Oracle® Retail Merchandising System

Installation Guide
Release 13.2.8
E54674-02

October 2014

ORACLE

Oracle® Retail Merchandising System Installation Guide, Release 13.2.8

Primary Author: Wade Schwarz

Contributors: Nathan Young

Value-Added Reseller (VAR) Language

Oracle Retail VAR Applications

The following restrictions and provisions only apply to the programs referred to in this section and licensed to you. You acknowledge that the programs may contain third party software (VAR applications) licensed to Oracle. Depending upon your product and its version number, the VAR applications may include:

(i) the **MicroStrategy** Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(ii) the **Wavelink** component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.

(iii) the software component known as **Access Via™** licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(iv) the software component known as **Adobe Flex™** licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

Contents

Send Us Your Comments

Oracle Retail Merchandising System Installation Guide, Release 13.2.8

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

> **Note:** Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Applications Release Online Documentation CD available on My Oracle Support and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Oracle Retail Installation Guides contain the requirements and procedures that are necessary for the retailer to install Oracle Retail products.

Audience

This Installation Guide is written for the following audiences:

- Database administrators (DBA)
- System analysts and designers
- Integrators and implementation staff

Related Documents

You can find more information about this product in these resources:

- *Oracle Retail Merchandising System Release Notes*
- *Oracle Retail Merchandising System Operations Guide*
- *Oracle Retail Merchandising System Data Model*
- *Oracle Retail Merchandising Data Conversion Operations Guide*

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

https://support.oracle.com

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

Review Patch Documentation

When you install the application for the first time, you install either a base release (for example, 13.2) or a later patch release (for example, 13.2.8). If you are installing the base release or additional patch releases, read the documentation for all releases that have occurred since the base release before you begin installation. Documentation for patch releases can contain critical information related to the base release, as well as information about code changes since the base release.

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times **not** be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Technology Network Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Technology Network at the following URL:

`http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html`

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-**02** is an updated version of a document with part number E123456-**01**.

If a more recent version of a document is available, that version supersedes all previous versions.

Oracle Retail Documentation on the Oracle Technology Network

Documentation is packaged with each Oracle Retail product release. Oracle Retail product documentation is also available on the following Web site:

`http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html`

(Data Model documents are not available through Oracle Technology Network. These documents are packaged with released code, or you can obtain them through My Oracle Support.)

Documentation should be available on this Web site within a month after a product release.

Conventions

Navigate: This is a navigate statement. It tells you how to get to the start of the procedure and ends with a screen shot of the starting point and the statement "the Window Name window opens."

```
This is a code sample
    It is used to display examples of code
```

1

Preinstallation Tasks

Note: The RMS installer provides the option to configure multiple application deployment methods. See "Appendix: Application Deployment Method" to help determine your deployment approach.

Patch Contents

Patch releases include all defect fixes that have been released through bundled hot fix releases since the last patch release. Patch releases may also include new defect fixes and enhancements that have not previously been included in any bundled hot fix release.

Implementation Capacity Planning

There is significant complexity involved in the deployment of Oracle Retail applications, and capacity planning is site specific. Oracle Retail strongly suggests that before installation or implementation you engage your integrator (such as the Oracle Retail Consulting team) and hardware vendor to request a disk sizing and capacity planning effort.

Sizing estimates are based on a number of factors, including the following:

- Workload and peak concurrent users and batch transactions
- Hardware configuration and parameters
- Amount of data
- Application features utilized
- Length of time history is retained

Additional considerations during this process include your high availability needs as well as your backup and recovery methods.

A Note to Retailers Using the Croatian Language

See My Oracle Support document ID #393320.1 for important information regarding steps to enable the Croatian language for Forms and Reports.

A Note to Brazil Localization Retailers

If you are using Oracle Retail Fiscal Management (ORFM)/Brazil localization, you must read the entire *ORFM/RMS Brazil Localization Installation Guide* before proceeding with this *RMS Installation Guide*. The database installation order for RMS and ORFM must be followed exactly.

Requesting Infrastructure Software

If you are unable to find the necessary version of the required Oracle infrastructure software (database server, application server, WebLogic, etc.) on the Oracle Software Delivery Cloud, you should file a non-technical 'Contact Us' Service Request (SR) and request access to the media. For instructions on filing a non-technical SR, see My Oracle Support Note 1071023.1 – *Requesting Physical Shipment or Download URL for Software Media*.

Check Supported Database Server Requirements

General requirements for a database server running RMS include:

Supported on:	Versions Supported:
Database Server OS	OS certified with Oracle Database 11gR2 Enterprise Edition. Options are: ▪ Oracle Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Red Hat Enterprise Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Oracle Linux 6 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Red Hat Enterprise Linux 6 for x86-64 (Actual hardware or Oracle virtual machine). ▪ AIX 6.1 (Actual hardware or LPARs) ▪ AIX 7.1 (Actual hardware or LPARs) ▪ Solaris 10 SPARC (Actual hardware or logical domains) ▪ Solaris 11 SPARC (Actual hardware or logical domains) ▪ HP-UX Itanium11.31 Integrity (Actual hardware, HPVM, or vPars)
Database Server 11gR2	Oracle Database Enterprise Edition 11gR2 (11.2.0.4) with the following specifications: **Components:** ▪ Oracle Partitioning ▪ Examples CD (Formerly the companion CD) **Oneoff Patches:** ▪ 18465025: MERGE REQUEST ON TOP OF 11.2.0.4.0 FOR BUGS 18016963 18302329. **Other components:** ▪ Perl compiler 5 or later ▪ X-Windows interface ▪ ANSI compliant C-compiler (certified with OS and database version).

Check Supported Application Server Requirements

General requirements for an application server capable of running RMS include the following.

Note: Files required for Oracle Configuration Manager (OCM) are removed after OPatch is used to patch a WebLogic server. This will not cause the product installers to fail, but will cause OCM installation to fail. To work around this issue, back up the $ORACLE_HOME/utils/ccr/lib directory outside of the parent directory prior to applying a patch using OPatch, and recopy the lib directory back after you apply any patches. (If you backup the lib dir inside of the ccr dir, it will get overwritten when OPatch is run.)ORACLE_HOME is the location where WebLogic has been installed.

Supported on	Versions Supported
Application Server OS	OS certified with Oracle Fusion Middleware 11g Release1 (11.1.1.7). Options are: ▪ Oracle Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Red Hat Enterprise Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Oracle Linux 6 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Red Hat Enterprise Linux 6 for x86-64 (Actual hardware or Oracle virtual machine). ▪ AIX 6.1 (Actual hardware or LPARs) ▪ AIX 7.1 (Actual hardware or LPARs) ▪ Solaris 10 SPARC (Actual hardware or logical domains) ▪ Solaris 11 SPARC (Actual hardware or logical domains) ▪ HP-UX Itanium 11.31 Integrity (Actual hardware, HPVM, or vPars)

Supported on	Versions Supported
Application Server	Oracle Fusion Middleware 11g Release 1 (11.1.1.7) **Components:** ▪ Oracle WebLogic Server 11g Release 1 (10.3.6) ▪ Oracle Forms Services 11g Release 2 (11.1.2.2) with Oneoff Patches 17448420: MANIFEST ATTRIBUTE ERROR IN JAVA CONSOLE WHILE RUNNING FORMS URL WITH 7U45_B11 Java: JDK 1.7.0+ 64 bit **IMPORTANT:** If there is an existing WebLogic installation on the server, you must upgrade to WebLogic 10.3.6. All middleware components associated with WebLogic server should be upgraded to 11.1.1.7 and ORACLE Forms must have new install to 11gR2 forms (11.1.2.2). Back up the weblogic.policy file ($WLS_HOME/wlserver_10.3/server/lib) before upgrading your WebLogic server, because this file could be overwritten. Copy over the weblogic.policy backup file after the WebLogic upgrade is finished and the post patching installation steps are completed. **Optional (SSO required)** ▪ Oracle Identity Management 11gR1 (11.1.1.7) optionally with Oracle Access Manager 11gR1 (11.1.1.7) using OSSO agent. Must have separate WebLogic 10.3.6 for Oracle Access Manager 11g. **Other components:** ▪ Oracle BI Publisher 11g (11.1.1.7)

Verify Single Sign-On

If RMS is not being deployed in a Single Sign-On environment, skip this section.

If Single Sign-On is to be used, verify the Oracle Identity Management (OIM/IDM) 11gR1 version 11.1.1.7 has been installed along with the components listed in the above Application Server requirements section. Verify the HTTP Server is registered with the Oracle Access Manager (OAM) 11gR1 as a partner application.

> **Note:** Oracle Application Server (OAS) 10.1.4 Single Sign-On is only supported for existing installations, this support may be dropped in a future 13.2.X patch release.

Check Web Browser and Client Requirements

General requirements for client running RMS include the following.

Requirement	Version
Operating system	Windows 7
Display resolution	1024x768 or higher
Processor	2.6GHz or higher
Memory	1GByte or higher
Networking	intranet with at least 10Mbps data rate
Oracle (Sun) Java Runtime Environment	1.7.0+
Browser	Microsoft Internet Explorer version 9, or Mozilla Firefox 24

Supported Oracle Retail Products

Note: RMS is dependent on RPM database objects and stored procedures for initial item pricing and requires that this portion of RPM is always deployed with RMS. Without this, RMS would require customization and Oracle Retail does not provide guidance for this type of implementation. In addition to initial price there are other areas where dependencies exist such as vendor funded markdowns, vendor funded promotions, and margin visibility.

Product	Version
Oracle Retail Analytics	13.2.8
Oracle Retail Active Retail Intelligence (ARI)	13.2
Oracle Retail Price Management (RPM)	13.2.8
Oracle Retail Allocation	13.2.8 or 13.3
Oracle Retail Invoice Matching (ReIM)	13.2.8
Oracle Retail Store Inventory Management (SIM)	13.2.8
Oracle Retail Warehouse Management System (RWMS)	13.2.8
Oracle Retail Advanced Inventory Planning (AIP)	13.2.8
Oracle Retail Merchandise Financial Planning (MFP)	13.4.3
Oracle Retail Demand Forecasting (RDF)	13.4.3
Oracle Retail Grade	13.4.3
Oracle Retail Predictive Application Server (RPAS)	13.4.3
Oracle Retail POS Suite	13.3.6 or 13.4.8

Supported Oracle Retail Integration Technologies

Integration Technology	Version
Oracle Retail Extract, Transform and Load (RETL)	13.2.5
Oracle Retail Integration Bus (RIB)	13.2.8
Oracle Retail Service Layer (RSL)	13.2.8

Supported Oracle Applications

Requirement	Version
Oracle E-Business Suite Financials	Oracle E-Business Suite 12.1.3 integration is supported using the Oracle Retail Financial Integration for Oracle Retail Merchandising Suite and Oracle E-Business Suite Financials. See the *Oracle® Retail Financial E-Business Suite Integration Solution Implementation/Operations Guide* for specific version information.
PeopleSoft Enterprise Financials	Oracle Retail Financial Integration (ORFI) Media Pack Oracle E-Business Suite 12.1.3 integration is supported using the Oracle Retail Financial Integration Pack for Oracle Retail Merchandising Suite and Oracle E-Business Suite Financials.

Verify RMS and SIM Inventory Adjustment Reason Codes

SIM and RMS must have the same inventory adjustment reason codes to work properly, with the exception of the Pending Reason Code, which is used for internal purposes only.

2

RAC and Clustering

The Oracle Retail Merchandising has been validated to run in two configurations on Linux:

- Standalone WebLogic and Database installations
- Real Application Cluster Database and WebLogic Clustering

The Oracle Retail products have been validated against an 11.2.0.4 RAC database. When using a RAC database, all JDBC connections should be configured to use THIN connections rather than OCI connections

Clustering for WebLogic Server 10.3.6 is managed as an Active-Active cluster accessed through a Load Balancer. Validation has been completed utilizing a RAC 11.2.0.4 Oracle Internet Directory database with the WebLogic 10.3.6 cluster. It is suggested that a Web Tier 11.1.1.7 installation be configured to reflect all application server installations if SSO will be utilized.

References for Configuration:

- Oracle Fusion Middleware High Availability Guide 11g Release 1 (11.1.1) Part Number E10106-09
- Oracle Real Application Clusters Administration and Deployment Guide 11g Release 2 (11.2) Part Number E16795-11

3

Patch Installation

The database portion of RMS can be upgraded from release 13.1.2 or patched from any 13.2.x release. This guide details the steps needed to perform a patch installation of RMS. For additional information on the upgrade, see the My Oracle Support document, *Oracle Retail Upgrade Guide* (ID 1073414.1).

The following chapters document the patch process:

- Chapter 4 RMS Database Installation – Patch
- Chapter 5 Batch Installation Tasks – Patch
- Chapter 6 Application Server Installation Tasks – Patch
- Chapter 7 RMS Reports Installation – Patch
- Chapter 8 Data Migration
- Chapter 9 WebServices Installation

4

RMS Database Installation — Patch

There are two different methods to use for installing the RMS 13.2.8 database schema patch. Option 1 uses the installer to apply the patch. Option 2 uses the patch controller scripts directly.

Note: The patching mechanism has been updated for the 13.2 release. Any patches that were released prior to 13.2 (For example, 13.1.1 and 13.1.2) will not be compatible with this installer.

Note: If any RMS, RPM, ReIM or Allocation hot fixes have been applied to the schema after 13.2, be aware that using the installer or controller scripts to apply the 13.2.8 patch can have unexpected results. You will need to decide if it is safe to run all the scripts in the patch, or if the scripts need to be selectively run.

Note: If you are patching from 13.2.1 or from an earlier release, and you have not already run the Data Migration utility in a previous release, this utility should be run after applying the 13.2.2 deltas and before applying any later patches (for example, 13.2.3). See Chapter 8, "Data Migration."

Note: Shut down any applications that may be using the RMS schema (for example RIB) before applying a schema patch.

Note: If you are running the 13.2.1 patch as part of the database installation, you must make sure all the cost events for RECLASS are processed and purged before the 13.2.1 patch is applied. Ensure that before running this patch the table is empty and the existing data is already processed and purged in nightly batch. See "Appendix: Common Installation Errors" for more details.

Note: If you are utilizing the AIA 2.5 solution for PeopleSoft or EBS, you should not apply the RMS 13.2.6 patch or later. Please contact customer support for details

Option 1: Patch RMS Database using the Patch Installer

The RMS 13.2.8 database schema patch installer may be used to apply the RMS patch. The installer should only be used to apply the patch if the schema being patched does not contain customizations or hot fixes. The patch may also be applied outside of the installer by calling the controller scripts directly. See "Option 2: Patch RMS Database using Controller Scripts" in this chapter for details on this method.

Before you apply the RMS 13.2.8 patch:

- Make a backup of all your objects and database schema.
- Determine which patches and hot fix bundles have already been installed.
- Review the enclosed RMS 13.2.8 Release Notes (rms-1328-rn.pdf).
- Review each of the enclosed defect documents.
- Make sure any applications that connect to the RMS schema are shut down. This includes RPM, ReIM, Allocation, RIB, and anything else that could be using the schema.

Create Staging Directory for RMS Database Schema Files

To create a staging directory for RMS database schema files, complete the following steps.

1. Log into the database server as a user that can connect to the RMS database.
2. Create a staging directory for the MOM 13.2.8 Patch.
3. Copy the mom-dbpatch.zip file from the RMS 13.2.8 release to the staging directory. This is referred to as DB_PATCH_DIR when patching a database schema.
4. Change directories to DB_PATCH_DIR and extract the mom-dbpatch.zip file. This creates a rms/dbschemapatch subdirectory under DB_PATCH_DIR

Edit controller ksh Scripts

To edit controller ksh scripts, complete the following steps.

1. Change directories to DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch.
2. For each product you are going to patch, edit the corresponding <product>_controller.ksh.

   If you edit rms_controller.ksh, you must also edit rpm_controller.ksh and vice versa. alloc_controller.ksh and alloc_rms_controller.ksh must both be edited if patching Allocation. To edit these files, open up <product>_controller.ksh, and comment or uncomment the sections that perform the patches or hot fixes you want to apply. If you are patching from 13.2 and want to go to 13.2.8, you need to run all the patches up to 13.2.8. For this example you would need to edit the files to uncomment each patch after 13.2 (13.2.1, 13.2.2, and 13.2.3, 13.2.3.1, 13.2.3.2, 13.2.3.3, 13.2.4, 13.2.5, 13.2.6, 13.2.7 and 13.2.8).

   If you are patching from 13.2.7 and want to go to 13.2.8, you only need to run the patches after 13.2.7 (which would only be 13.2.8).

   For this example you would not need to edit the files (the correct patches should already be uncommented in the default scripts).

Run the RMS Database Schema Patch Installer

Note: See "Appendix: RMS Application Installer Screens" for details on screens and fields in the RMS database schema patch installer.

1. Change directories to DB_PATCH_DIR/rms/dbschemapatch.
2. Source the oraenv script to set up the Oracle environment variables (ORACLE_HOME, ORACLE_SID, PATH, etc)

   Example:
   ```
   prompt$ . oraenv
   ORACLE_SID = [] ? mydb
   prompt$
   ```

   Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example:
   ```
   prompt$ echo $ORACLE_HOME
           /u00/oracle/product/mydbversion
   prompt$ echo $ORACLE_SID
           mydb
   ```

3. Set and export the following environment variables. These variables are needed in addition to the environment variables set by the oraenv script above.

Variable	Description	Example
NLS_LANG	Locale setting for Oracle database client	NLS_LANG=AMERICAN_AMERICA.UTF8 export NLS_LANG
DISPLAY	Address and port of X server on desktop system of user running install. Optional for dbschema installer	DISPLAY=<IP address>:0.0 export DISPLAY

4. If you are going to run the installer in GUI mode using an X server, you need to have the XTEST extension enabled. This setting is not always enabled by default in your X server. See "Appendix: Common Installation Errors" for details.
5. If the patch installer has already been run in this location you may wish to back up the ant.install.properties file. The settings from the RMS 13.2.8 patch install will be refreshed with the latest input every time the installer runs.
6. Run the install.sh script to start the installer.

