Oracle® Revenue Management and Billing for Financial Services

Version 2.2.4.0

Upgrade Guide

Revision 2.0

E26898-01

November, 2011

Oracle Revenue Management and Billing Upgrade Guide

E26898-01

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

Preface

Purpose

This document will help you to understand the pre-requisites for upgrading the Oracle Revenue Management and Billing (ORMB) application from the current release to the new release or service pack. It contains procedures that you should know for upgrading the Oracle RMB Framework, ORMB application, and its database.

Refer to the Upgrade Path guide before proceeding with the upgrade process.

Intended Audience

The following are the intended audience of this document:

- Users
- Consulting staff
- Administrators

Organization of the Document

The information in this document is organized into the following chapters and appendices:

Chapter No.	Chapter Name	Chapter Description
Chapter 1	Planning for the Upgrade	Provides an overview and prerequisites for upgrading the ORMB application.
Chapter 2	Upgrading the Oracle RMB Framework	Provides the steps to upgrade the Oracle RMB Framework through single fixes and service pack.
Chapter 3	Upgrading the Oracle Revenue Management and Billing Application	Provides the steps to upgrade the ORMB application using the service pack.
Chapter 4	Upgrading the Oracle Revenue Management and Billing Database	Provides the steps to upgrade the database using the blueprints defined for the service packs.
Chapter 5	Additional Tasks	Lists and describes the additional tasks that need to be completed after upgrading the ORMB application.
Appendix A	Known Issues	Lists the known issues in the current release of the ORMB application.
Appendix B	Third Party Software Upgrade	List the steps to upgrade various third party software.

Conventions

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

Related Documents

The following documents are related to this document:

- *Oracle Revenue Management and Billing Release Notes* - Provides information about release notes and known issues.
- *Oracle Revenue Management and Billing Installation Guide* – Provides steps to install the application for the first time on a server.
- *Oracle Revenue Management and Billing Database Administrator's Guide* – Provides steps to install the database for the first time on a server.
- *Oracle Revenue Management and Billing Upgrade Path Guide* – Provides a list of upgrade paths which apply to the current product releases.

