDOWS**. Run the following commands:

```
set  JAVA_HOME=d:\bea\jdk150_11

set PATH=%JAVA_HOME%\bin;%PATH%
```

6 Run the following command to start migration:

```
ant
```

7 Select Option 1 to install the migration-related objects.

8 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. Select 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. Select Option 3 to return to the Main Menu.

NOTE: If you merge both Business and Consumer Editions into one schema, then the ant script checks for data conflicts; you must resolve any conflicts before continuing with the merge.

9 Select Option 4 to run the Java process and perform post-migration operations.

10 Review all log files for possible errors. If there were no errors, then select Option 5, Post Migration Cleanup.

11 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 6.1.1

To migrate from Oracle Communications Billing Analytics 5.1.1 Quick Fix 3 to Oracle Self-Service E-Billing version 6.1, 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 196

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 "Process of Migrating Oracle Self-Service E-Billing 6.0.2 to 6.1.1" on page 153

Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0.

This topic describes the process of migrating the Oracle Communications Billing Analytics 5.1.1 database to Oracle Self-Service E-Billing version 6.0.

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 6.1.1" on page 196

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 197

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 198

4 "Installing the Payment Database on an Existing Oracle Communications Billing Analytics 5.1.1 OLTP Database" on page 199

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 201 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 202

6 "Migrating the Oracle Communications Billing Analytics 5.1.1 OLAP Schema to Oracle Self-Service E-Billing 6.0 on UNIX" on page 204 or Migrating the Oracle Communications Billing Analytics 5.1.1 OLAP Schema to Oracle Self-Service E-Billing 6.0 on Windows on page 205

7 "Compiling the Schema After Migrating the OLTP and OLAP Oracle Communications Billing Analytics Databases" on page 206

8 "Populating the Role Authorization Database Table" on page 207

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 196.

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. In this command, *OLTP_Schema* is the name of the OLTP schema.

```
SQL> GRANT CREATE DATABASE LINK TO OLTP_Schema;
```

b Go to the following directory:

- **UNIX.** *EDX_HOME*\db\ebilling\oracle

- **Windows.** *EDX_HOME*/db/ebilling/oracle

In the path, *EDX_HOME* is the location where Oracle Self-Service E-Billing is installed.

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 "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 196.

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.
2. Go to the following directory. In the path, *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

3 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.

4 Also in the migrate.properties file, edit the following statements to specify the appropriate file location:

- **Large Data Tablespace.** `LARGE_DB_EDX_DATA_TB_FILE_LOC=/export/home/oracle/oradata`
- **Large Index Tablespace.** `LARGE_DB_EDX_INDX_TB_FILE_LOC=/export/home/oracle/oradata`
- **Medium Data Tablespace.** `MEDIUM_DB_EDX_DATA_TB_FILE_LOC=/export/home/oracle/oradata`
- **Medium Index Tablespace.** `MEDIUM_DB_EDX_INDX_TB_FILE_LOC=/export/home/oracle/oradata`

5 Go to your eStatement database home directory, for example:

- **UNIX.** *EDX_HOME*`/db/eStatement/oracle`
- **Windows.** *EDX_HOME*`\db\eStatement\oracle`

6 Run the following Ant target:

```
ant migrate
```

The script 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)*

7 Enter Y if you have a backup, or N if you do not.

8 Enter Y to continue the migration process.

9 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 a Payment database on your existing Oracle Communications Billing Analytics 5.1.1 OLTP database before migrating OLTP.

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 196.

To install the Payment database on an existing Oracle Communications Billing Analytics 5.1.1 OLTP database

1. Open the payusr.properties file, located in the following directory:
   - **UNIX.** *EDX_HOME*/db/ePayment/oracle
   - **Windows.** *EDX_HOME*\db\epayment\oracle

   In the path, *EDX_HOME* is the location where Oracle Communications Billing Analytics is installed.
2. In the payusr.properties file, specify 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.
3. On UNIX, log in as the ORACLE user for migration activity.
4. 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
5. If you have not configured the Apache Ant environment, then do so now.
   - **UNIX.** Run the following commands, where *JDK150_11* is your specific JDK version:

     ```
     export ANT_HOME=/opt/apache-ant-1.6.5

     export PATH=$ANT_HOME/bin:$PATH

     export JAVA_HOME=$WEBLOGIC_HOME/JDK150_11

     export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$PATH
     ```
   - **Windows.** Run the following commands, where *JDK150_11* is your specific JDK version:

     ```
     set ANT_HOME=C:\apache-ant-1.6.5

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

     set JAVA_HOME= %WEBLOGIC_HOME%\JDK150_11
     ```
6. Enter Ant to run the build script.

   By default, the Ant command runs the build.xml file in the current directory.
7. From the top level Main Menu select Option 1, Install Application Database I.
8. Enter Ant again, and from the top level Main Menu select Option 2, Install Application Database II.
9. Enter 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 "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 196.

To migrate the Oracle Communications Billing Analytics 5.1.1 OLTP database to Oracle Self-Service E-Billing 6.0 on UNIX

1. On UNIX, log in as the ORACLE user for migration activity.
2. 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.
3. Verify that the following files exist:
   - migrate_oltp_511_to_6.0.sh
   - migrate_oltp_511_to_6.0.sql
4. Run the following commands at the shell prompt:

   ```
   chmod 777 migrate_oltp_511_to_6.0.sh

   ./migrate_oltp_511_to_6.0.sh
   ```

5. 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

6. 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 "Process of Migrating the Oracle Communications Billing Analytics 5.1.1 Database to Oracle Self-Service E-Billing 6.0." on page 196.

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. In the path, *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

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 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 196.

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. In the command, *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 196.

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. In the command, *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 196.

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, which causes 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 196.

Index

O