   Note: Below are the usage details for install.sh. The typical usage for GUI mode is no arguments.

   install.sh [text | silent]

7. On the "Apply a Patch" page for each product, provide the path to the corresponding controller ksh script. If you are only applying a single patch or hotfix bundle (for example, to get from 13.2.7 to 13.2.8), this path will be DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product>. If you are applying multiple patches or hotfix bundles (for example, to get from 13.2.1 to 13.2.8), this will be DB_PATCH_DIR/ rms/dbschemapatch/mom-dbpatch. This directory should contain a <product>_controller.ksh file (for example, rms_controller.ksh), which the installer runs to apply the RMS 13.2.8 patch.

> **Note:** Depending on which patches are run, database installation can take several hours to complete.

8. After the installer is complete, you can check its log file: rms-install-dbschema.<timestamp>.log.
9. The installer leaves behind the ant.install.properties file for future reference and repeat installations. This file contains inputs you provided. As a security precaution, make sure that the file has restrictive permissions.

> **Example:** `chmod 600 ant.install.properties`

10. For instructions on installing the new languages after running the RMS patch installer, see "Appendix: Inserting New Languages."

> **Note:** For Brazilian retailers planning to install the ORFM/RMS Brazil Localization patch, you may have invalid objects in your schema after running the RMS patch. If the RMS installation finished successfully without reporting that the installer failed, you can safely ignore these invalids and continue with the ORFM patch installer. After running the ORFM patch, these invalids should be resolved.

Option 2: Patch RMS Database using Controller Scripts

While the installer can be used to apply the entire RMS database patch, there are situations in which it is better to run the patch directly with the scripts released in the patch. The installer calls start-all ksh scripts named <product>_controller.ksh which run all of the files in the patch. If there are any customizations or hot fixes in the schema then certain statements in the patch may result in errors. In this situation it is better to investigate where the conflicts are and fix the SQL scripts accordingly.

Before you apply the RMS 13.2.8 patch:

- Make a backup of all your objects and database schema.
- Determine which patches and hot fix bundles have already been installed
- Review the enclosed RMS 13.2.8 patch Release Notes (rms-1328-rn.pdf).

Create Staging Directory for RMS Database Schema Files

To create a staging directory for RMS database schema files, complete the following steps.

1. Log into the database server as a user that can connect to the RMS database.
2. Create a staging directory for the MOM 13.2.8 patch.
3. Copy the mom-dbpatch.zip file from the RMS 13.2.8 release to the staging directory. This is referred to as DB_PATCH_DIR when upgrading a database schema.
4. Change directories to DB_PATCH_DIR and extract the mom-dbpatch.zip file. This creates a rms/dbschemapatch subdirectory under DB_PATCH_DIR.

Run the RMS Database Controller Scripts

To run the RMS database controller scripts, complete the following steps.

1. Change directories to DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/.
2. Source the oraenv script to set up the Oracle environment variables (such as ORACLE_HOME and ORACLE_SID, PATH).

   Example:
   ```
   prompt$ . oraenv
   ORACLE_SID = [] ? mydb
   prompt$
   ```

3. Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example:
   ```
   prompt$ echo $ORACLE_HOME
           /u00/oracle/product/mydbversion
   prompt$ echo $ORACLE_SID
           mydb
   ```

4. Set and export the NLS_LANG environment variable.

   Example:
   ```
   NLS_LANG=AMERICAN_AMERICA.UTF8
   export NLS_LANG
   ```

5. For each product and version you want to patch, configure the individual controller.cfg files as follows:
   - Copy DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch /<version>/<product>/templates/controller.cfg to DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch /<version>/<product>/controller.cfg
   - Open the controller.cfg file you just created and replace the tokens for the following variables with the appropriate values:
     i. export PATCH_DIR= DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product>
     ii. export SCHEMA_OWNER=<The name of the RMS schema>
     iii. export MMUSER=<The name of the schema to Patch >

          For RMS, RPM, ReIM, and Alloc_RMS, this will be the RMS schema

          For Alloc, this will be the Allocation schema
     iv. export ORACLE_SID=<SID for the database the MMUSER schema resides in>
     v. export TNS_ADMIN=/path/to/wallet/files/dir/
     vi. export UP=/@<Schema Owner Wallet Alias>

     Note: See "Appendix: Setting Up Password Stores with Oracle Wallet," later in this document for instructions to set up database wallet.

6. The patches should be run in the following order: RMS, RPM, ReIM, Alloc_RMS, and Allocation. If you are upgrading from 13.2.1 and want to get to 13.2.8, you need to run all the patches from DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/ (13.2.1 through 13.2.8). If you are patching from 13.2.7 and want to get to 13.2.8, you just need to apply the 13.2.8 patch. The Alloc controller is used to apply the necessary Allocation patch to the Allocation schema, while the Alloc_RMS controller is used to apply the necessary Allocation patch to the RMS schema. While you can choose not to run any of the patches, all of the non-RMS patches depend on the RMS patch being run. If you patch RMS you should also patch RPM; there is also a dependency between Alloc_RMS/Alloc. For each product you wish to patch, cd to DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product> and run the following commands:

   ```
   For RMS run: $ ./rms_controller.ksh DBO N
   For RPM run: $ ./rpm_controller.ksh DBO Y
   For ReIM run: $ ./reim_controller.ksh DBO Y
   For Alloc_rms run: $ ./alloc_controller.ksh DBO Y
   For Allocation run: $ ./alloc_rms_controller.ksh DBO Y
   ```

   Note: The controllers should be run in this order.

7. If the installation fails for any of the patches before completion, look at the logs in the DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product>/error and DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product>/log directories to determine the source of the error. You can continue the patch by rerunning the <product>_controller.ksh file, but only if the files generated in the DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product>/processed directory from the last patch attempt are still there. Any scripts that ran previously will be skipped. If you wish to start a new patch, delete all files in the DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/<version>/<product>/processed directory.
8. If you are patching Allocation, after finishing the database patch you may need to manually create some synonyms from the RMS schema to the Allocation schema, and grants from the Allocation schema to the RMS schema. Verify that the following synonyms exist in the RMS schema and that they are pointing to the table with the same name in the Allocation schema:
   - ALC_DEFAULT_CHRGS_TEMP
   - ALC_ON_HAND_QTY_TEMP
   - ALC_RLOH_TEMP
   - ALC_ITEM_LIST_TEMP
   - ALC_ITEM_LOC_TEMP

   If any of these synonyms do not exist, run the following command as the RMS schema owner for each table to create the missing synonym:

   ```
   create or replace synonym <RMS schema>.<TABLE NAME> for <ALLOC
   schema>.<TABLE_NAME>
   ```

   For example:

   ```
   create or replace synonym RMS01.ALC_RLOH_TEMP for ALLOC01.ALC_RLOH_TEMP
   ```

   In addition, grant permissions on tables in the Allocation schema to the RMS schema. Run the following commands as the Allocation user to grant permissions:

   ```
   grant insert on ALC_ON_HAND_QTY_TEMP to <RMS schema>
   grant insert, update on ALC_ITEM_LOC_TEMP to <RMS schema>
   ```

For example:

```
grant insert on ALC_ON_HAND_QTY_TEMP to RMS01
grant insert, update on ALC_ITEM_LOC_TEMP to RMS01
```

9. For instructions on installing the new languages after running the RMS patch installer, see "Appendix: Inserting New Languages."

> **Note:** For Brazilian retailers planning to install the ORFM/RMS Brazil Localization patch, you may have invalid objects in your schema after running the RMS patch. If the RMS installation finished successfully without reporting that the installer failed, you can safely ignore these invalids and continue with the ORFM patch installer. After running the ORFM patch, these invalids should be resolved.

5

Batch Installation Tasks—Patch

There are two different methods to use for installing the RMS 13.2.8 batch patch. Option 1 uses the installer to apply patch. Option 2 compiles the batch directly.

Option 1: Use Batch Installer to Patch

As shipped, the RMS 13.2 batch installer will install and compile the batch programs for version 13.2. Patches for RMS batch may be applied by copying the new source files and recompiling in place in the environment using the profile scripts created by the installer. This is the method to use if there is already an environment at the previous patch level. Go to the section "Option 2: Compile RMS Batch Directly" for these instructions.

The installer method is only intended for new environments. Do not use the installer patching utility to attempt patching of batch in existing environments with the installer. If the patch is applied to customizations, they will be overwritten.

In this section, STAGING_DIR refers to the location where the RMS 13.2 batch installer was originally expanded. The installer files from the original RMS 13.2 installation can be re-used or a new directory can be created with a fresh copy of the RMS 13.2 application installer.

Before you apply the RMS 13.2.8 batch patch:

- Make a backup of all your batch files.
- Review the enclosed RMS 13.2.8 Patch Release Notes (rms-1328-rn.pdf).

Before copying over any files:

- Note whether customizations have been made to the module. If so, then the customizations must be reapplied over the new version of the file (or the fix may need to be applied to the custom version of the code).
- Copy the original files to a different directory before copying over them in case they need to be referred to at a later date.

Create Staging Directory for RMS Batch Patch Files

To create a staging directory for RMS batch patch files, complete the following steps.

1. Log into the database server as a user that can connect to the RMS database.
2. Create a staging directory for the RMS 13.2.8 batch patch.
3. Copy the rms1328batchpatch.zip file from the RMS 13.2.8 release to the staging directory. This is referred to as BATCH_PATCH_DIR when patching a database schema.
4. Change directories to BATCH_PATCH_DIR and extract the rms1328batchpatch.zip file. This creates a batch-patch subdirectory under BATCH_PATCH_DIR.
5. If you do not already have one, create a staging directory for the RMS batch installation software or use the same staging directory as created in the database schema step above.

6. Copy the **rms13batch.zip** file from the RMS 13. 2 release to the staging directory. This is referred to as STAGING_DIR when installing the RMS batch software.
7. Change directories to STAGING_DIR and extract the rms13batch.zip file. This creates an rms/batch subdirectory under STAGING_DIR.

Copy Batch Files

For new environments, the installer can be used to install and compile the batch programs at the latest patch level using the installer patching utility included with RMS batch patches. The utility is located under BATCH_PATCH_DIR/batch-patch/patch-util. This utility will accept as input the RMS patch files and add them to the RMS 13.2 Batch installer package. After running this utility, the RMS batch installer can be used to install the environment, and they will install the latest version of each batch module.

Custom Modules

Custom source can be provided by the user in a folder named BATCH_PATCH_DIR/batch-patch/patch-util/custom. The source code in this folder is applied last, after all patches have been applied.

Run the Installer Patching Utility

To run the installer patching utility, complete the following steps.

1. Set the JAVA_HOME environment variable to point to a JDK.
2. Set the ANT_HOME environment variable to point to an Ant installation. There is one included with the RMS installer that can be used for this.

   ANT_HOME=STAGING_DIR/rms/batch/ant

   export ANT_HOME
3. Change directories to BATCH_PATCH_DIR/batch-patch/patch-util/
4. Modify the patch.properties file. Set the installer.dir and patch.to.version properties.

Variable	Description
installer.dir	The directory where the installer files are located under STAGING_DIR. This directory will contain the install.sh file. For example: /opt/rms/batch
patch.to.version	The version to which you want to patch. For example: 13.2.8

5. Run the patch.sh script. This script will copy the files from each patch from 13.2.1 up to the patch specified in the **patch.to.version** property. These files are copied into the installer package.

Run Batch Installer

> **Note:** See "Appendix: RMS Batch Installer Screens" for details on every screen and field in the batch installer.

1. Change directories to STAGING_DIR/rms/batch. This directory was created when the rms13batch.zip file was expanded under STAGING_DIR.
2. Source the oraenv script to set up the Oracle environment variables (ORACLE_HOME, ORACLE_SID, PATH, etc)

   Example:
   ```
   prompt$ . oraenv
   ORACLE_SID = [] ? mydb
   prompt$
   ```

   Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example:
   ```
   prompt$ echo $ORACLE_HOME
   /u00/oracle/product/mydbversion
   prompt$ echo $ORACLE_SID
   mydb
   ```
3. Verify that the following executables are available from PATH: make, makedepend, cc, ar.

   Example: Here are some locations where makedepend is commonly found:

Linux:	/usr/X11R6/bin
SUN:	/usr/openwin/bin
AIX:	/usr/X11R6/bin
HP-UX:	/opt/imake/bin

   > **Note:** For environments running Solaris and Oracle Database 11.2.0.4, ensure Sun Studio 12 compiler is in the path.
   >
   > Example: export PATH=/vol.rtk/compilers/sunstudio12.1/bin:/usr/xpg4/bin:$PATH
4. Set and export the following environment variables. These variables are needed in addition to the environment variables set by the oraenv script above.

Variable	Description	Example
DISPLAY	Address and port of X server on desktop system of user running install. Optional for batch installer	DISPLAY=<IP address>:0.0 export DISPLAY

5. If you are going to run the installer in GUI mode using an X server, you need to have the XTEST extension enabled. This setting is not always enabled by default in your X server. See "Appendix: Common Installation Errors" for more details.
6. Run the install.sh script to start the installer.

Note: Below are the usage details for install.sh. The typical usage for GUI mode is no arguments.

./install.sh [text | silent]

Depending on system resources, a typical RMS batch installation takes anywhere from 20 to 60 minutes.

The installer will ask for an installation directory. This is the destination directory for the RMS files. This directory is referred to as INSTALL_DIR for the remainder of this chapter. Do not provide an INSTALL_DIR that is located at or underneath STAGING_DIR.

7. After the installer is complete, you can check its log file: rms.batch.install.<timestamp>.log.
8. The installer leaves behind the ant.install.properties file for future reference and repeat installations. This file contains inputs you provided. As a security precaution, make sure that the file has restrictive permissions.

Example: `chmod 600 ant.install.properties`

Note: The object counts performed by the installer may be off due to the patch adding or removing modules. The installer warnings about this can be ignored

Resolving Errors Encountered During Batch Installation

The RMS batch installer is a full install that starts from the beginning each time it is run. If you encounter errors in your environment, after resolving the issue you can safely run the batch installer again to attempt another installation.

RETL

The RMS batch installer installs the RETL files under INSTALL_DIR /RETLfor<product>/rfx.

See "Appendix: RMS RETL Instructions" for more information.

Data Conversion Scripts

The RMS batch installer installs the data conversion scripts under INSTALL_DIR /external/scripts. To complete the setup of these files, perform the following steps.

1. Create the following new directories:

   INSTALL_DIR/external/data

   INSTALL_DIR/external/logs

   The RMS Batch installer should have already created INSTALL_DIR/external/scripts.
2. Log into sqlplus as SYSTEM and run the following commands:

```
SQL> create or replace directory rmsdc_ext_data as
'INSTALL_DIR/external/data';
SQL> create or replace directory rmsdc_ext_logs as
'INSTALL_DIR/external/logs';
```

Note: You need to replace INSTALL_DIR with your INSTALL_DIR and you can rename the external data and log directory.

Note: The user that creates these directories owns them.

Note: The data and logs directories should be chmoded 777.

3. Log into sqlplus as SYSTEM and grant access to them by running the following commands:

```
SQL> grant read on directory rmsdc_ext_data to public;
SQL> grant read, write on directory rmsdc_ext_logs to public
```

4. Grant the following privileges to any other users who will be using data conversion.

```
SQL> grant read on directory rmsdc_ext_data to RMS01;
SQL> grant read, write on directory rmsdc_ext_logs to RMS01;
```

Option 2: Compile RMS Batch Directly

Note: Warning messages may appear during the compilation of the batch. These warnings can be ignored if the batch executables are successfully generated.

Create Staging Directory for RMS Batch Patch Files

To create a staging directory for RMS batch patch files, complete the following steps.

1. Log into the database server as a user that can connect to the RMS database.
2. Create a staging directory for the RMS 13.2.8 batch patch.
3. Copy the rms1328batchpatch.zip file from the RMS 13.2.8 release to the staging directory. This is referred to as BATCH_PATCH_DIR when patching the RMS batch.
4. Change directories to BATCH_PATCH_DIR and extract the rms1328batchpatch.zip file. This creates a batch-patch subdirectory under BATCH_PATCH_DIR.

Set Environment Variables

Note: INSTALL_DIR is the location where RMS 13 batch was installed.

Make sure the following variables are set. The RMS 13.2.8 batch installer should have created a batch.profile file located at INSTALL_DIR/batch.profile. This profile script can be used to set all of the environment variables listed below.

Example: cd <INSTALL_DIR>
. ./batch.profile

Variables set by batch.profile:

- PATH must include make, makedepend and the C compiler
- MMHOME=INSTALL_DIR/rms
- MMUSER=RMS Schema Owner
- ORACLE_HOME=Location of Oracle install
- ORACLE_SID=The Oracle Sid for the RMS database
- UP=/@< Schema Owner Wallet Alias >
- TNS_ADMIN=/path/to/wallet/files/dir/

AIX:

- LIBPATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$LDPATH
- OBJECT_MODE=64
- LINK_CNTRL=L_PTHREADS_D7

HP:

- SHLIB_PATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$SHLIB_PATH

Solaris:

- LD_LIBRARY_PATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$LD_LIBRARY_PATH

> **Note:** For environments running Solaris and Oracle RDBMS 11.2.0.4, ensure Sun Studio 12 compiler is in the path.
>
> Example: export PATH=/vol.rtk/compilers/sunstudio12.1/bin:/usr/xpg4/bin:$PATH

Linux:

- LD_LIBRARY_PATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$LD_LIBRARY_PATH

Compile Batch Libraries

To compile batch libraries, complete the following steps.