Contents

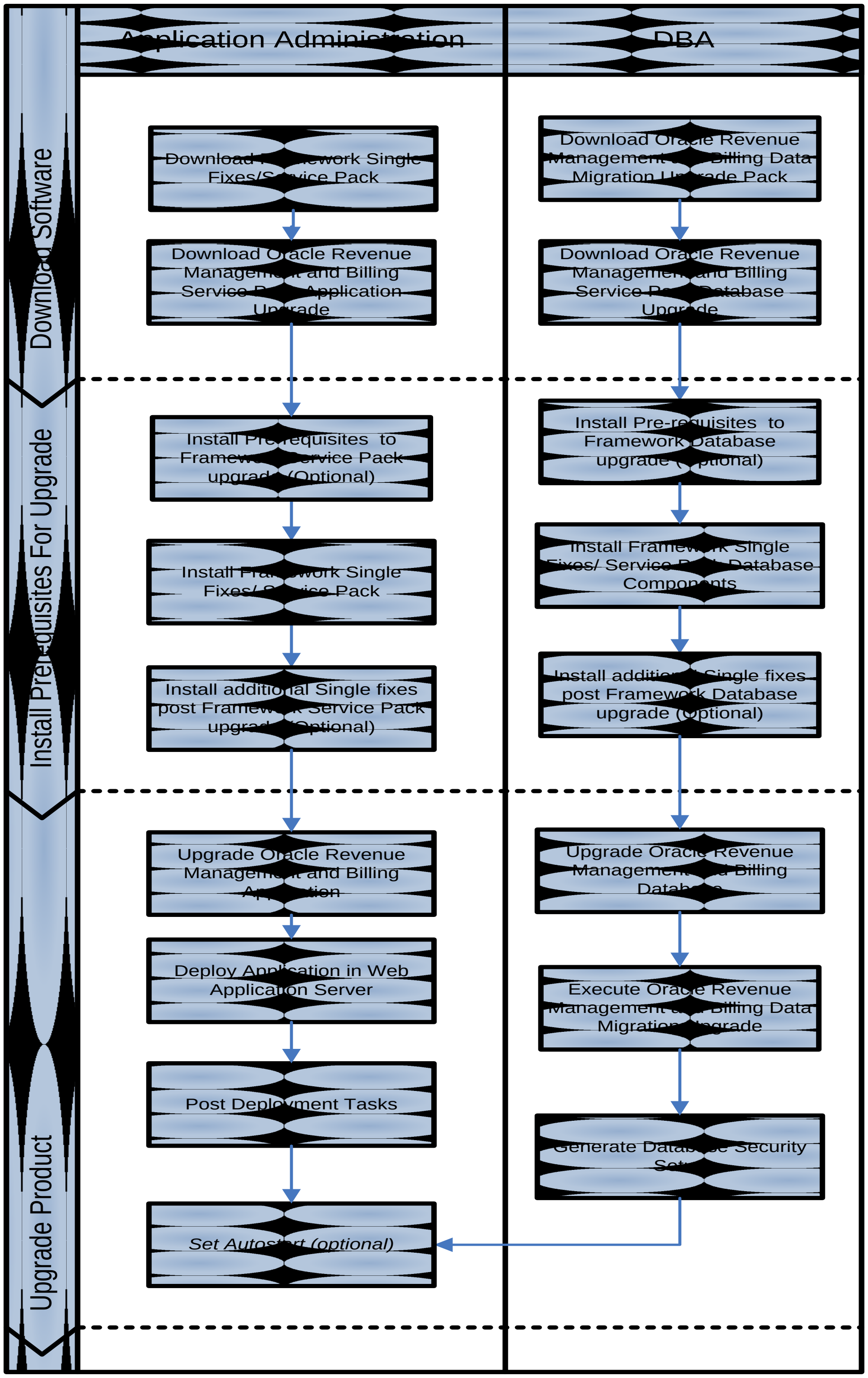
Chapter 1: Planning for the Upgrade

This chapter provides information for planning the upgrade of Oracle Revenue Management and Billing application, including:

- Upgrade Overview
- Oracle RMB Framework Upgrade Pre-requisites
- Oracle Revenue Management and Billing Upgrade Requirements

Upgrade Overview

The following diagram provides an overview of the steps that need to be taken to upgrade the Oracle Revenue Management and Billing application:


Application Administration
DBA
Download Software
Install Prerequisites For Upgrade
Upgrade Product
Download Framework Single Fixes/Service Pack
Download Oracle Revenue Management and Billing Data Migration Upgrade Pack
Download Oracle Revenue Management and Billing Service Pack Application Upgrade
Download Oracle Revenue Management and Billing Service Pack Database Upgrade
Install Prerequisites to Framework Service Pack upgrade (Optional)
Install Pre-requisites to Framework Database upgrade (Optional)
Install Framework Single Fixes/ Service Pack
Install Framework Single Fixes/ Service Pack Database Components
Install additional Single fixes post Framework Service Pack upgrade (Optional)
Install additional Single fixes post Framework Database upgrade (Optional)
Upgrade Oracle Revenue Management and Billing Application
Upgrade Oracle Revenue Management and Billing Database
Deploy Application in Web Application Server
Execute Oracle Revenue Management and Billing Data Migration Upgrade
Post Deployment Tasks
Generate Database Security Sets
Set Autostart (optional)

Oracle RMB Framework Upgrade Pre-requisites

Before upgrading the Oracle Revenue Management and Billing application to the Service Pack or major/minor release application version, the Oracle RMB Framework needs to be upgraded to the appropriate version for which it is certified for.

Note: Please refer to the Oracle Revenue Management and Billing Upgrade Path Guide for the list of certified Framework and application service packs and dependencies.

Chapter 2: Upgrading the Oracle RMB Framework

This section gives the steps for installing and upgrading the Oracle RMB Framework Service Pack(s). It includes the following

- Preinstallation Tasks
- Applying Framework Service Pack on Database
- Applying Framework Service Pack on Application Server

Preinstallation Tasks

This process replaces any previously delivered and installed version of the Oracle RMB Framework Server. Before proceeding with installing the Service Pack, ensure that you have a back up of the application and the database.

Download Framework Service Pack

You can download the Framework Service Pack from Metalink by searching for the appropriate patch number.

Copying and Decompressing Install Media

The installation file is delivered in zip format for UNIX, Windows and Linux platforms.

The Oracle RMB Framework Service Pack is delivered as a separate installation package, common for all platforms.

Download the installation package for your operating system and proceed with the following instructions.

1. Log in to the host server as the Oracle Revenue Management and Billing administrator userid (default cissys).
2. Create a <TEMPDIR> directory on the server, which is independent of any current or other working Oracle Revenue Management and Billing application environment (preferably on the C or D drive on windows or the root folder in UNIX/Linux).
3. Copy the zip file in the delivered package to a <TEMPDIR> on your Windows/UNIX/Linux box. If you are using FTP to transfer this file, remember to use the BINARY option for the FTP transfer.
4. Unzip the file using the command:

   ```
   unzip <filename>.zip
   ```

5. This will extract a jar file which can be decompressed using the following command:

   ```
   jar -xvf <filename>.jar
   ```

6. This will create a new folder which will have the service pack contents. It will have a database component and an application component.

Preparing for the Installation

1. Log on to the application server as administrator
2. Initialize the Framework environment that you want to install the product into.