1. If they exist, copy the files from BATCH_PATCH_DIR/batch-patch/<version>/oracle/lib/src to INSTALL_DIR/oracle/lib/src. This step should be done with each version you want to apply in order of earliest to latest patch starting at 13.2 and ending with the 13.2.8 deltas.
2. Change directories to INSTALL_DIR/oracle/lib/src.
3. To make library dependencies run one of the following commands:
   - For Linux use:
     ```
     make –f retek.mk –r depend 2>&1 | tee libdpnd.log
     ```
   - For other platforms use:
     ```
     make –f retek.mk depend 2>&1 | tee libdpnd.log
     ```

   Check the libdpnd.log file for errors

4. To make batch libraries:
   - For Linux use:
     ```
     make –f retek.mk –r retek rms resa 2>&1 | tee libretek.log
     ```
   - For other platforms use:
     ```
     make –f retek.mk retek rms resa 2>&1 | tee libretek.log
     ```

   Check the libretek.log file for errors
5. To install batch libraries:
   ```
   make –f retek.mk install
   ```
   The batch libraries should now be in INSTALL_DIR/oracle/lib/bin

Compile Batch Source Code

To compile batch source code, complete the following steps.

1. If they exist, copy the files from BATCH_PATCH_DIR/batch-patch/<version>/oracle/proc/src to INSTALL_DIR/oracle/proc/src. This step should be done with each version in order of earliest to latest patch starting at 13.2 and ending with the 13.2.8 deltas.
2. Change directories to INSTALL_DIR/oracle/proc/src.
3. Create dependencies.
   a. Run one of the following commands:
      - For Linux use:
        ```
        make –f mts.mk –r depend 2>&1 | tee srcdpnd.log
        ```
      - For other platforms use:
        ```
        make –f mts.mk depend 2>&1 | tee srcdpnd.log
        ```

   b. Check the srcdpnd.log file for errors.
4. Create batch programs.
   a. Run the following commands in the order stated.
      - For Linux use:
        ```
        make –f rms.mk –r PRODUCT_PROCFLAGS=dynamic=ansi ditinsrt
        make –f mts.mk –r rms-ALL recs-ALL resa-ALL rtm-ALL fif-ALL 2>&1 | tee srcall.log
        ```
      - For other platforms use:
        ```
        make –f rms.mk PRODUCT_PROCFLAGS=dynamic=ansi ditinsrt
        make –f mts.mk rms-ALL recs-ALL resa-ALL rtm-ALL fif-ALL 2>&1 | tee srcall.log
        ```

   b. Check the srcall.log file for errors.
5. Install the batch programs.
   ```
   make –f mts.mk install
   ```

The batch programs should now be in INSTALL_DIR/oracle/proc/bin.

Copy RETL Code

If they exist, copy the files from BATCH_PATCH_DIR/batch-patch/<version>/RETLfor<product> to INSTALL_DIR/RETLfor<product>. This step should be done with each version in order of earliest to latest patch starting at 13.2 and ending with the 13.2.8 deltas.

Copy Data Conversion Scripts

If they exist, copy the files from BATCH_PATCH_DIR/batch-patch/<version>/external to INSTALL_DIR/external. This step should be done with each version in order of earliest to latest patch starting at 13.2 and ending with the 13.2.8 deltas.

6

Application Server Installation Tasks—Patch

There are two methods for installing the RMS 13.2.8 application. Option 1 uses the installer to apply the patch. Option 2 compiles the RMS toolset and forms directly.

Note: If the 13.2.8 database schema patch has been run, the entire set of toolsets and forms must be recompiled.

IMPORTANT: If there is an existing WebLogic installation on the server, you must upgrade to WebLogic 10.3.6. All middleware components associated with WebLogic server should be upgraded to 11.1.1.7. A new forms 11gr2 install (11.1.2.2) will be needed along with the WebLogic upgrade.

Back up the weblogic.policy file ($WLS_HOME/wlserver_10.3/server/lib) before upgrading your WebLogic server, because this file could be overwritten. Copy over the weblogic.policy backup file after the WebLogic upgrade is finished and the post patching installation steps are completed.

Option 1: Use Application Installer to Patch

As shipped, the RMS 13.2 Forms installer installs and compiles the forms for version 13.2. Patches for RMS Forms may be applied by copying the new source files and recompiling in place in the environment using the profile scripts created by the installer. This is the method to use if there is already an environment at the previous patch level. Go to the section, "Option 2: Compile RMS Toolset and Forms Directly," for these instructions.

The installer method is only intended for new environments. Do not use the installer patching utility to attempt patching of forms in existing environments with the installer. If the patch is applied to customizations, they will be overwritten.

In this section, STAGING_DIR refers to the location where the RMS 13.2 application installer was originally expanded. A new directory should be created with a fresh copy of the RMS 13.2 application installer.

Before you apply the RMS 13.2.8 patch:

- Make a backup of all your forms and library files.
- Review the enclosed RMS 13.2.8 Patch Release Notes (rms-1328-rn.pdf).

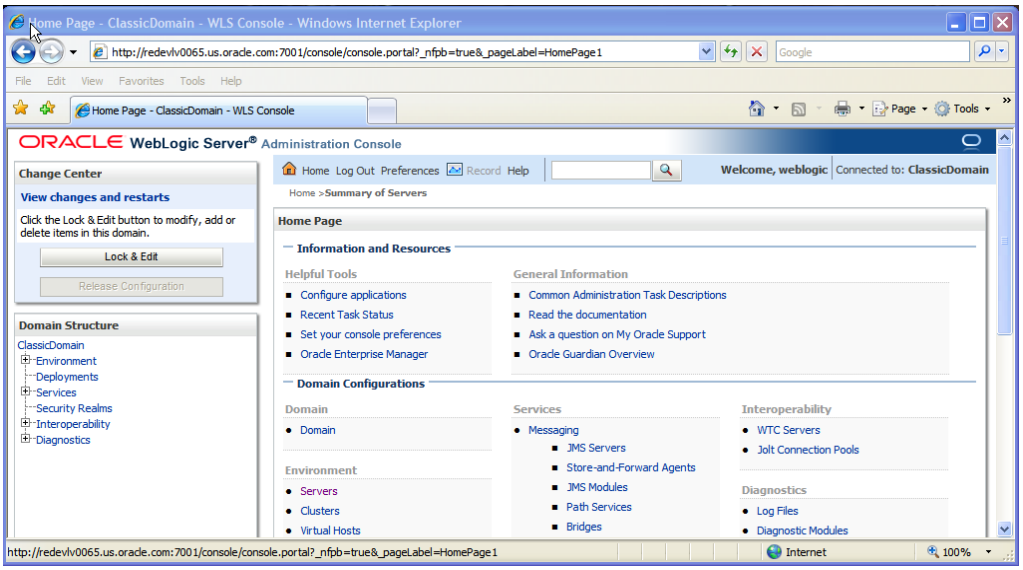
Before copying over any files:

- Note whether customizations have been made to the module. If so, then the customizations must be reapplied over the new version of the file (or the fix may need to be applied to the custom version of the code).
- Copy the original files to a different directory before copying over them in case they need to be referred to at a later date.

Create RMS Help Managed Server

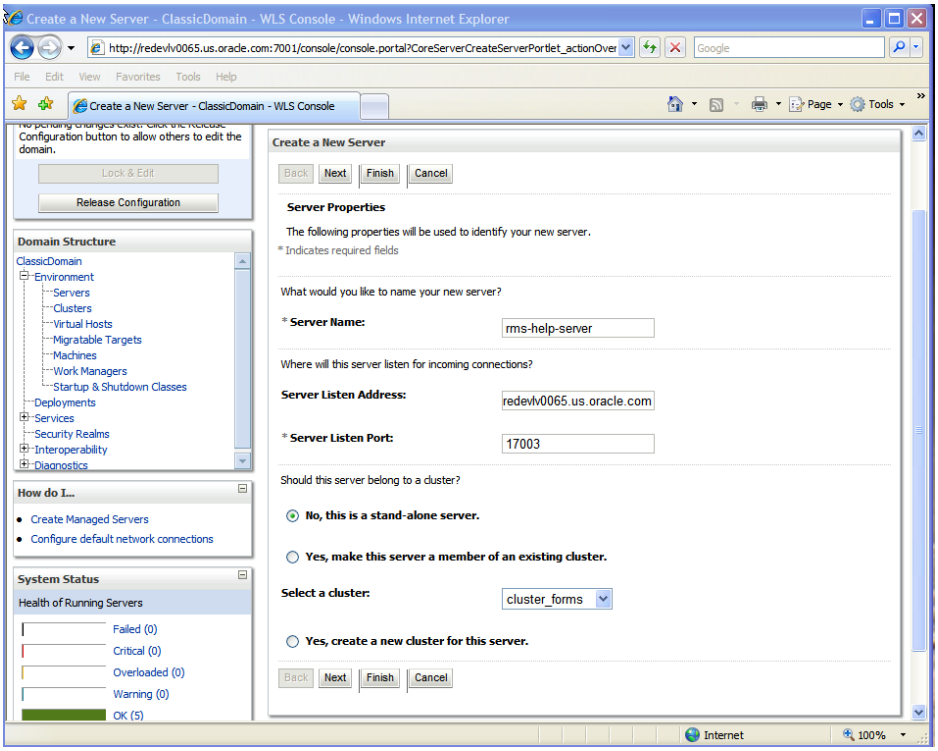
Note: If RMS help managed server is already installed, skip this section.

1. Log in to the administration console.



2. Click **Lock & Edit**.

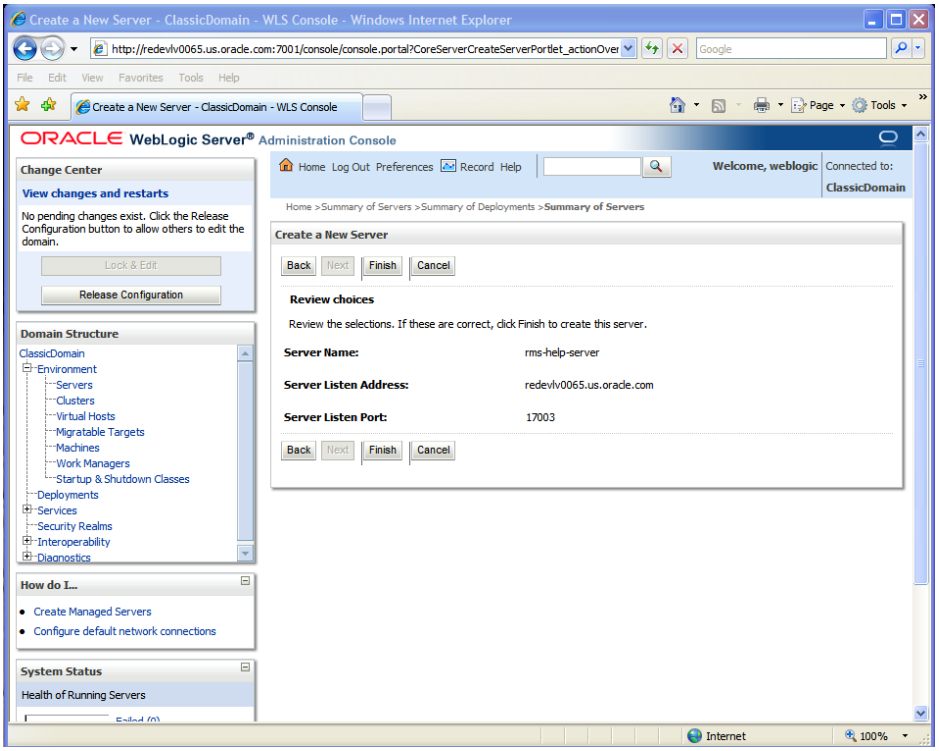
3. Navigate to Environment > Servers and select new tab of the servers on the right side.



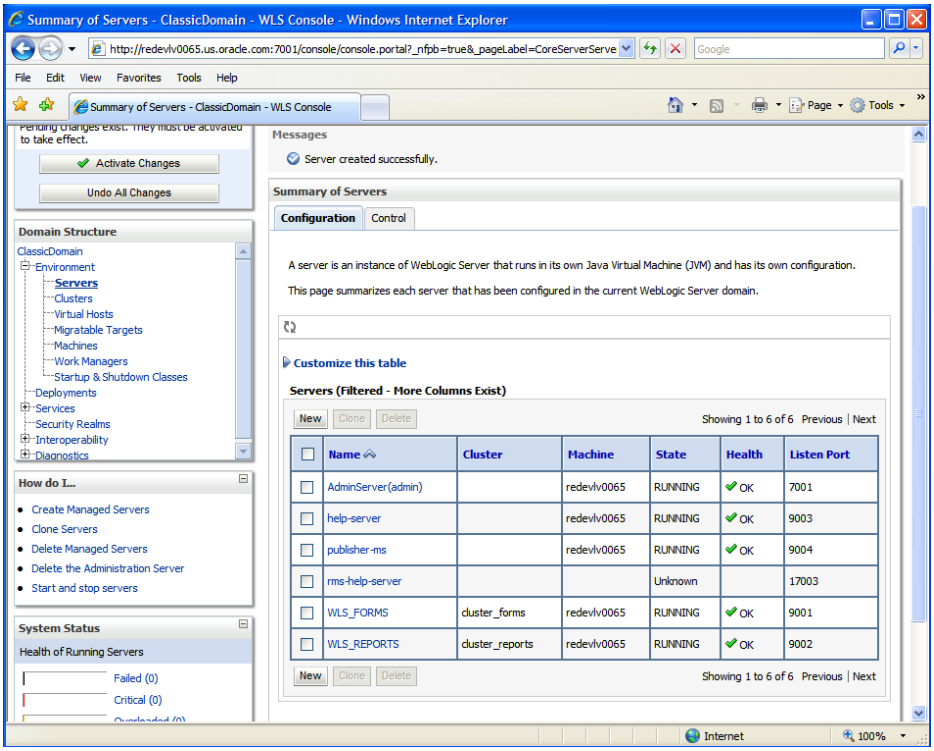
4. Set the following variables:
   - **Server Name**: These should be some name specific to your application targeted (for example, rms-help-server).
   - **Server Listen Address**: <weblogic server> (for example, redevlv0065.us.oracle.com)
   - **Server Listen Port**: A free port; check for availability.

     A suggestion is to increment the AdminServer port by two and keep incrementing by two for each managed server (for example, 17003, 17005, 17007, and so on).

5. Click **Next**.


Create a New Server - ClassicDomain - WLS Console - Windows Internet Explorer
http://redevlv0065.us.oracle.com:7001/console/console.portal?CoreServerCreateServerPortlet_actionOver
Create a New Server - ClassicDomain - WLS Console
ORACLE WebLogic Server® Administration Console
Change Center
View changes and restarts
No pending changes exist. Click the Release Configuration button to allow others to edit the domain.
Lock & Edit
Release Configuration
Domain Structure
ClassicDomain
Environment
Servers
Clusters
Virtual Hosts
Migratable Targets
Machines
Work Managers
Startup & Shutdown Classes
Deployments
Services
Security Realms
Interoperability
Diagnostics
How do I...
Create Managed Servers
Configure default network connections
System Status
Health of Running Servers
Home Log Out Preferences Record Help
Welcome, weblogic
Connected to:
ClassicDomain
Home >Summary of Servers >Summary of Deployments >Summary of Servers
Create a New Server
Back
Next
Finish
Cancel
Review choices
Review the selections. If these are correct, click Finish to create this server.
Server Name:
rms-help-server
Server Listen Address:
redevlv0065.us.oracle.com
Server Listen Port:
17003
Back
Next
Finish
Cancel
Internet
100%

6. Click **Finish**.

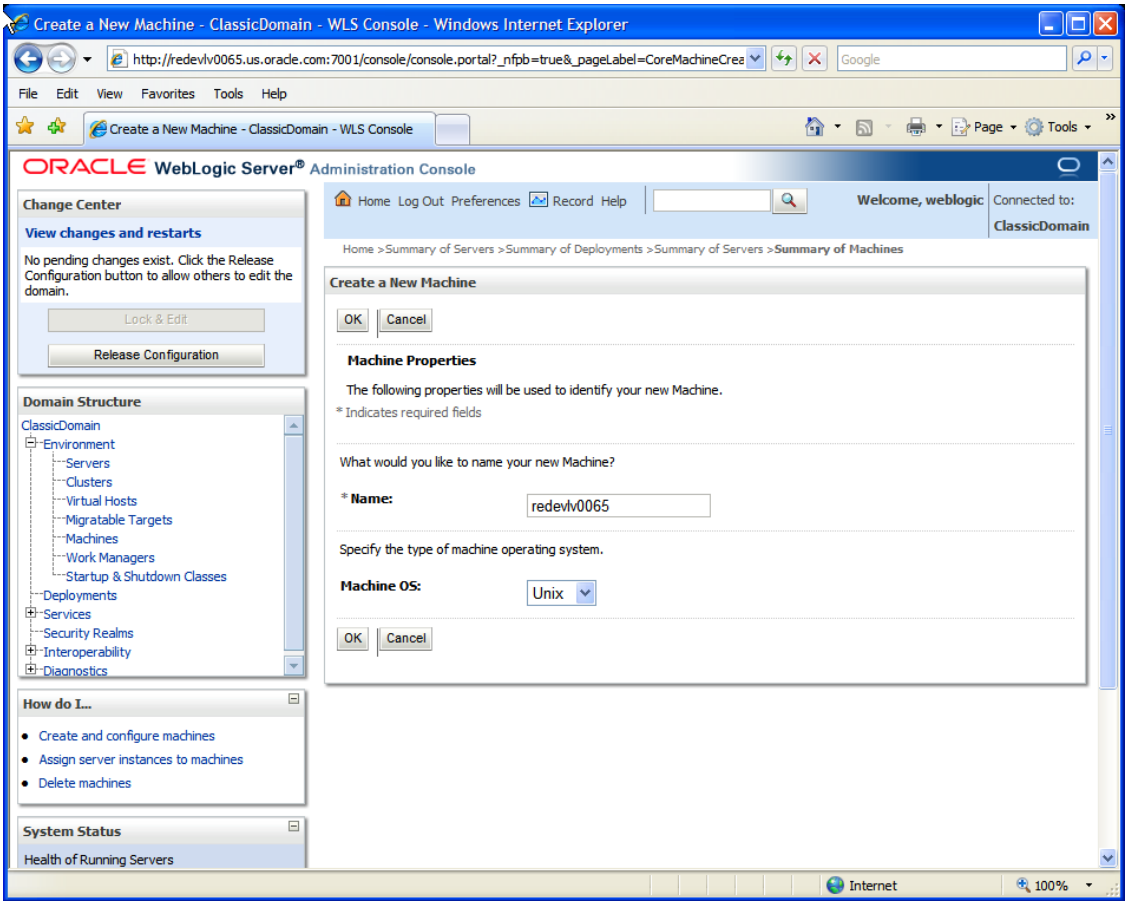


7. Click **Activate Changes** on the left side.

Install Node Manager

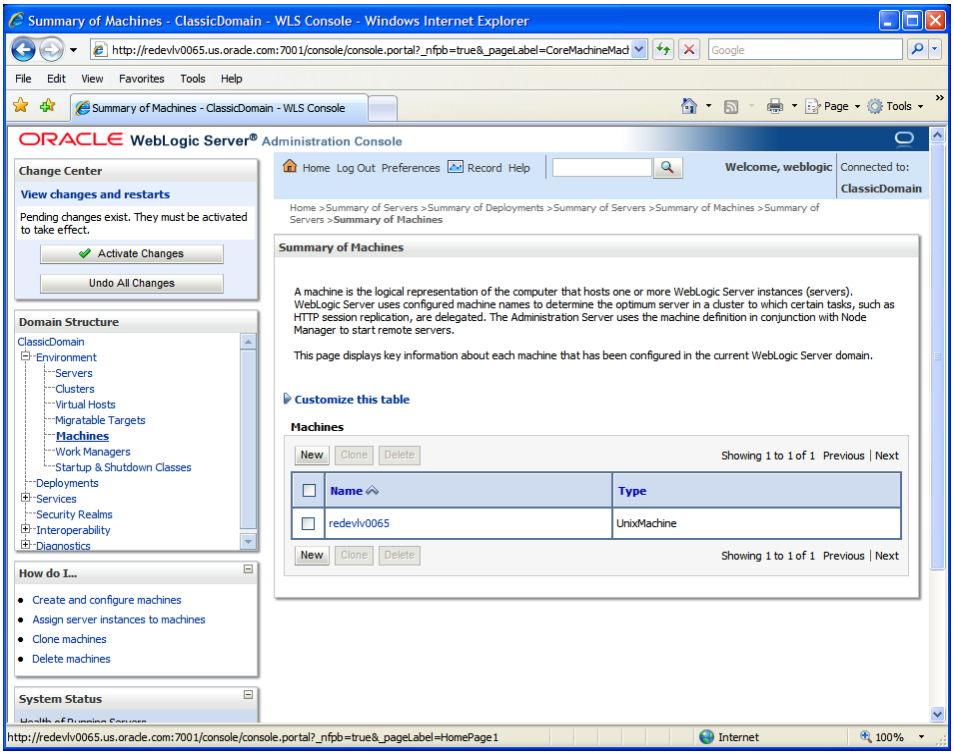
Install Node Manager if it was not created during domain install. Node Manager is required so that the managed servers can be started and stopped through the administration console. Only one node manager is needed per WebLogic installation.

1. Log in to the administration console.
2. Click **Lock & Edit** button and navigate to Environments > Machines.
3. Click **New**.

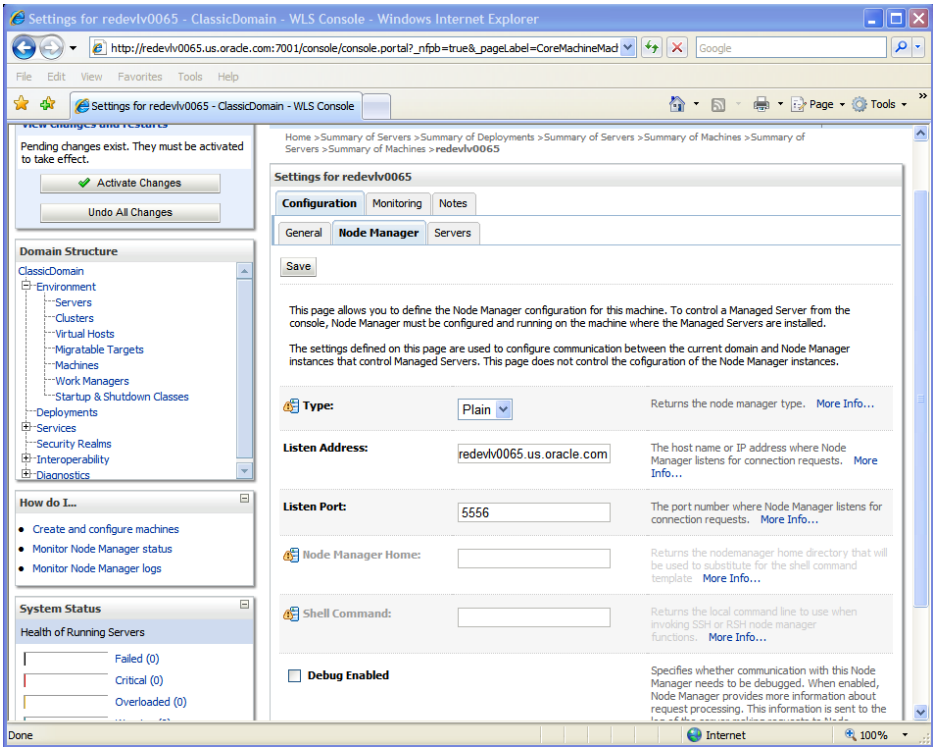


4. Set the following variables:
   - **Name**: Logical machine name
   - **Machine OS**: UNIX
5. Click **OK**.

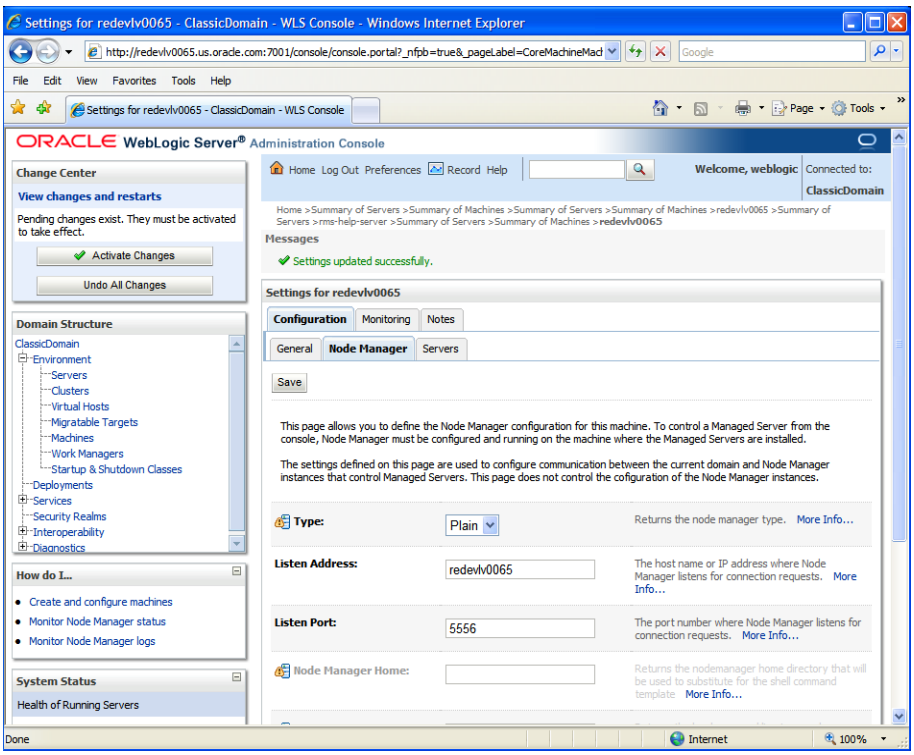
6. Click on the machine created.



7. Click the Node Manager tab and update the details below.
   - **Type:** Plain
   - **Listen Address:** Machine Name (for example, redevlv0065.us.oracle.com)
   - **Listen Port:** Node manager will be assigned a default port (for example, 5556)

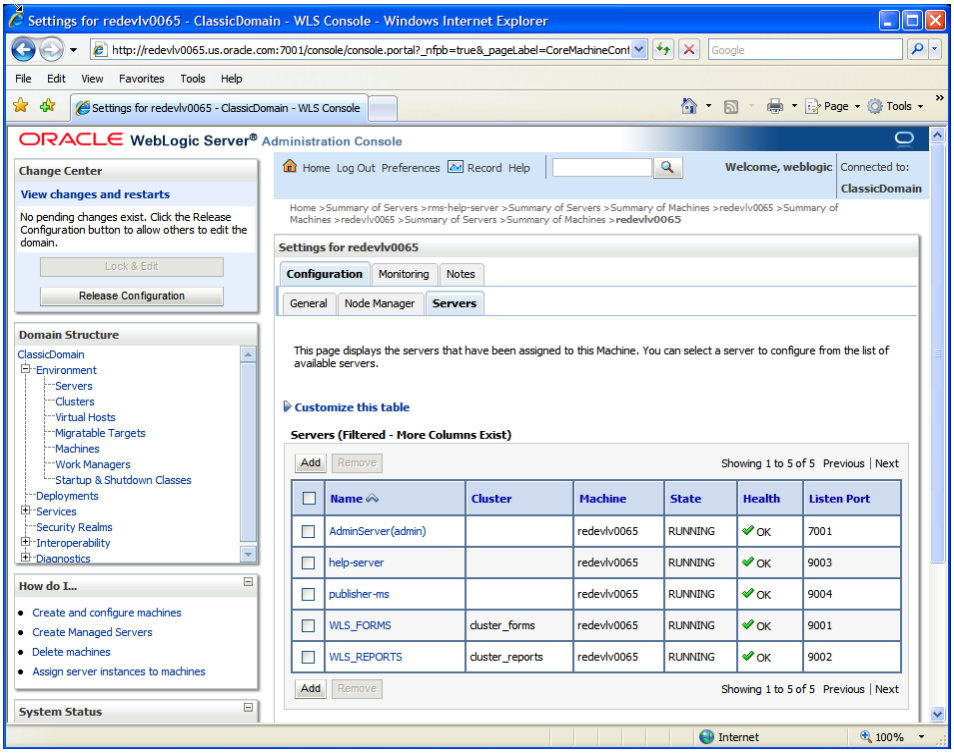


8. Click **Save**.

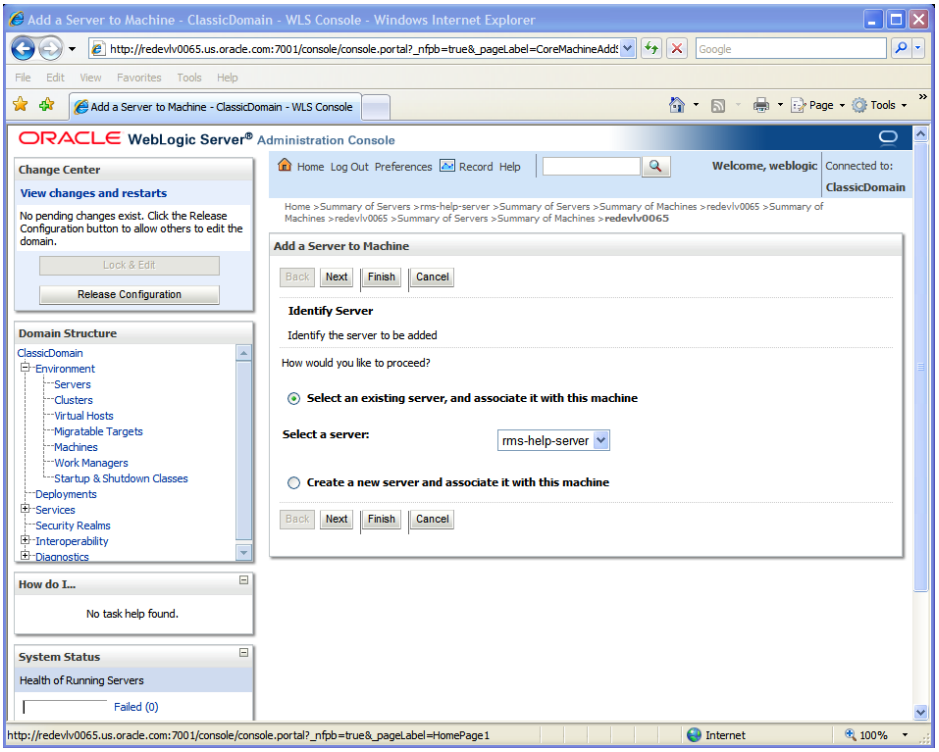


9. Click **Activate Changes**.
10. Click **Lock & Edit**.

11. Navigate to Environments->machines->click on the machine name and select the Servers tab.



12. Click **Add**. Add the managed servers that need to be configured with NodeManager.



13. Set the following variables:
    - Server: name of server previously created (for example, rms-help-server)
14. Click **Next**. Click **Finish**.
15. Click **Activate Changes**.

> **Note:** To activate changes the server must be stopped if it is running:
>
> $WLS_HOME/user_projects/domains/<domain-name>/bin/stopManagedWebLogic.sh <rms-help-server> ${server_name}:${server_port}
>
> Go to each managed server that is being added to the machine and click the Server Start tab. In the Class Path box, add the following:
>
> `<full-path-to-domain>/servers/<managed-server>`
>
> For example:
> `/u00/webadmin/product/10.3.x/WLS/user_projects/domains/ClassicDomain/servers/rms-help-server`

16. After the CLASSPATH changes are finished, click **Save.**
17. Click **Activate Changes**.
18. Start the Nodemanager from the server using the startNodeManager.sh at $WLS_HOME/wlserver_10.3/server/bin.

19. Edit the nodemanager.properties file at the following location with the below values:
    $WLS_HOME/wlserver_10.3/common/nodemanager/nodemanager.properties
    - SecureListener=false
    - StartScriptEnabled=true
    - StartScriptName=startWebLogic.sh.
20. The NodeManager must be restarted after making changes to the nodemanager.properties file.

> **Note:** The nodemanager.properties file is created after NodeManager is started for the first time. It is not available before that point.

Create Staging Directory for RMS Application Patch Files

To create a staging directory for RMS application patch files, complete the following steps.

1. Log on to your application server as a user with read and write access to the WebLogic files.
2. Create a staging directory for the RMS Application patch.
3. Copy the file rms1328apppatch.zip from the RMS 13.2.8 release to staging directory. This will be referred to as APP_PATCH_DIR when installing application software and reports.
4. Change directories to APP_PATCH_DIR and extract the file rms1328apppatch.zip. This creates an app-patch subdirectory under APP_PATCH_DIR.
5. If you do not already have one, create a staging directory for the RMS application installation software or use the same staging directory as created in the database schema step above.
6. Copy the file rms13application.zip from the RMS 13.2 release to staging directory. This will be referred to as STAGING_DIR when installing application software and reports.
7. Change directories to STAGING_DIR and extract the file rms13application.zip. This will create an rms/application subdirectory under STAGING_DIR.

Copy Forms and Library Patch Files

For new environments, the installer can be used to install and compile the forms at the latest patch level using the installer patching utility included with RMS Forms patches. The utility is located under APP_PATCH_DIR/app-patch/patch-util. This utility will accept as input the RMS patch files and add them to the RMS 13.2 Forms installer package. After running this utility, the RMS Forms installer can be used to install the environment, and they will install the latest version of each batch module.

Custom Modules

Custom source can be provided by the user in a folder named APP_PATCH_DIR/app-patch/patch-util/custom. The source code in this folder is applied last, after all patches have been applied.

Run the Installer Patching Utility

To run the installer patching utility, complete the following steps.

1. Set the JAVA_HOME environment variable to point to a JDK which is being used by WebLogic install.
2. Set the ANT_HOME environment variable to point to an Ant installation. There is one included with the RMS installer that can be used for this.

   ANT_HOME=< STAGING_DIR >/rms/application/ant

   export ANT_HOME
3. Change directories to APP_PATCH_DIR/app-patch/patch-util/
4. Modify the patch.properties file. Set the installer.dir and patch.to.version properties.

Variable	Description
installer.dir	The directory where the installer files are located under STAGING_DIR. For example: /opt/rms/application. This directory will contain the install.sh file.
patch.to.version	The version to which you want to patch. For example: 13.2.8

5. Run the patch.sh script. This script will copy the files from each patch from 13.2.1 up to the patch specified in the **patch.to.version** property. These files are copied into the installer package.

Run the RMS Application Installer

Note: See "Appendix: RMS Application Installer Screens" for details on every screen and field in the application installer.

Note: It is necessary to have $ORACLE_HOME/network/admin/tnsnames.ora file configured in this WLS installation. Forms will use this information for connectivity.

A copy tnsnames.ora file must be created for the $ORACLE_INSTANCE/config location. If the file is not copied to this location, forms will not compile correctly.

Note: ORACLE_HOME is the location where Oracle Forms 11gR2 has been installed**.**

ORACLE_INSTANCE is the instance that is created during configuration of Oracle forms 11gR2 and contains the executables to compile.

Note: If you are using AIX 7.1, the **"retail-OCM-wls.zip"** file present in the STAGING_DIR/rms must be renamed or removed, before running the installer. See Appendix: Common Installation Errors for more information.

1. Log on to your application server as a user with read and write access to the WebLogic files.
2. Change directories to STAGING_DIR/rms/application. This directory was created when the rms13application.zip file was expanded under STAGING_DIR.
3. Set and export the following environment variables.