   UNIX:

   ```
   $SPLEBASE/bin/splenviron.sh -e $SPLENVIRON
   ```

   Windows:

   ```
   $SPLEBASE\bin\splenviron.cmd -e %SPLENVIRON%
   ```

   Linux:

   ```
   $SPLEBASE/bin/splenviron.sh -e $SPLENVIRON
   ```

Note: Substitute $SPLEBASE with appropriate values for your installation. You will need to restart the server_Name before you attempt to start the application on the server.

Applying Framework Service Pack on Database

This installation software will install a single fix Service Pack on top of an existing Oracle database.

To apply the single fix, you must setup a Microsoft Windows or NT workstation desktop with Oracle Client (10.2.0.3 or later) installed. In addition, you must be able to connect to the database from the workstation.

Installation

1. Select your database platform from the database folder.
2. Unzip the file.
3. Execute the CDXPatch.exe utility. The utility will ask you for the type of database you will be using. Enter the target database type (O/M/D) [O]: `O`, the default for Oracle and M for MSQL
4. The utility will prompt you for the value of the following parameters after choosing your database type:

   - Enter the name of the user that owns CISADM objects: `CISADM`
   - Enter the password for the user (in silent mode): `CISADM`
   - Enter the name of the target database: `<Database Name>`

   CDXPatch.exe can be executed by selecting it from Windows explorer or in command line mode from a DOS window in the folder in which the program exists. Use option "-h" to see the help for the same.
5. The utility will run and apply the SQLs included in each single fix that has been included in the Service Pack but has not been applied to your database. For each run the utility creates a new working directory with name of format “XXXXXnnn”, where “XXXXXX” denotes the database name and “nnn” a running number.
6. Also, a separate log file is created in this working directory for each single fix applied during each run.

Note: After the installation of the database portion of the service pack, you are required to run the OraGenSec utility. See Security Configuration.

Security Configuration

1. Execute the OraGenSec.exe.
2. The script will prompt for parameter values.

   - Enter the Oracle user that owns the schema (e.g. CISADM): `CISADM`
   - Enter the password for the CISADM user: `CISADM`
   - Enter the name of the Oracle Database: `<Database Name>`
   - Enter a comma-separated list of Oracle users in which synonyms need to be created (e.g. cisuser, cisread): `CISUSER,CISREAD`
3. The utility configures the security for the CISADM schema objects.

Applying Framework Service Pack on Application Server

This installation software will install a single fix Service Pack on top of the existing FW V2.2.0.0 application server.

The file SERVICEPACK.txt contains a list of single fixes that are included in this Service Pack.

The README.txt file contains the description for each fix included in the Service Pack/Roll Up

The Single Fix group installation process is used to install all the fixes to the target environment.

When the installation is completed check carefully the logs in each single fix directory inside a Service Pack directory. A utility script is included in this package that will scan each single fix installation log file for errors.

Installation

1. Logon as Oracle RMB administrator. Logon as cissys (on UNIX) or as a user with Administrator privileges (on Windows)

2. Configure application server and third-party software.

3. Change to the <TEMPDIR>/FW-V220-SPx (UNIX) or <TEMPDIR>\FW-V220-SPx (Windows).

4. Run the installation script.

   UNIX: `ksh ./installSPack.sh`

   Windows: `installSPack.cmd`

5. Run the error checker script which will scan all single fix installation log files and report any errors if found.

   UNIX/Windows: `perl SPErrorChecker.plx`

6. IMPORTANT! The following commands need to be executed when the install completes:

   ```
   UNIX:

   splenviron.sh -e <ENV_NAME>

   configureEnv.sh

   Type "P" and <ENTER> (you don't need to change anything)

   splenviron.sh -e <ENV_NAME>

   initialSetup.sh
   ```

   Windows:

   ```
   splenviron.cmd -e <ENV_NAME>

   configureEnv.cmd

   Type "P" and <ENTER> (you don't need to change anything)

   splenviron.cmd -e <ENV_NAME>

   initialSetup.cmd
   ```

Service Pack Install Verification

1. The installation script completes the rest of the steps. A log file is created in the installation directory for each environment on which you installed the fix. If for any reason the script fails, the errors and error diagnosis steps are reported on the screen and in the log file.

2. The installation utility does not allow the installation of the same single fix twice, since the historical order of the single fix installation is essential for correct application functionality. You can see the list of single fixes installed on your environment in the order of their installation in the file <SPLEBASE>/etc/installed_fixes.txt.

Chapter 3: Upgrading the Oracle Revenue Management and Billing Application

This chapter describes the procedure for upgrading the Oracle Revenue Management and Billing application from the existing version to the new product service pack version. This service pack is essentially an upgrade only and does not include setting up a new application instance. This section includes:

- Preinstallation Tasks
- Installing the Application Upgrade
- Additional Tasks Required for WebSphere
- Operating the Application

Preinstallation Tasks

This process replaces any previously delivered and installed version of the Oracle Revenue Management and Billing Application Server. As a **pre-requisite**, you must upgrade the framework to the specified Oracle RMB Framework Service Pack before upgrading the Oracle Revenue Management and Billing application. The instructions for upgrading the framework are specified in the **Chapter 2: Upgrading the Oracle RMB Framework**.