Variable	Description	Example
DOMAIN_HOME	The location where Forms 11.12.0 domain has been installed.	DOMAIN_HOME= /u00/webadmin/product/10.3.x/WLS_Forms/user_projects/domains/ClassicDomain/ export DOMAIN_HOME
WLS_INSTANCE	The name of the managed server that contains Oracle Forms.	WLS_INSTANCE=WLS_FORMS export WLS_INSTANCE
ORACLE_SID	The database/SID where the RMS schema resides.	ORACLE_SID=mydb export ORACLE_SID
NLS_LANG	Locale setting for Oracle database client.	NLS_LANG=AMERICAN_AMERICA.UTF8 export NLS_LANG
JAVA_HOME	Location of Java. Usually the same Java being used by WebLogic. Java choices are: ▪ JDK 1.7.0+ 64 bit	JAVA_HOME= /u00/webadmin/java/jdk1.7 export JAVA_HOME
DISPLAY	Address and port of X server on desktop system of user running install. Required for forms application installer.	DISPLAY=<IP address>:0.0 export DISPLAY

4. To install the RMS application you need to be using an X server such as Exceed and have set the DISPLAY environment variable. The installer does not continue otherwise.
5. Run the install.sh script to start the installer.

Note: Below are the usage details for install.sh. The typical usage for GUI mode is no arguments.

./install.sh [text | silent]

6. The Installer automatically sets additional environment variables based on the values of the environment variables set in step 3. At the end of the preinstall checks it will print out a summary containing these new environment variables:

Example:

MW_HOME=/u00/webadmin/product/10.3.x/WLS_Forms

ORACLE_HOME=/u00/webadmin/product/10.3.x/WLS_Forms/as_1

ORACLE_INSTANCE=/u00/webadmin/product/10.3.x/WLS_Forms/asinst_1

DOMAIN_HOME=/u00/webadmin/product/10.3.x/WLS_Forms/user_projects/domains/ClassicDomain

WLS_INSTANCE=WLS_FORMS

ORACLE_SID=mydb

JAVA_HOME=/u00/webadmin/java/jdk1.7

Verify that these environment variables are correct. If any of them are incorrect, you need to verify that the WebLogic shell scripts that set them are configured properly. Check the following scripts:

```
$DOMAIN_HOME/bin/setDomainEnv.sh
$WEBLOGIC_HOME/wlserver_10.3/common/bin/commEnv.sh
```

Example:

/u00/webadmin/product/10.3.x/WLS_Forms/user_projects/domains/ClassicDomain/bin/setDomainEnv.sh

/u00/webadmin/product/10.3.x/WLS_Forms/wlserver_10.3/common/bin/commEnv.sh

Depending on system resources, a typical installation takes anywhere from 45 minutes to two hours.

The installer asks for an installation directory. This is the destination directory for the RMS files. This directory will be referred to as INSTALL_DIR for the remainder of this chapter. Do not provide an INSTALL_DIR that is located at or underneath STAGING_DIR.

Note: You may see the following warning repeated during installation:

```
     [exec] Warning! One or more of your selected
locales are not available.
     [exec] Please invoke the commands "locale" and
"locale -a" to verify your
     [exec] selections and the available locales.
     [exec]
     [exec] Continuing processing using the "C" locale.
```

Or

```
     [exec] couldn't set locale correctly
```

This warning can be ignored.

7. After the installation is complete, you can check its log file: INSTALL_DIR/base/log/rms.app.install.<timestamp>.log. The INSTALL_DIR/base/error will contain information about possible failed compilations.
8. The installer leaves behind the ant.install.properties file for future reference and repeat installations. This file contains inputs you provided. As a security precaution, make sure that the file has restrictive permissions.

Example: `chmod 600 ant.install.properties`

Note: The object counts performed by the installer may be off due to the patch adding or removing modules. The installer warnings about this can be ignored

9. If during the screens you chose not to have the installer automatically configure WebLogic, after the installation is complete follow the post installation tasks by making backups of the listed files and copying the required files to the specified location.

Example:

```
###############################################################################
##                    WebLogic Configuration Tasks                           ##
###############################################################################
Contact your WebLogic administrator and have them make backups of the following
files:

/u00/webadmin/product/10.3.x/WLS_Forms/user_projects/domains/ClassicDomain/config/
fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/forms/registry/ora
cle/forms/registry/Registry.dat
/u00/webadmin/product/10.3.x/WLS_Forms/user_projects/domains/ClassicDomain/config/
fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/formsweb.cfg

Have the WebLogic administrator stop WLS_FORMS and ohs1,
copy everything in /home/oretail/rms132/install/post
to /u00/webadmin/product/10.3.x/WLS_Forms to update the files
and then start WLS_FORMS and ohs1
for the changes to take effect.

example: cp -R * /u00/webadmin/product/10.3.x/WLS_Forms
```

Verifying FORMS Configuration file details:

```
Once the installation is done, the formsweb.cfg located at
<FORMSDOMAIN_HOME>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2
/config should have the RMS configured as an example below:[webutil]
WebUtilArchive=frmwebutil.jar,jacob.jar
WebUtilLogging=off
WebUtilLoggingDetail=normal
WebUtilErrorMode=Alert
WebUtilDispatchMonitorInterval=5
WebUtilTrustInternal=true
WebUtilMaxTransferSize=16384
baseHTML=webutilbase.htm
baseHTMLjpi=webutiljpi.htm
archive=frmall.jar
lookAndFeel=oracle

[rms132]
envfile=./rms132.env
width=950
height=685
separateFrame=true
form=rtkstrt.fmx
lookAndFeel=Oracle
colorScheme=swan
archive=frmall.jar,icons.jar
imageBase=codebase
```

```
#userid=<dbusername>/<dbpassword>@<database_sid>
#http://<hostname>:<port>/forms/frmservlet?config=rms132
```

If Oracle Single Sign-On is to be used with RMS, do the following.

- Set ssoMode to true.

If Resource Access Descriptors are allowed to be dynamically created,

```
Set ssoDynamicResourceCreate to true.
```

```
rms132.env mentioned above should have the following variables set:
NLS_DATE_FORMAT=DD-MON-RR
NLS_LANG=AMERICAN_AMERICA.UTF8
FORMS_REJECT_GO_DISABLED_ITEM=FALSE
```

Resolving Errors Encountered During Application Installation

In the event a form or menu does not compile, go to INSTALL_DIR/base/error and see which objects did not compile. To try and manually recompile the object run INSTALL_DIR/base/forms.profile and run the following command:

```
# frmcmp.sh userid=$UP module_type=form module=FORM_OR_MENU
```

You can also safely rerun the installer to see if the form compiles.

> **Note:** If you rerun the installer, and choose to check the box **Configure WebLogic** in the installer screens, you may need to clean up duplicate entries in the WebLogic formsweb.cfg file.

Clustered Installations – Post-Installation Steps

If you are installing the RMS application to a clustered Oracle Application Server environment, there are some extra steps you need to take to complete the installation. In these instructions, the application server node with the ORACLE_HOME you used for the RMS application installer is referred to as *master node*. All other nodes are referred to as *remote nodes*.

To complete the RMS forms application install, the installer provided new versions of formsweb.cfg and the newly-created env files for the new RMS installation. The entries added to formsweb.cfg and env files for these new environments should be copied from the master node to the remote nodes.

> **Note:** The newly created env files will have a change to the FORMS_PATH variable as well as entries appended to the end of the file.

> **Note:** Do **NOT** copy the entire file from one node to another. Only copy the RMS entries modified in these files by the installer. There is node-specific information in this file that is different between ORACLE_HOME installations.

Oracle Configuration Manager

The Oracle Retail OCM Installer packaged with this release installs the latest version of OCM.

The following document is available through My Oracle Support (formerly MetaLink). Access My Oracle Support at the following URL:

`https://support.oracle.com`

Oracle Configuration Manager Installer Guide (Doc ID: 1071030.1)

This guide describes the procedures and interface of the Oracle Retail Oracle Configuration Manager Installer that a retailer runs at the beginning of the installation process.

Note: OCM is not supported on AIX 7.1.

OCM Documentation Link

`http://www.oracle.com/technology/documentation/ocm.html`

RMS Reports Copied by the Application Installer

The application installer copies RMS report files to INSTALL_DIR/base/reports. These files should be installed into BI Publisher as documented in the chapter, RMS Reports Installation—Patch.

Test the RMS Application

Oracle Retail provides test cases that allow you to smoke test your installation. See the My Oracle Support document, *Oracle Retail Merchandising Installation Test Cases* (ID 1277131.1).

Option 2: Compile RMS Toolset and Forms Directly

To compile the RMS toolset and forms directly requires the following steps.

Create Staging Directory for RMS Application Patch Files

To create a staging directory for RMS application patch files, complete the following steps.

1. Log on to your application server as a user with read and write access to the WebLogic files.
2. Create a staging directory for the RMS Application patch.
3. Copy the file rms1328apppatch.zip from the RMS 13.2.8 release to staging directory. This will be referred to as APP_PATCH_DIR when installing application software and reports.
4. Change directories to APP_PATCH_DIR and extract the file rms1328apppatch.zip. This creates an app-patch subdirectory under APP_PATCH_DIR.

Set Environment Variables

Note:

INSTALL_DIR is the location where RMS 13 forms were installed.

ORACLE_HOME is the location where Oracle Forms 11gR1 has been installed.

ORACLE_INSTANCE is the instance that is created during configuration of Oracle forms 11gR1 and contains the executables to compile.

Make sure the following variables are set. The RMS 13.2 forms installer should have created a forms.profile file located at INSTALL_DIR/base/forms.profile. This profile script can be used to set all of the environment variables listed below.

Example: cd <INSTALL_DIR>/base

. ./forms.profile

Variables set by forms.profile:

- **All OS Platforms**
  - MMHOME=INSTALL_DIR/base
  - ORACLE_HOME=/path/to/WebLogic/as_1
  - ORACLE_INSTANCE=/path/to/WebLogic/asinst_1
  - ORACLE_SID= The Oracle Sid for the RMS database
  - UP=/@< Schema Owner Wallet Alias >
  - TNS_ADMIN=/path/to/wallet/files/dir/
  - NLS_LANG=AMERICAN_AMERICA.UTF8
  - DISPLAY=<IP address of X server>:0.0
  - PATH=$ORACLE_HOME/bin:$ORACLE_HOME/opmn/bin:$ORACLE_HOME/dcm/bin:INSTALL_DIR/base/forms_scripts:$PATH
  - CLASSPATH=$ORACLE_HOME/jlib/importer:$ORACLE_HOME/jlib/debugger.jar:$ORACLE_HOME/jlib/utj.jar:$ORACLE_HOME/jlib/ewt3.jar:$ORACLE_HOME/jlib/share.jar:$ORACLE_HOME/jlib/dfc.jar:$ORACLE_HOME/jlib/help4.jar:$ORACLE_HOME/jlib/oracle_ice.jar:$ORACLE_HOME/jlib/jewt4.jar
  - FORMS_BUILDER_CLASSPATH=$CLASSPATH
  - FORMS_PATH=INSTALL_DIR/base/toolset/bin:INSTALL_DIR/rms/forms/bin:$ORACLE_HOME/forms
  - TK_UNKNOWN=$ORACLE_INSTANCE/config/FRComponent/frcommon/guicommon/tk/admin
  - PATH=$ORACLE_INSTANCE/bin:$PATH

Note: See "Appendix: Setting Up Password Stores with Oracle Wallet" for how to set up database wallet.

Note: Verify that TNS is set up correctly by using the UP variable to successfully log in to the RMS 13 schema.

Example: /u00/oracle> sqlplus $UP

- **AIX**
  - LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/lib32:$ORACLE_HOME/jdk/jre/lib
  - LIBPATH=$LD_LIBRARY_PATH
- **Linux**
  - LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/lib32:$ORACLE_HOME/jdk/jre/lib

Copy Forms and Library Patch Files

To copy forms and library patch files, complete the following steps.

1. Make a backup copy of the existing INSTALL_DIR/base/toolset and INSTALL_DIR/base/forms directories.
2. For the version number of the patch you are installing, navigate to the corresponding directory in APP_PATCH_DIR and copy the contents of the following directories into INSTALL_DIR. Depending on the patch, these directories may not exist. Copy the contents of APP_PATCH_DIR/app-patch/<version>/base/toolset into INSTALL _DIR/base/toolset, APP_PATCH_DIR/app-patch/<version>/base/forms into INSTALL _DIR/base/forms, and APP_PATCH_DIR/app-patch/<version>/base/forms_scripts into INSTALL _DIR/base/forms_scripts.

Delete Obsolete Files

The following forms and menus are obsolete as of 13.2.8 and should be deleted from INSTALL_DIR.

```
INSTALL_DIR/base/forms/src/company.mmb
INSTALL_DIR/base/forms/src/ribapierr.fmb
INSTALL_DIR/base/forms/bin/company.mmx
INSTALL_DIR/base/forms/bin/ribapierr.fmx
```

Install RMS Toolset

To install the RMS toolset, complete the following steps.

1. Copy all libraries (.pll files) in the INSTALL _DIR/base/toolset/src directory to the INSTALL _DIR/base/toolset/bin directory.
2. Change directories to INSTALL _DIR/base/toolset/bin.
3. Verify that the PATH variable contains the path INSTALL_DIR/base/forms_scripts. The forms.profile script should have set this up already.
4. Run toolset.pll.sh to compile all Toolset .pll's.

> **Note:** If the toolset.pll.sh script is not used and the libraries are compiled individually, then they must be compiled in the following order (which is noted in the toolset.pll.sh script):

- messge45.pll
- ariiflib.pll
- stand45.pll
- calend45.pll
- find45.pll
- item45.pll

- tools45.pll
- mblock45.pll
- mview45.pll
- nav45.pll
- work45.pll
- itnumtype.pll
- hierfilter.pll
- rmslib.pll
- cflex.pll

5. Check to make sure that each .pll file has a corresponding .plx (to ensure that all .pll's compiled successfully). Remove all newly created .plx files.
6. Copy all forms (*.fmb files) in the INSTALL_DIR/base/toolset/src directory to the INSTALL_DIR/base/toolset/bin directory.
7. Run forms.fm_fmb.sh (in INSTALL_DIR/base/toolset/bin) to compile the Toolset reference forms.
8. Remove all newly created fm_*.fmx files (reference forms should not have executable files).
9. Run forms.fmb.sh (in INSTALL_DIR/base/toolset/bin) to generate Toolset runtime forms – .fmx's.
10. Check to make sure that each non-reference form .fmb file has a corresponding .fmx file.
11. Remove all non-reference form forms from INSTALL_DIR/base/toolset/bin; the following syntax will leave all reference forms (fm_*.fmb) in the bin directory, while removing all other forms:

    ```
    > for PROG in `ls *.fmb | grep -v fm_`
    > do PROGNAME=`echo $PROG`
    > rm $PROGNAME
    > done
    ```

12. Copy all menus (*.mmb files) in the INSTALL_DIR/base/toolset/src directory to the INSTALL_DIR/base/toolset/bin directory.
13. Run menus.mmb.sh (in INSTALL_DIR/base/toolset/bin) to generate Toolset runtime menus – .mmx's and make sure that each .mmb file has a corresponding .mmx file.
14. Remove all .mmb files from INSTALL_DIR/base/forms/bin.

> **Note:** .err files may be created by the compilation scripts above. These files are logs of the compilation process and can be removed.

Install RMS Forms

To install RMS forms, complete the following steps.

1. Copy all libraries (.pll files) in the INSTALL_DIR/base/forms/src directory to the directories to the INSTALL_DIR/base/forms/bin directory.
2. Change directories to INSTALL_DIR/base/forms/bin.
3. Run forms.pll.sh to compile all RMS .pll's.

4. Check to make sure that each .pll file has a corresponding .plx (to ensure that all .pll's compiled successfully). Remove all newly created .plx files.
5. Copy all forms (*.fmb files) in the INSTALL_DIR/base/forms/src directory to the INSTALL_DIR/base/forms/bin directory.
6. Run forms.fm_fmb.sh (in INSTALL_DIR/base/forms/bin) to compile the RMS reference forms.
7. Remove all newly created fm_*.fmx files (reference forms should not have executable files).
8. Run forms.fmb.sh (in INSTALL_DIR/base/forms/bin) to generate RMS runtime forms – .fmx's.
9. Check to make sure that each non-reference form .fmb file has a corresponding .fmx file.
10. Remove all non-reference form forms from INSTALL_DIR/base/forms/bin; the following syntax will leave all reference forms (fm_*.fmb) in the bin directory, while removing all other forms:

```
> for PROG in `ls *.fmb | grep -v fm_`
> do PROGNAME=`echo $PROG`
> rm $PROGNAME
> done
```

11. Copy all menus (*.mmb files) in the INSTALL_DIR/base/forms/src directory to the INSTALL_DIR/base/forms/bin directory.
12. Run menus.mmb.sh (in INSTALL_DIR/base/forms/bin) to generate RMS runtime menus – .mmx's and make sure that each .mmb file has a corresponding .mmx file.
13. Remove all .mmb files from INSTALL_DIR/base/forms/bin.

> **Note:** .err files may be created by the compilation scripts above. These files are logs of the compilation process and can be removed.

Install Helpfile

To install the helpfile, complete the following steps.

1. Log into the WebLogic instance to which online help will be installed.
2. Create a server if necessary. In this example, rms-help-server is being used.
3. Select Deployments.
4. If there is an existing rms-help deployment, it must be stopped and deleted. Check the box next to rms-help. Click **Stop->Force Stop Now**. Click **Yes** on the next screen. Check the box next to rms-help. Click **Delete**. On the next screen, click **Yes**. When finished deleting, bounce the rms-help-server.
5. Select Deployments.
6. Click **Install**.
7. Click in the Path: box and enter APP_PATCH_DIR/app-patch/13.2.8/online-help/rms-help.ear, the ear file that will be deployed.
8. Leave **Install this deployment as an application selected**. Click **Next**.
9. Select the rms-help-server created in Step 2. Click **Next**.
10. Leave rms-help for the application name. Click **Next**.
11. Verify settings. Click **Finish**.
12. Select Deployments.
13. Check the box next to rms-help and click Start -> Servicing all Requests
14. Click **Yes**.
15. In the database, set the WEBHELP_SERVER column in the LANG table for the RMS schema owner to point to the RMS help server and port. For the example, it is set to http://redevlv0065.us.oracle.com:17003.

Install rms-icons.jar

If the patch contains a new rms-icons.jar, follow these steps to install it:

1. Copy the rms-icons.jar from APP_PATCH_DIR/app-patch/<version>/web_html/ and overwrite the one in $ORACLE_HOME/as_1/forms/java/
2. Bounce the WebLogic managed server that contains Forms (for example: WL_FORMS).

7

RMS Reports Installation—Patch

Starting with RMS 13.2.6, reports now support ONLY BI Publisher 11g.

Upgrading from BI Publisher 10g to 11g is not trivial. Among other things, the BI Publisher report program in 10g is the <report_name>.xdo file. In 11g, this <report_name>.xdo report file gets split into two new folders, a <report_name>.xdo folder along with a <report_name>.xdm folder. Both of these two new folders have report files within them. Your BI Publisher 10g reports programs will not work without a change in BI Publisher 11g.

> **Note:** If BI Publisher application 11g is already deployed to a BI Publisher managed server in WebLogic, you can directly go to the "Manually Copy Reports to Install Directory" section. If not, continue with the "BI Server Component Installation Tasks".

BI Server Component Installation Tasks

Oracle BI Publisher is used as the main RMS, RWMS, REIM, and SIM reporting engine and can be used in conjunction with external printing solutions like label printing. This section describes the installation of Oracle BI Publisher as a server application within WebLogic 10.3.6. One deployment of BI Publisher can be used for any of the RMS, RWMS, REIM, and SIM reports.

If you are installing BI Publisher as a part the Oracle BI EE suite(which you will if installing BI Publisher 11g), refer to the appropriate Fusion Middleware guides for the installation of the product in a WebLogic server environment.

BI Publisher 11g Installation Process Overview

Installing the BI Publisher server as a standalone web application in a WebLogic server involves the following tasks:

1. Run RCU to create BI Publisher related database schemas and other db objects.
2. Install Oracle BI EE under an existing WebLogic Server (WLS) 10.3.6 and choose "software only install".
3. Configure Oracle BI EE, create default bifoundation_domain and configure component "Business Intelligence Publisher" only.
4. Select the BIPlatform schema for update of the ORACLE 11.2.0.4 DB
5. Configure ports and document and test the URL's that are created.

The following post-installation tasks are involved once BI Publisher has been installed:

1. Configure the BI Publisher repository. Set security model, add users, assign roles, add reports, add printers, set repository path, set data source, etc.
2. Set up and copy the RMS BI Publisher Report Templates produced for RMS.
3. Set up for the RMS application specific configuration files to integrate BI Publisher with the RMS online app.

BI Publisher 11g– Install Oracle BI EE 11g

1. Run the Repository Creation Utility to create the BI Publisher-related database schemas and other database objects. Create the BIPlatform schema into an existing ORACLE 11.2.0.4 DB

   Note: Download Repository Creation Utility software from http://www.oracle.com/technetwork/middleware/bi-enterprise-edition/downloads/bi-downloads-1525270.html. Install it on your desktop

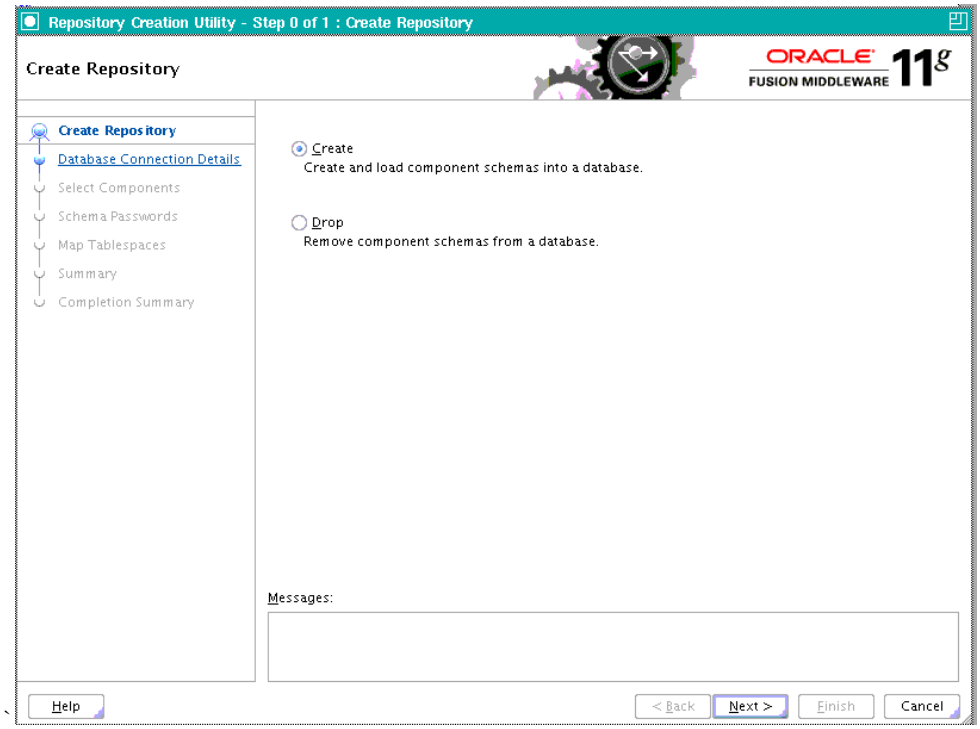
2. Export your DISPLAY

   Example: export DISPLAY=10.141.10.110:0.0

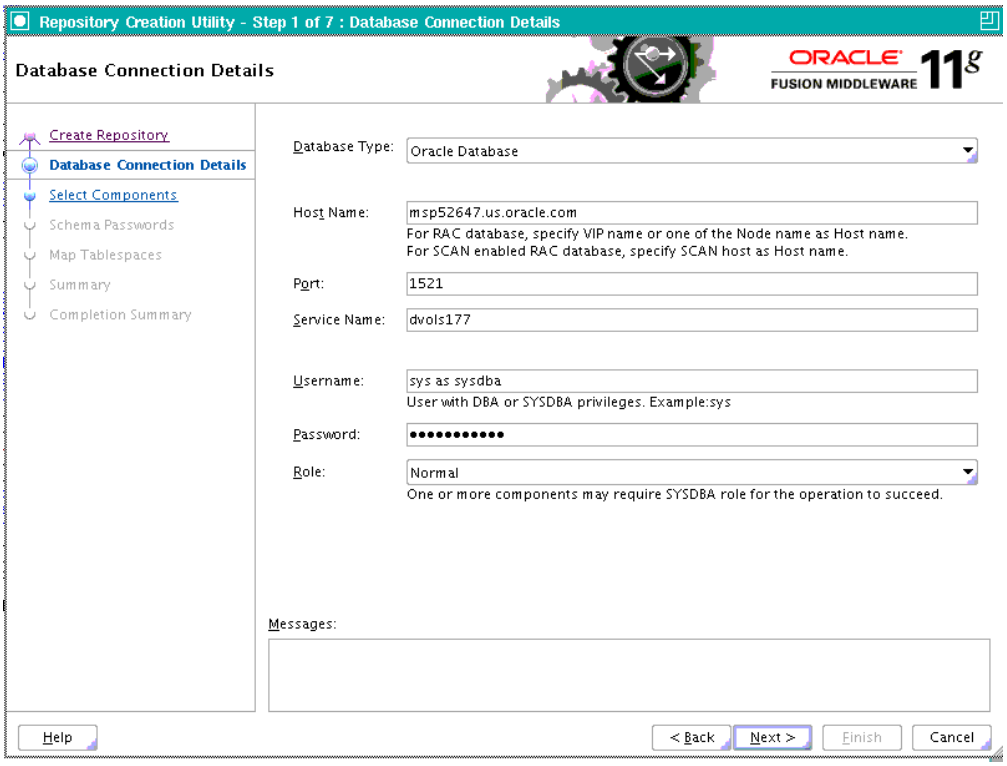
3. Go to $RCU_HOME/bin.

   Example: /linux/x86_64/ofm_11g/RCU_11.1.1.7/rcuHome/bin>

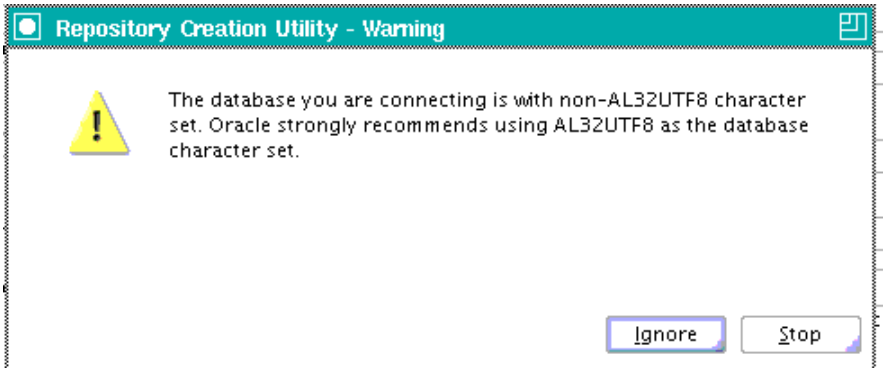
   `Start RCU:  ./rcu`



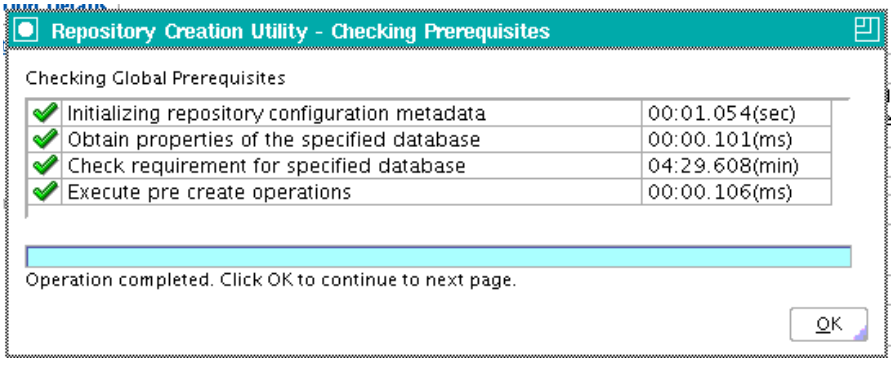
4. Launch Oracle BI EE RCU Repository Creation Utility to create the Oracle BI EE schemas need for the Oracle BI EE BI Publisher installation. On this screen select “Create Repository”.



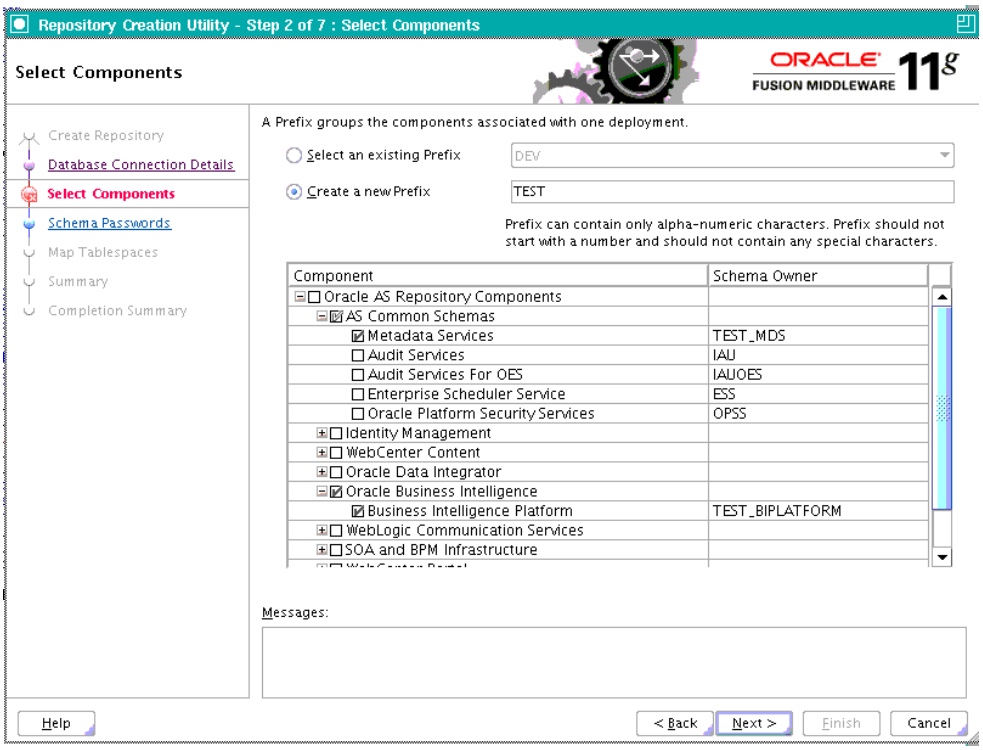
5. On the Database Connection Details screen, enter your Oracle Database information.



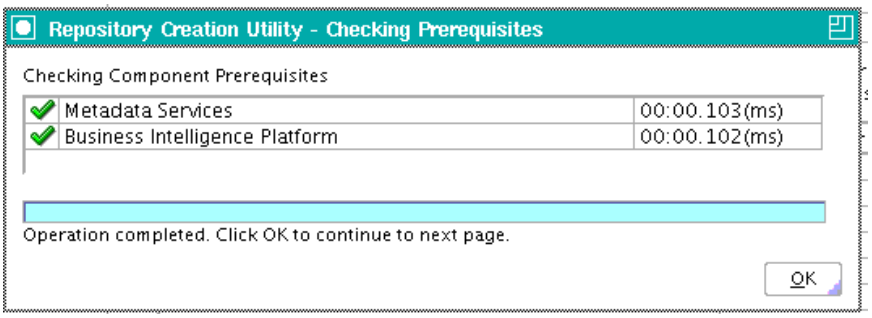
6. Click Ignore and proceed.



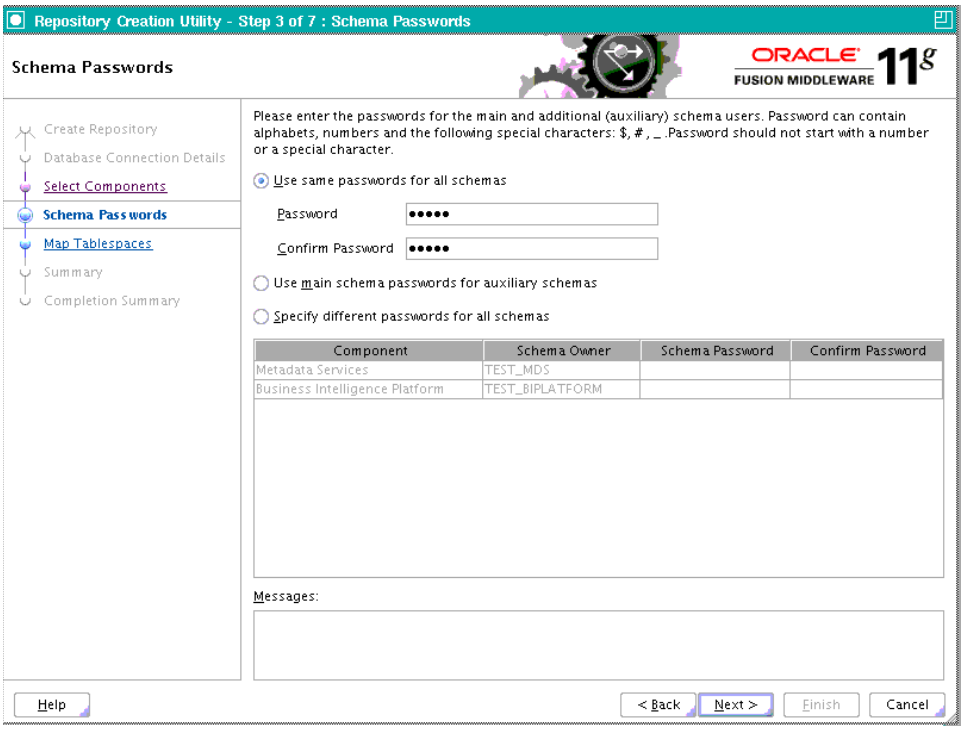
7. Click OK.



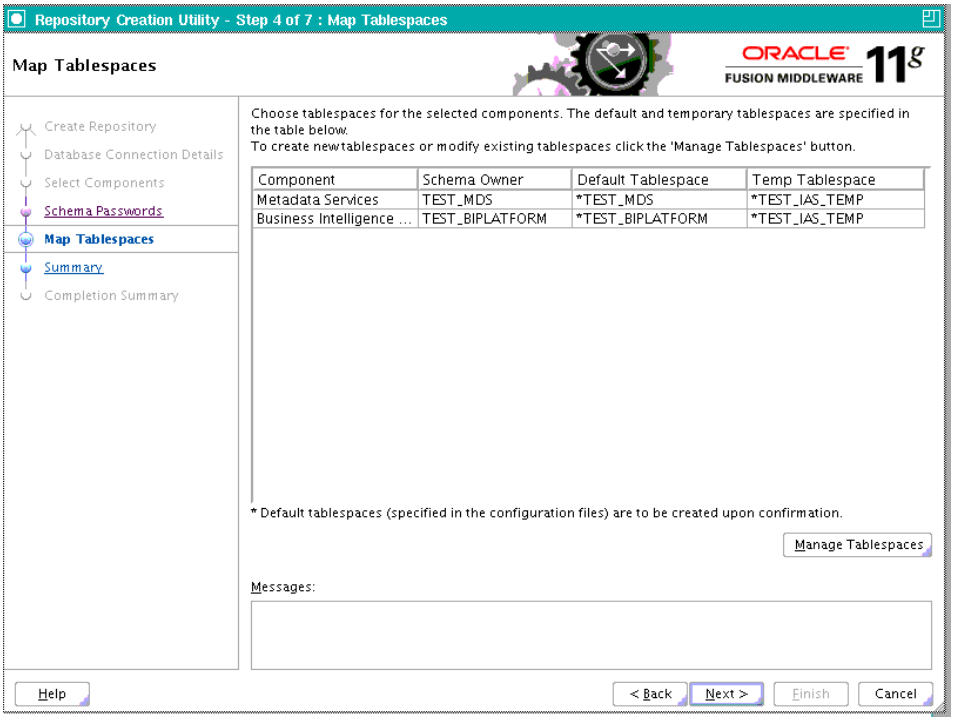
8. On the Select Components screen, select "Oracle Business Intelligence" check box.