Before going ahead with installing Oracle Revenue Management and Billing Application Service Pack, ensure that you have a back up of the application and the database. This will also ensure that the customizations are backed up.

Note: Upgrading Oracle Revenue Management and Billing application will not overwrite any Customer Modifications done during any customer implementations.

Download Application Service Pack

You can download the Application Service Pack from Metalink by searching for the appropriate patch number

To proceed with the Oracle Revenue Management and Billing upgrade, you need to be connected to the target framework application environment. See the detailed installation instructions in the following section.

You must initialize the Framework environment along with the required Patch Set prior to proceeding with Oracle Revenue Management and Billing Application product installation. For detailed instructions see **Preparing for the Installation** section below.

Note: For General Availability (GA) releases, we don't have application service packs.

Copying and Decompressing Install Media

Oracle Revenue Management and Billing Application Service Pack or GA Release is delivered in a separate installation package for each supported operating system. The installation file is in zip format for UNIX, Windows and Linux platforms.

After downloading the Service Pack or the GA Release for your operating system, proceed with the following instructions.

1. Log in to the host server as the Oracle Revenue Management and Billing administrator userid (default cissys).
2. Create a <TEMPDIR> directory on the server, which is independent of any current or other working Oracle Revenue Management and Billing application environment (preferably on the C or D drive on windows or the root folder in UNIX/Linux).
3. Copy the zip file in the delivered package to a <TEMPDIR> on your Windows/UNIX/Linux box. If you are using FTP to transfer this file, remember to use the BINARY option for the FTP transfer.
4. Decompress the file:

```
cd <TEMPDIR>

unzip <filename>.zip
```

5. This will create a new folder in <TEMPDIR> and extract the install files and the Readme.txt.

On UNIX/Linux and Windows platforms, for the RMB build, a new sub-directory is created. The contents of the installation directory are identical for both platforms. The directory contains the install software for the application product.

6. For UNIX and Linux environments, edit the cobdir.txt file to include the path where you have installed Micro Focus 5.1.

Preparing for the Installation

1. Log on as Administrator (default cissys).
2. Initialize the Framework environment that you want to install the product into.

   UNIX:

   ```
   $SPLEBASE/bin/splenviron.sh -e $SPLENVIRON
   ```

   Windows:

   ```
   $SPLEBASE\bin\splenviron.cmd -e %SPLENVIRON%
   ```

   Linux:

   ```
   $SPLEBASE/bin/splenviron.sh -e $SPLENVIRON
   ```

3. Stop the environment if running.

   UNIX:

   ```
   $SPLEBASE/bin/spl.sh stop
   ```

   Windows:

   ```
   $SPLEBASE%\bin\spl.cmd stop
   ```

   Linux:

   ```
   $SPLEBASE/bin/spl.sh stop
   ```

4. On Windows, UNIX and Linux systems, set the following on command prompt:

   ```
   set ANT_OPTS= -Xms512m -Xmx1024m -XX:PermSize=256M
   ```

Installing the Application Upgrade

1. Change to the <TEMPDIR> Directory where the application was extracted.
2. Execute the script:

   UNIX:

   ```
   ksh ./install.sh
   ```

   Windows:

   ```
   install.cmd
   ```

   Linux:

   ```
   ksh ./install.sh
   ```

   You will be prompted to answer the following question:

   ```
   Do you wish to proceed with the installation? Y/N: Y
   ```

   If you are upgrading an existing environment, you will be prompted to answer the following question:

   ```
   Product CCB is already installed in the environment $SPLENVIRON. Do you want to
   reinstall it? [Y/N]

   Enter Choice: Y
   ```

Note: On UNIX, ensure that you have the proper execute permission on install.sh.

3. Follow the messages and instructions that are produced by the install utility. Please note that some of the steps (such as the genAppViewer) will take some time to complete.
4. If the install execution was not stopped due to errors and you did not interrupt the execution, you have finished the upgrade of the Oracle Revenue Management and Billing Application product.
5. Once the install utility has finished successfully, it will again ask for user inputs
   Do you wish to start the environment? Y/N: N
   Before starting the application, you will need to execute the following steps:
   UNIX:

   ```
   ./splenviron.sh -e <ENV_NAME>
   ./configureEnv.sh
   ```

   Type **P** and <ENTER> (you don't need to change anything)

   ```
   ./splenviron.sh -e <ENV_NAME>
   ./initialSetup.sh
   ```

   Windows:

   ```
   splenviron.cmd -e <ENV_NAME>
   configureEnv.cmd -e <ENV_NAME>
   ```

   Type **P** and <ENTER> (you don't need to change anything)

   ```
   splenviron.cmd -e <ENV_NAME>
   ```

```
initialSetup.cmd
```

Linux:

```
./splenviron.sh -e <ENV_NAME>
./configureEnv.sh
```

Type **P** and <ENTER> (you don't need to change anything)

```
./splenviron.sh -e <ENV_NAME>
./initialSetup.sh
```

Note:

Please ensure that the cobdir path points to the location where Micro Focus 5.1 is installed on the system.