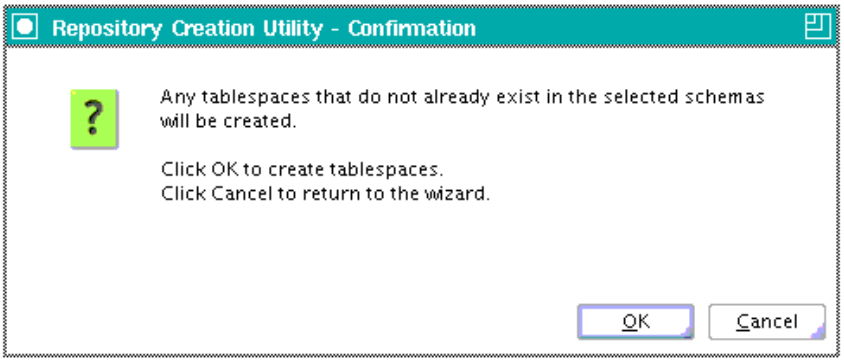
9. Click OK.



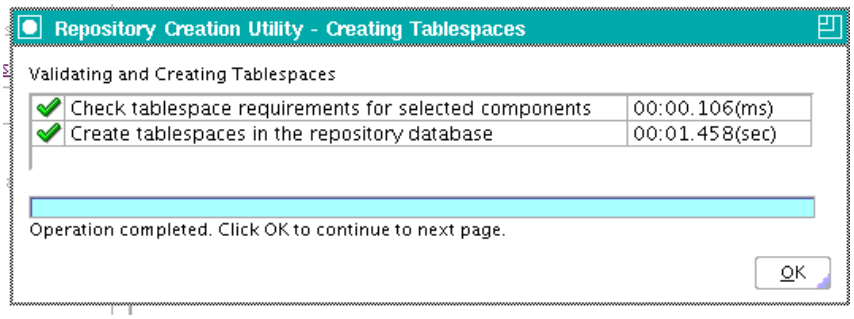
10. Enter the password. Make a note of this password as this password will be used while creating datasources.



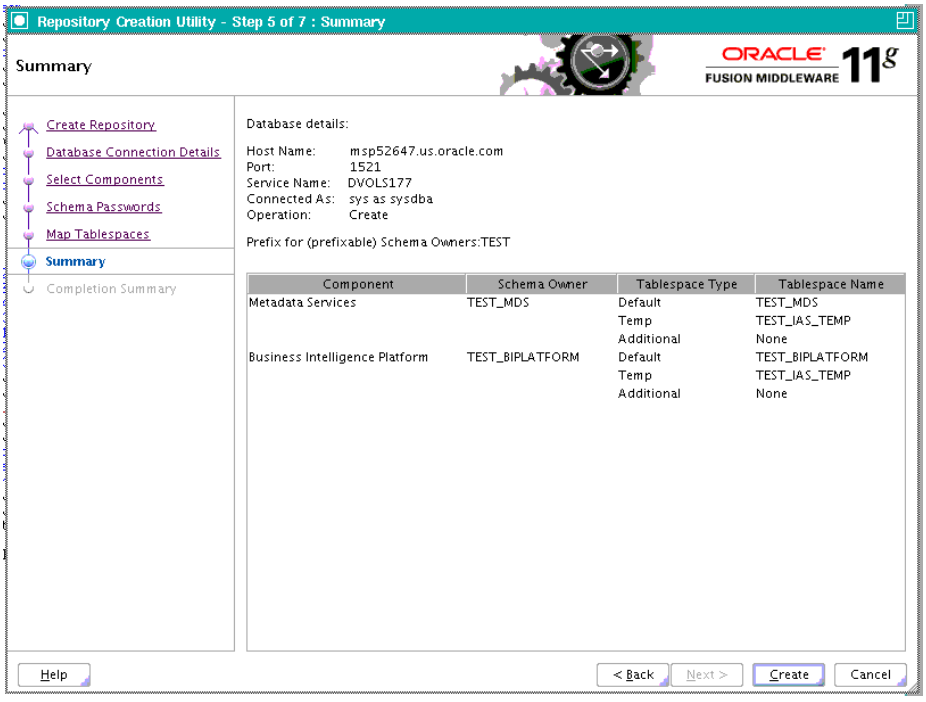
11. Click Next.



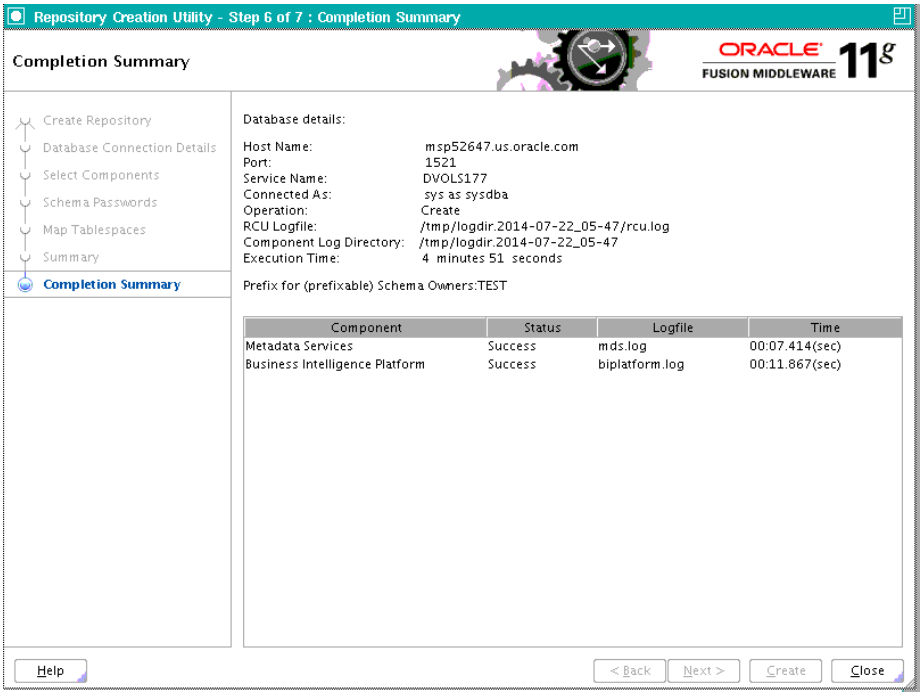
12. Click OK.



13. Tablespaces are created. Click OK.

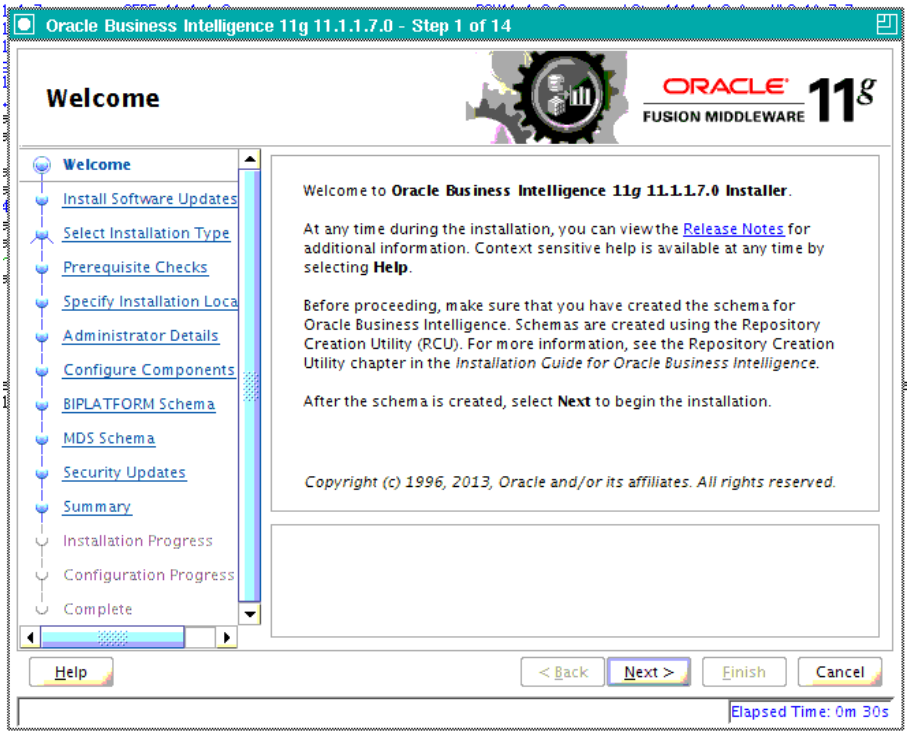


14. Click Create.

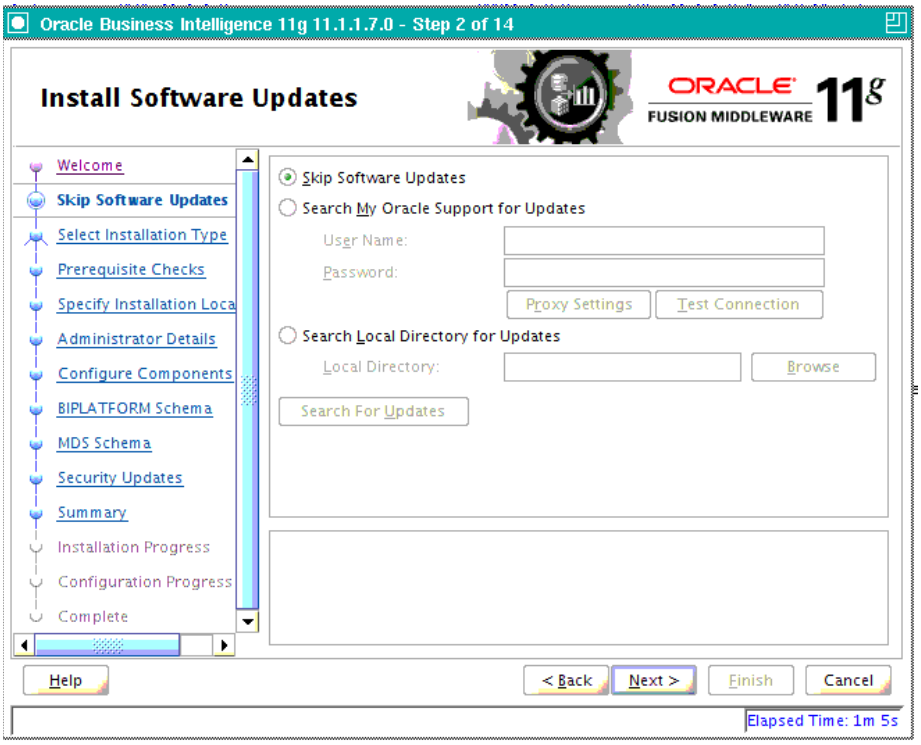


15. The Summary of the Components created by the RCU tool is displayed.
16. Install a new instance of WebLogic Server 10.3.6 or use an existing one. Having one WebLogic Server for Oracle BI EE-BI Publisher 11g related items is recommended.
17. Install Oracle BI EE and select “Software Only Install”. You launch Oracle BI EE by going to OBIEE_INSTALL/obiee11.1.1.7/bishiphome/Disk1 and entering:

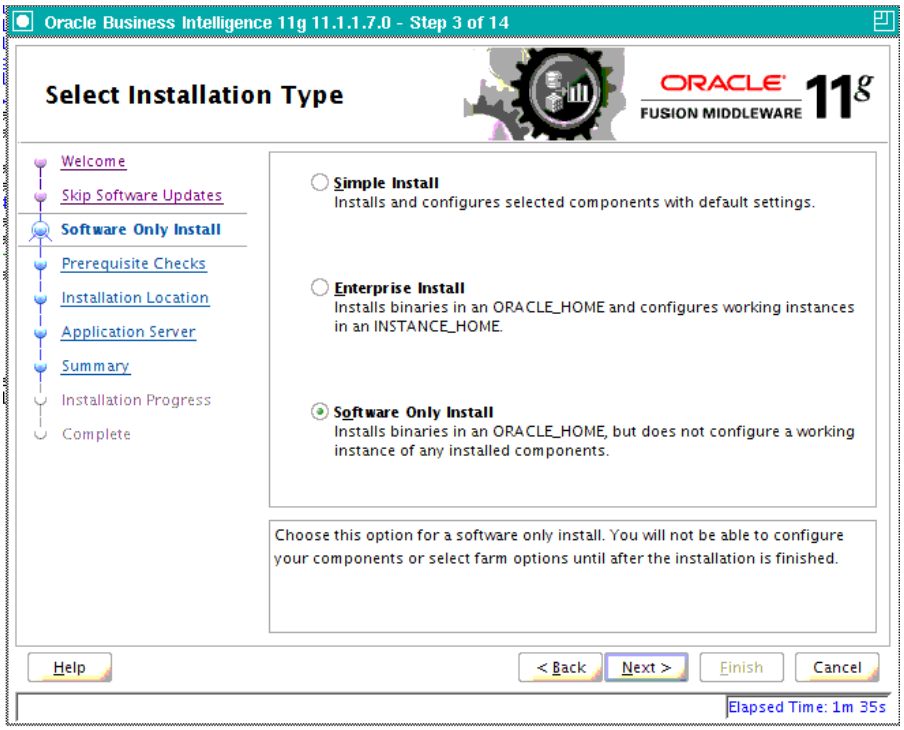
    ./runInstaller
18. Configure Oracle BI EE, create default bifoundation_domain and configure component “Business Intelligence Publisher” only.



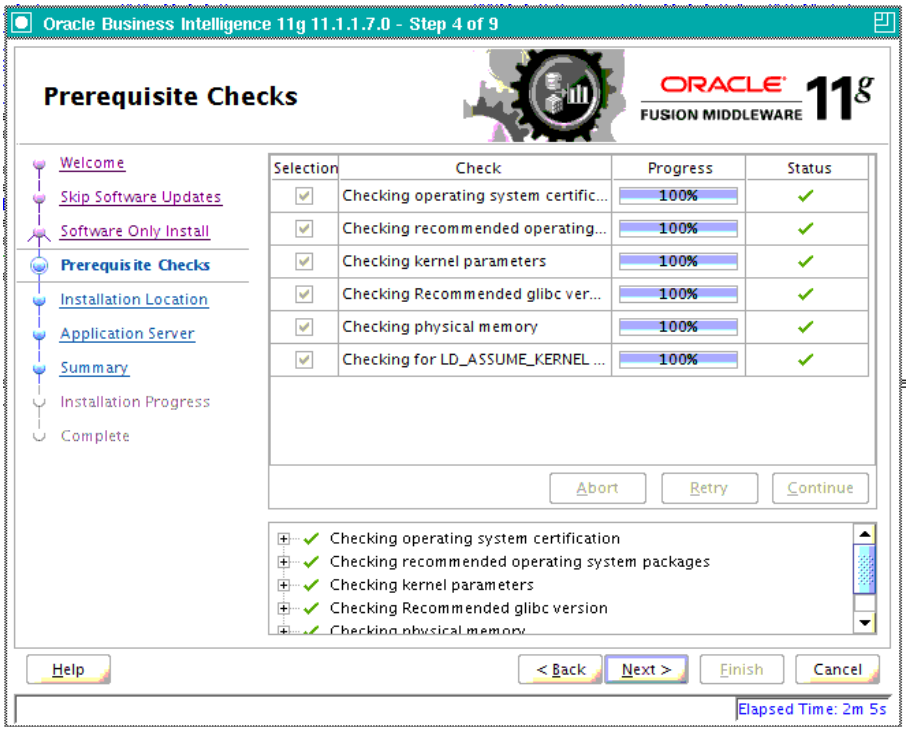
19. Click Next.



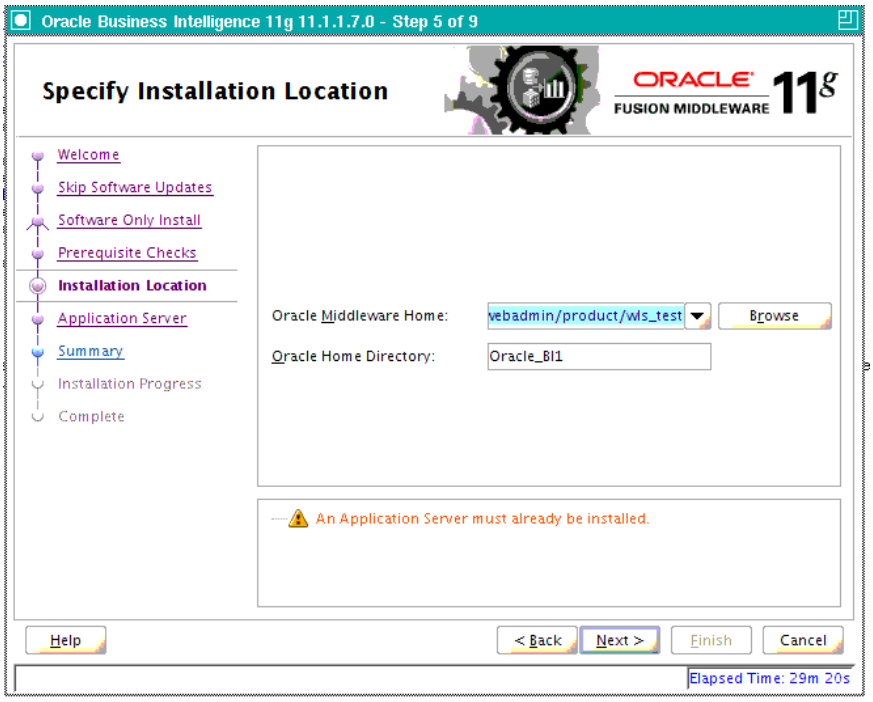
20. Select 'Skip Software Updates' and Click Next.