At the end of WebSphere environment installation, you will be prompted whether you want to deploy the application on the WebSphere server. You can either reply N and deploy the application on the WebSphere server manually using WebSphere admin console or reply Y and use WebSphere deployment script (i.e. genwasdeploy.sh) provided as a part of the installation. Steps to install the application ear files to WebSphere using the console are mentioned in the Additional Tasks Required for WebSphere section.

6. Start up the environment.
   The final step of the installation process is the environment startup. The install utility executes the command spl.sh start (for UNIX or Linux) or spl.cmd start (for Windows) to start up the environment. You may start the environment by this command any time from the following location:

   ```
   (%SPLEBASE%\bin) spl.cmd start or ./spl.sh start
   ```

   Follow the messages on the screen and check the logs in `$SPLSYSTEMLOGS` (%SPLSYSTEMLOGS% on Windows) directory to ensure that the environment was started successfully. If the startup failed, identify the problem by reviewing the logs, and start up the environment manually while you are connected to the new environment in your online session.

Note: Since the WebSphere server usually runs under root userid, it needs to be restarted with the new environment variable settings after Oracle Revenue Management and Billing environment installation and deployment into WebSphere. To do this switch to root userid, setup $WAS_HOME environment variable (and the rest of environment variables as described in Boldparatext: Boldparatext), initialize the newly installed environment by executing the command: $SPLEBASE/bin/splenviron.sh -e $SPLENVIRON and after that shutdown and startup WebSphere server using WebSphere commands, e.g. $WAS_HOME/bin/startServer.sh server1. After the initial server restart you may start the application by the environment startup/shutdown commands or through WebSphere Admin console.

Additional Tasks Required for WebSphere

Configure the Environment Entry

Under **Application servers, <server_Name>, Java and Process Management, Process Definition, Environment Entries,** Add the entry:

Name: LIBPATH

Value: <$SPLEBASE>/runtime

Note: Substitute $SPLEBASE with appropriate values for your installation. You will need to restart the server_Name before you attempt to start the application on the server.

Note: For deploying ORMB application to WebSphere, it is recommended that you DO NOT deploy into server1. We reserve this exclusively for admin console.

Application Deployment

You will need to deploy the application after running the application installation utility.

You can manually deploy the ear files, which the application installation utility has created.

Deployment via the Admin Console

Follow these steps to deploy the application using the Admin Console. If the SPLService.ear and SPLWeb.ear files are already installed to the WebSphere server, then they need to be uninstalled before proceeding with the following steps.

Deployment Order

The application needs to be deployed in the following order:

1. SPLService.ear
2. SPLWeb.ear

Note: The SPLService.ear should be successfully deployed before deploying SPLWeb.ear

Deploying SPLService.ear

1. Stop the server instance if running and log into the WebSphere admin console.
2. Select the ear file to deploy.
3. Select **Applications, Install New Application.**
4. Select **Remote file system.**
5. Browse to the SPLService.ear or enter the full path to the file. The ear files can be found under $SPLEBASE/splapp/applications.
6. Click **Next**.
7. On “Select Installation Options” screen, no actions are required. Click on **Next**.

8. Assign the module to the WebSphere server instance from the “Map Modules to Servers” screen.
9. Select the server from *Clusters and Servers*.
10. Check the Select check-box against the module ServiceBean. Click **Apply** and **Next**.
11. On “Provide JNDI Names” screen, specify the JNDI name.

    The JNDI name is spl-<*server name*>/servicebean

    For example, *spl-server6/servicebean*. The JNDI name can be found in the spl.properties in SPLApp/web-inf/classes/ spl.properites.
12. Review the summary page.
13. Review the installation options.
14. Click **Finish** - the application will then deploy. The deployment process takes about 5 minutes.
15. Click **Save** to save the changes to master configuration. The save process can take about 20 minutes.

Deploying SPLWeb.ear

1. Select the ear file to deploy.

   Select **Applications, Install New Application**.

   Select **Remote file system.** Browse to the SPLWeb.ear or enter the full path.

   Select Show **me all installation options and parameters**. Click on Next.
2. On “Select Installation Options” screen, no actions are required.

   Select Precompile JavaServer Pages files.

   Click on **Next**.
3. “Preparing for the application installation” screen, no actions are required. Click on Next
4. On “Provide options to compile JSP’s” screen, enter 15 in JDK Source Level field for all the URIs
5. On “Provide JSP reloading options for Web modules” screen, no actions are required. Click on Next
6. On “Map shared libraries” screen, no actions are required. Click on Next
7. On “Initialize parameters for servlets” screen, no actions are required. Click on Next
8. Assign the module to the WebSphere server instance from “Map Modules to Servers” screen.

   When deploying an application from the console make sure you select the correct server, check all Select check boxes. Click **Apply** and click **Next**.
9. On “Map virtual hosts for Web modules" no actions are required. Click on **Next**.
10. On “Map context roots for Web modules” screen, no actions are required. Click on **Next**
11. On “Map environment entries for Web modules” screen, no actions are required. Click on **Next**
12. On “Map security roles to users or groups” screen, no actions are required. Click on **Next**
13. Review the summary page.

14. Click **Finish** - the application will then deploy. The deployment process above can take 5 minutes or more
15. Click **Save** to save the changes to the master configuration. The save process can take 20 minutes.

Configure the Applications

Following steps need to be done to configure the SPLWeb and SPLService applications.

Configure SPLService.ear

1. Navigate to Applications --> Enterprise Applications --> SPLService-server2 --> Manage Modules --> ServiceBean
2. Enter Starting weight as 1, click OK and Save to Master Configuration.

Configure SPLWeb.ear

1. Navigate to Applications --> Enterprise Applications --> SPLWeb-server2 --> Startup behavior
2. Set Startup order: **2**
3. Click **OK** and **Save** to Master Configuration.
4. Navigate to Applications --> Enterprise Applications --> SPLWeb-server2 --> Class loading and update detection
5. Set Polling interval: **0**
6. Set Class loader order: Classes loaded with application class loader first
7. Click OK and Save to Master Configuration.
8. Navigate to Applications --> Enterprise Applications --> SPLWeb-server2 --> Manage Modules
9. Select SPLApp.war
10. Set Starting weight: **10000**
11. Set Class loader order: **Classes loaded with application class loader first**
12. Click **OK**
13. Repeat above for the other 3 war files.
14. **Save** to Master Configuration.

Configure Application Security

After using the supplied script to deploy the application to WebSphere you will need to configure each application's security before starting the application.

Using the WebSphere administration console select **Applications, Enterprise Applications, SPLService-<server name>** (for example, SPLService-server2), **Security role to user/group mapping**.

For role cisusers:

1. Check **All Authenticated.**
2. Check **Select** and click **Look up users**:
3. Search for SYSUSER and add to the **Selected users** list.

4. Click **OK**
5. Click **OK**
6. **Save** to Master Configuration.

Note: Be sure to configure security for both SPLService and the SPLWeb applications.

Restart the WebSphere Server

It is recommended to stop and then restart the WebSphere server.

If the application is deployed in server1 you can use the admin console to stop and start the server. If the application is deployed in another server you will need to use the scripts that are supplied with WebSphere (stopServer.sh, startServer.sh).

Note: WebSphere admin console runs under server1.

Application URL

The Web link to the WebSphere application will be:

http://<hostname>:<WebSphere Port No>/spl-<server name>/cis.jsp

For example: http://sf-aixapp-02:9081/spl-server2/cis.jsp

Operating the Application

At this point your installation and custom integration process is complete.

Be sure to refer to the Oracle Revenue Management and Billing User Guides for more information on further configuring and operating the system.

Chapter 4: Upgrading the Oracle Revenue Management and Billing Database

This section describes the process of upgrading an existing Oracle Revenue Management and Billing Database with the latest ORMB version which covers service pack and future releases.

It is a pre-requisite that the database has already been created and Oracle Revenue Management and Billing product has been installed on the database. In addition, the Oracle RMB Framework Service Pack should be installed on the database. Steps for this are mentioned in the Chapter 2: Upgrading the Oracle RMB Framework.

This section describes the process for installing the Oracle Revenue Management and Billing Service Pack Upgrade on the database. It includes:

- Preinstallation Tasks
- Database Upgrade

Preinstallation Tasks

This process will overwrite the database schema and the meta-data present in the database. The transactional data will not be affected.

Decompressing Install Media

Unzip Windows-Oracle-Database.zip to your desktop. The contents of this zip include sub folders which are referred to in the Installation process.

It is advised that you take a back up of the database before going ahead with the database upgrade.

Note: This Upgrade Service Pack will only upgrade the Oracle Revenue Management and Billing Application and not the Framework. For setting up a new database please refer to the *ORMB Installation Guide*.

Database Upgrade

Database Upgrade Tasks

The files for the Oracle Revenue Management and Billing Database Upgrade are located in the Upgrade/Install-Upgrade folder. The folder contains the scripts and utilities that you will run in order to complete the installation process. This upgrade process is similar for service packs upgrades and major/minor release upgrades.

The process prompts for the names of three database users during this process:

- A user that will own the application schema (for example, CISADM).
- A user that has select/update/insert/delete (read-write) privileges to the objects in the application schema. The application will access the database as this user. (for example, CISUSER).
- A user with read-only privileges to the objects in the application schema. (for example, CISREAD).

Perform the following steps:

1. Run CDXDBI.exe. The utility prompts you to enter values for the following parameters:
    - The name of the target database.
    - The password for the SYSTEM user account in the database (in silent mode).
    - The name of the owner of the Database Schema.
    - The password for the user (in silent mode).
    - The Oracle user with read-write privileges to the Database Schema
    - The Oracle user with read-only privileges to the Database Schema
2. The utility at this point is ready to perform the initial install and prompts you for permission to start the process.
3. If you chose to continue, CDXDBI first checks for the existence of each of the users specified and prompts for their password, default tablespace, and temporary tablespace, if they do not exist.
4. After setting up the roles and users, the utility continues upgrading schema and system data definitions. If an error occurs while executing an SQL or another utility, it logs and displays the error message and allows you to re-execute the current step.

Security Configuration

Once the database upgrade is complete, you need to configure the security for the database. The configuration utility and scripts are located in the Security folder.

Execute the OraGenSec.exe utility.

The script will prompt for parameter values:

- Enter the Oracle user that owns the schema (e.g. CISADM): **cisadm**
- Enter the password for the CISADM user: **cisadm**
- Enter the name of the Oracle Database: **<database name>**
- Enter a comma-separated list of Oracle users in which synonyms need to be created (e.g. cisuser, cisread): **cisuser, cisread**
- Select the following options:
  ```
  (A/a): Generate security for All objects in the Database?
  (O/o): Generate security for specific Objects inputted in this terminal?
  (F/f): Generate security for specific objects generated from an input File?
  ```

Choose the relevant option and continue. The utility configures security for the CISADM schema objects.

Chapter 5: Additional Tasks

This section describes the additional tasks that should be completed after upgrading Oracle Revenue Management and Billing including:

- Generating the Application Viewer
- Batch Daemon Submission Options
- Building Javadocs Indexes

Generating the Application Viewer

The Application Viewer that is delivered with the system needs to be regenerated after upgrading the application. This section details the steps necessary to generate the application viewer

1. Initialize a command shell.

   UNIX: `ksh $SPLEBASE/bin/splenviron.sh`

   Windows: `%SPLEBASE%\bin\splenviron.cmd`

   Linux: `ksh $SPLEBASE/bin/splenviron.sh`

2. Set ANT_OPTS = -Xms512m –Xmx1024m –XX:PermSize=256m

3. Execute the script to generate all information

4. Execute the following command for your operating system

   UNIX: `ksh $SPLEBASE/bin/genappvieweritems.sh`

   Windows: `%SPLEBASE%\bin\genappvieweritems.cmd`

   Linux: `ksh $SPLEBASE/bin/genappvieweritems.sh`

5. Redeploy your application

After you have performed the above steps you will need to redeploy the application into the web application server.

You may use supplied scripts genwasdeploy.sh for WebSphere or use a rapid deployment method supplied with your particular web application server.

Batch Daemon Submission Options

The Batch Jobs need to be re-submitted after upgrading the application. This section details the steps to submit the batch jobs again.

Batch Submission using the Online JVM

Note: The functionality provided in previous releases with the scripts submitbatch.plx and cdxcronbatch.sh is no longer supported. These scripts have been deprecated in favor of the following multithreaded java batch daemon.

This step is necessary only if your organization uses either Oracle Enterprise Taxation Management 's batch scheduler or if you wish to submit ad-hoc batch jobs from the online application. This option is a good choice for low-volume, low activity environments.

Before jobs can be submitted from the on-line Batch Job Submission page, two batch components need to have been started: 1) a batch "worker" and 2) the batch-scheduling daemon. A "worker" node is responsible for running the batch jobs that are requested by the daemon.

The following describes how to start a worker and the daemon to allow for job submissions from the on-line system.

A batch worker node can be started as part of the application server, or as a standalone process.

Starting the batch worker

A batch worker node can be started as part of the application server, or as a standalone process.

When using the Online JVM option you should have the following settings when you run configureEnv.sh /cmd:

```
Batch Server Enabled:      true
Batch Threads Number:      5
Batch Scheduler Daemon:    true
```

Batch worker as part of the application server

For low-volume, low-activity environments (e.g. development, testing), this is a good choice because of the smaller resource footprint. To start it like this, simply specify the following application server properties:

```
com.splwg.grid.online.enabled=true
com.splwg.grid.distThreadPool.threads.DEFAULT=5
```

The first property allows the application server JVM to run batch jobs. See Thread Pool below for a description of the second property.

Standalone batch worker

To start a worker node as a standalone process, a JVM is started with the following properties:

```
com.splwg.batch.submitter=false
com.splwg.grid.distThreadPool.threads DEFAULT=5
```

The first property specifies that the JVM being started is a "worker" JVM (i.e. it is not a "submitter" JVM). See Thread Pool below for a description of the second property.

Thread Pool

The distributed thread pool property in the examples above assign the worker to thread pool "DEFAULT" and permits it to run 5 threads concurrently. Unless explicitly overridden at time of submission, the "DEFAULT" thread pool is where the online submissions automatically get directed to, so this should in most cases not be changed; if it is set to anything else, the online submissions may not be picked up. The number of threads, specified as "5" above, can be adjusted.

Starting the batch-scheduling daemon

The batch-scheduling daemon can be started as part of the application server, or in a standalone batch worker node. In either case, add the following property to enable the JVM to also be a scheduler:

```
com.splwg.batch.scheduler.daemon=true
```

Batch Submission using Thread Pool Worker

This step is necessary if your organization uses either Oracle Revenue Management and Billing's batch scheduler or you wish to submit ad-hoc batch jobs from the online application.

When using the Thread Pool Worker option you should have the following settings when you run configureEnv.sh / cmd:

```
Batch Server Enabled: false

Batch Threads Number: 5

Batch Scheduler Daemon: false
```

Starting the Tread Pool Worker on UNIX

Once you have initialized an environment shell with splenviron.sh, the application threadpoolworker.sh should be executed as a background task.

```
nohup threadpoolworker.sh -p DEFAULT=5 -d Y
```

This will start a standalone threadpool worker process that offers 5 threads in threadpool DEFAULT ("-p DEFAULT=5") and hosts a scheduler daemon for online job submissions ("-d Y"). Threadpool DEFAULT is the default threadpool name that is used for jobs submitted from the online; if set to anything else, online submissions may not be picked up. The number of threads, specified as "5" here, can be adjusted.

The batch scheduler should be running as a background daemon under UNIX. To verify that it is running you can use a command similar to "ps -ef |grep threadpool".

Starting the Thread Pool Worker on Windows

```
threadpoolworker.cmd -p DEFAULT=5 -d Y
```

Note: If the JAVA_HOME environment variable is set, the script can be launched from Windows Explorer.

This will start a standalone threadpool worker process that offers 5 threads in threadpool DEFAULT ("-p DEFAULT=5") and hosts a scheduler daemon for online job submissions ("-d Y"). Threadpool DEFAULT is the default threadpool name that is used for jobs submitted from the online; if set to anything else, online submissions may not be picked up. The number of threads, specified as "5" here, can be adjusted.

Note: For AIX, you would need to change the LDR_CNTRL parameter in the threadpoolworker.sh file and change it to **"0XB0000000"**. This will help in improving threadpool performance.

Building Javadocs Indexes

The Javadocs would need to be rebuilt after upgrading the application. The following script allows rebuilding the Javadocs Indexes in the appViewer java module. This is needed after Customer Modification's have been applied to an environment; this needs to be run only if the Customer Modification includes Java Code.

Windows: `%SPLEBASE%\bin\buildJavadocsIndex.cmd`

UNIX: `ksh $SPLEBASE/bin/buildJavadocsIndex.sh`

Linux: `ksh $SPLEBASE/bin/buildJavadocsIndex.sh`

Appendix A: Known Issues

The known issues with the Oracle Revenue Management and Billing product “General Availability” release are listed in the release notes document which is delivered as part of the media download.

Appendix B: Third Party Software Upgrade

This section contains steps for carrying out any third party software upgrades.

Hibernate

Hibernate 3.2.5 to Hibernate 3.2.7 Upgrade Steps

This section contains steps for carrying out the hibernate upgrade for an environment which already has Oracle Revenue Management and Billing running Hibernate 3.2.5 on the application server.

1. Download hibernate3.2.7 jar from the internet.
2. Once downloaded, copy hibernate3.jar to the environment location as specified by HIBERNATE_JAR
3. Delete *hibernate-3.2.5.ga.jar* from businessapp and standalone folders
4. Copy hibernate3.jar into businessapp and standalone/lib folders and rename the file as *hibernate-3.2.7.ga.jar*
5. Open SPLApp.war @ splapp/applications to delete hibernate-3.2.5.ga.jar and copy *hibernate-3.2.7.ga.jar* in WEB-INF/lib within the war
6. Edit MANIFEST.MF @ splapp/businessapp/config to rename hibernate-3.2.5.ga to *hibernate-3.2.7.ga.jar*
7. Edit etc/ENVIRON.INI to replace *hibernate-3.2.5.ga.jar* to *hibernate-3.2.7.ga.jar*
8. Edit spl.properties @ splapp/businessapp/properties to replace *hibernate-3.2.5.ga.jar* to *hibernate-3.2.7.ga.jar*
9. Run bin/initialsetup.sh

Note: Refer to the Chapter 3: Software Requirements in the *Installation Guide* to view the third party software that you need to upgrade for the ORMB v2.2.4.0.