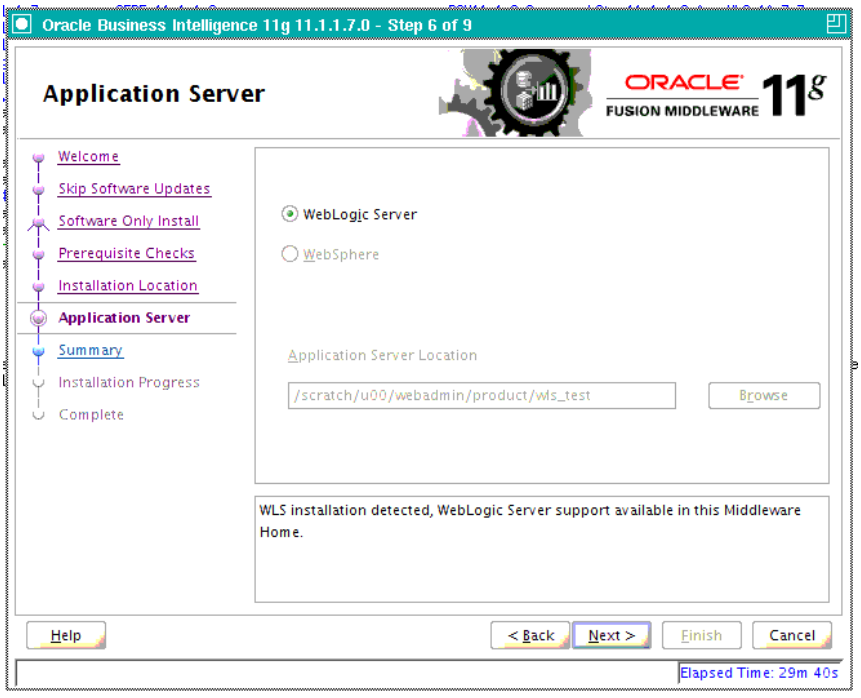
21. Select "Software Only Install" and Click Next.



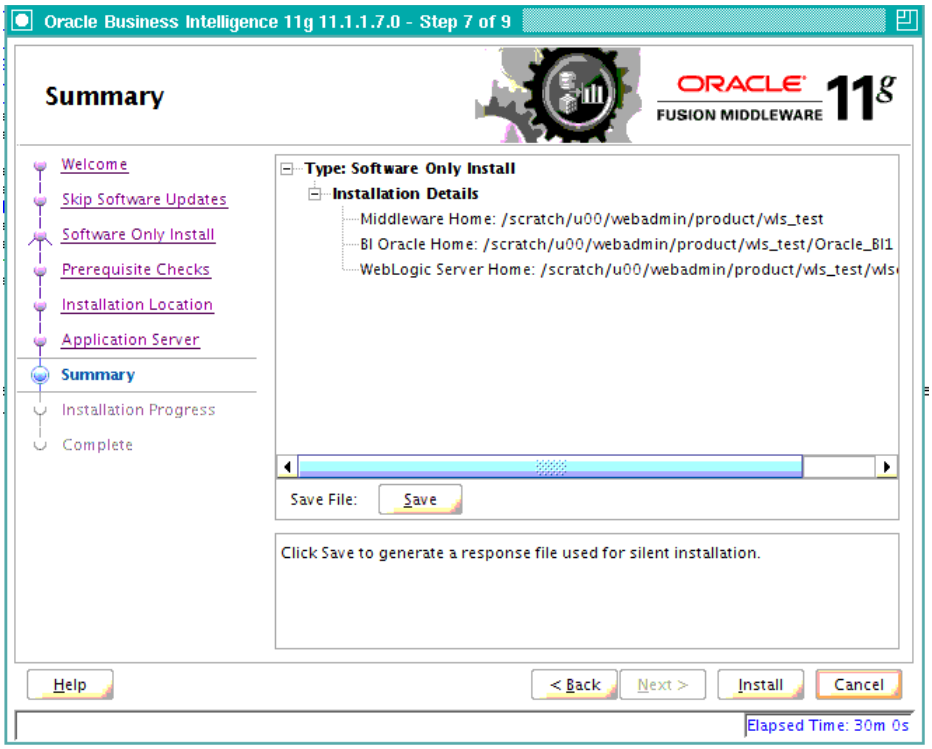
22. Click Next once all the prerequisite checks are done.



23. Provide details of the Oracle Middleware Home. Click Next

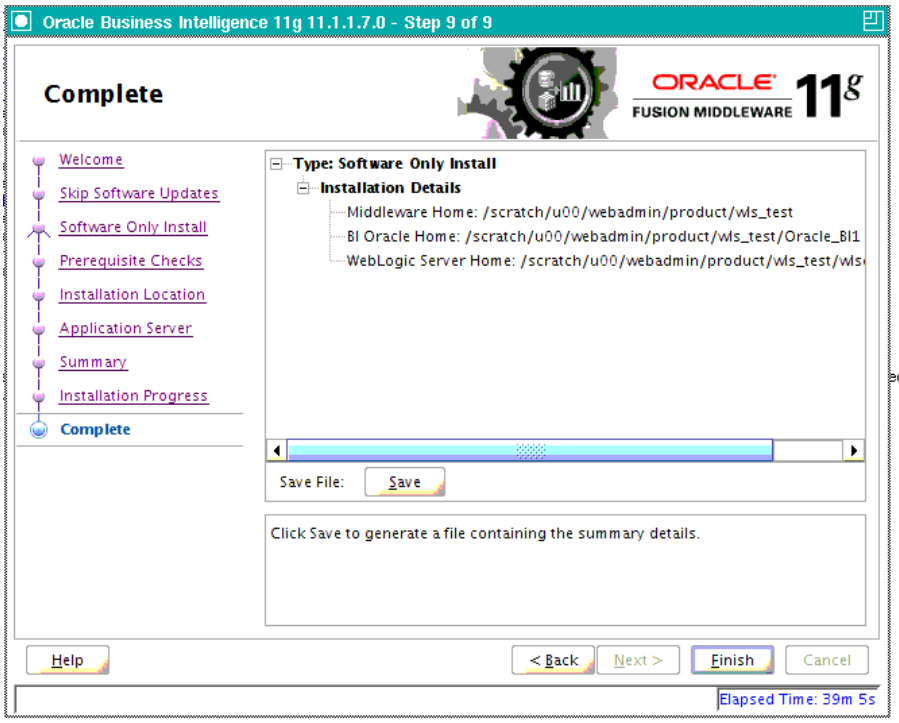


24. Select Weblogic Server and Click Next



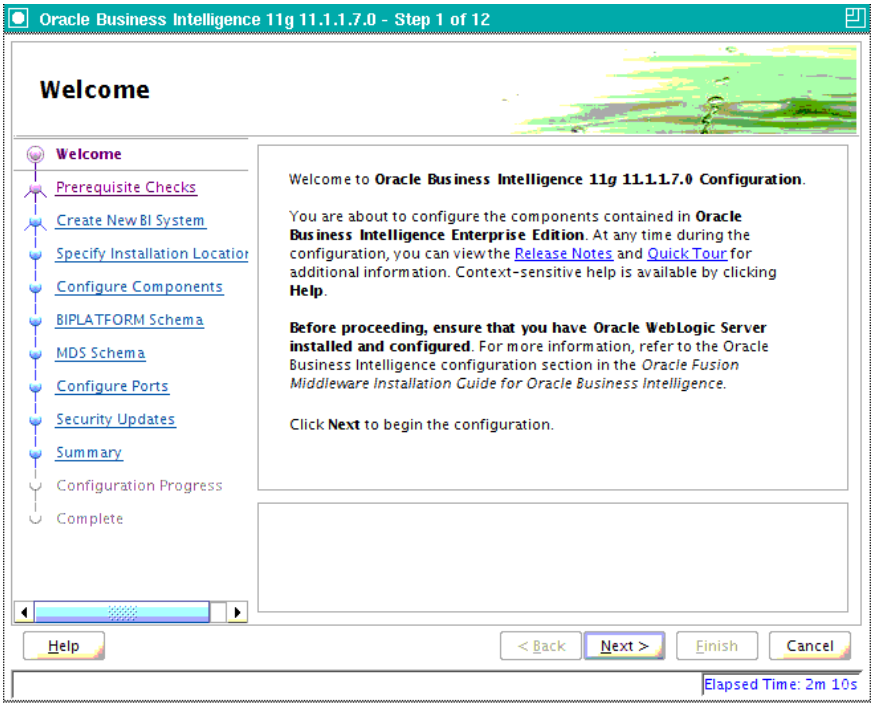
25. Click Install

26. Clcik Next.

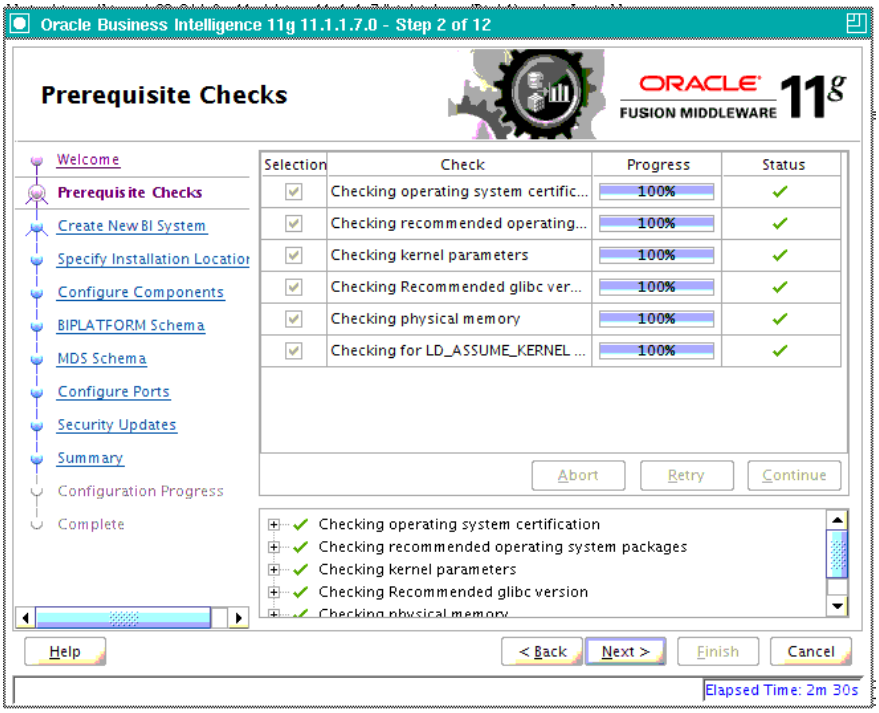


27. Installation of OBIEE is done. Save this screen for Installation details. Run config.h to create bi_foundation domain. The script can be found under

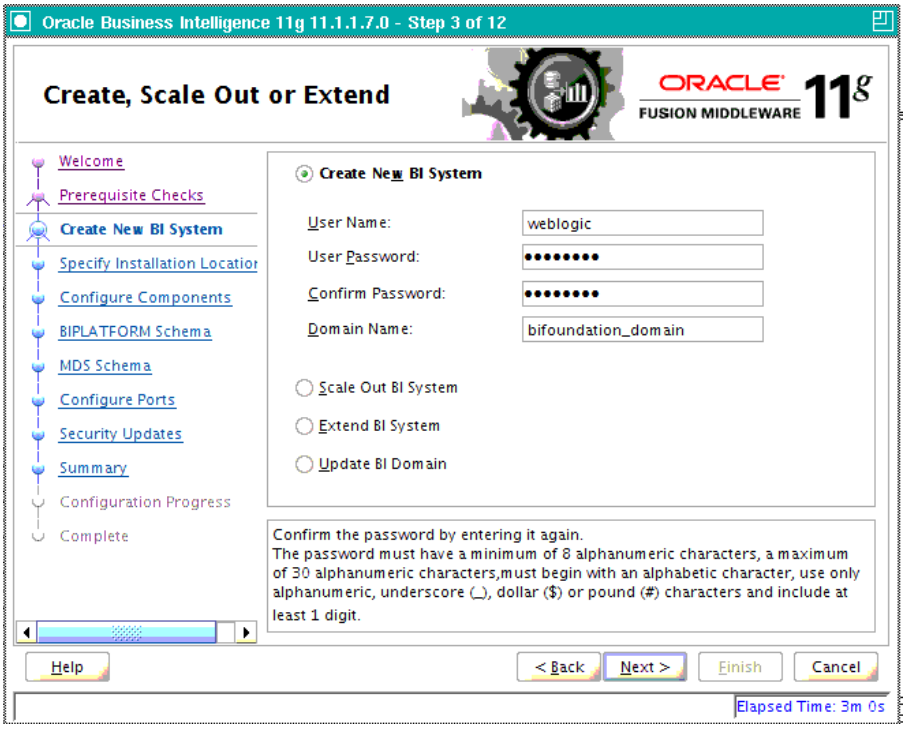
    $WEBLOGIC_DOMAIN_HOME/Oracle_BI1/common/bin



28. Click Next.

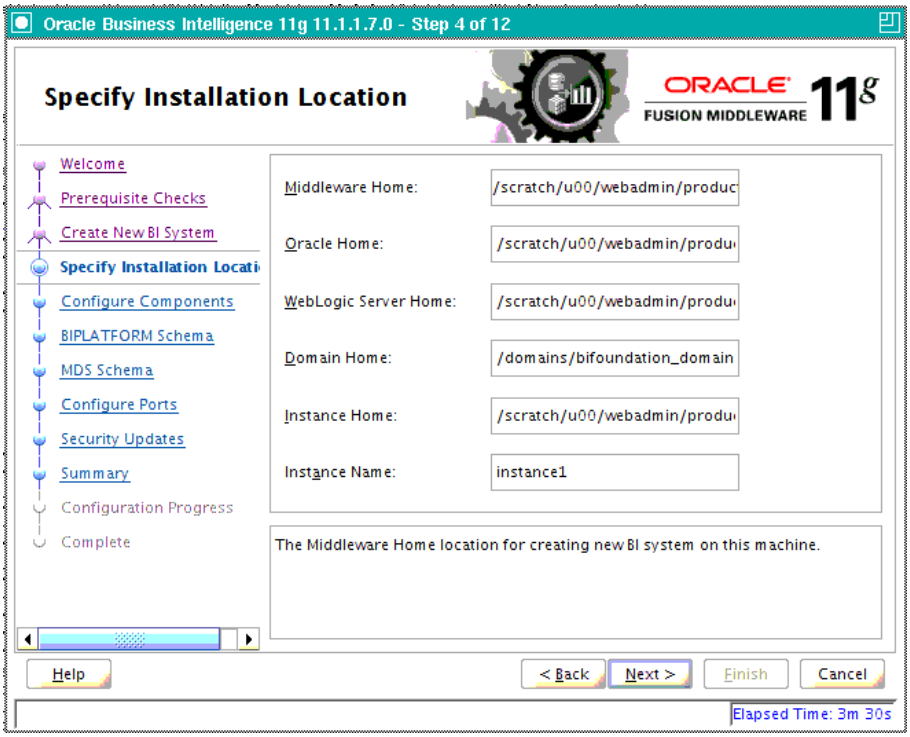


29. Click Next once the prerequisite checks are complete.

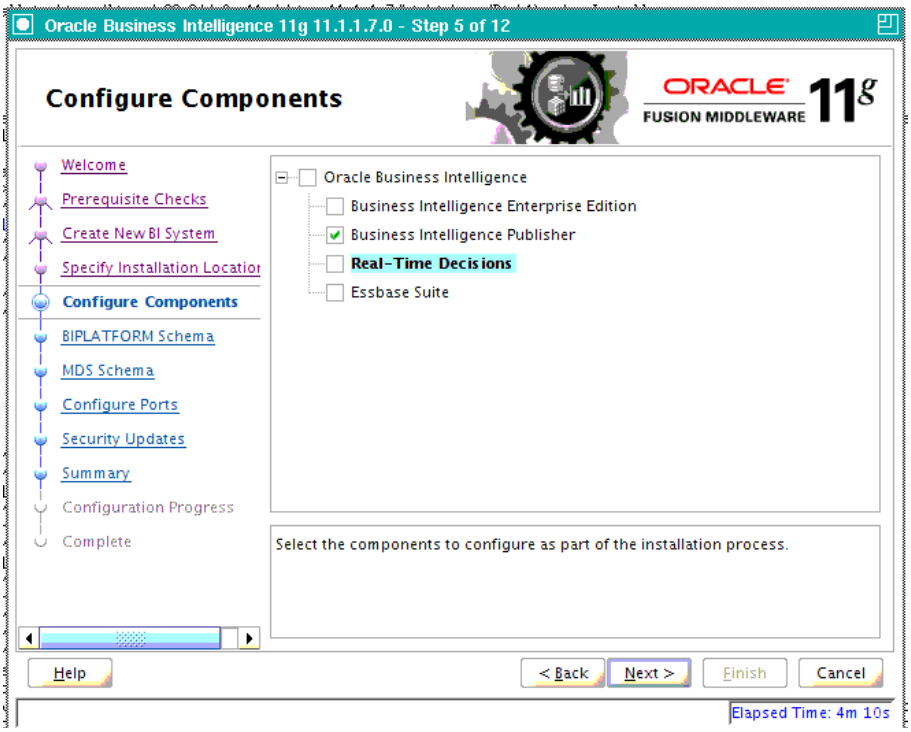


30. On the Create or Scale Out BI System screen, you are asked for the WebLogic password and provided with a recommended a Domain Name. Enter and confirm

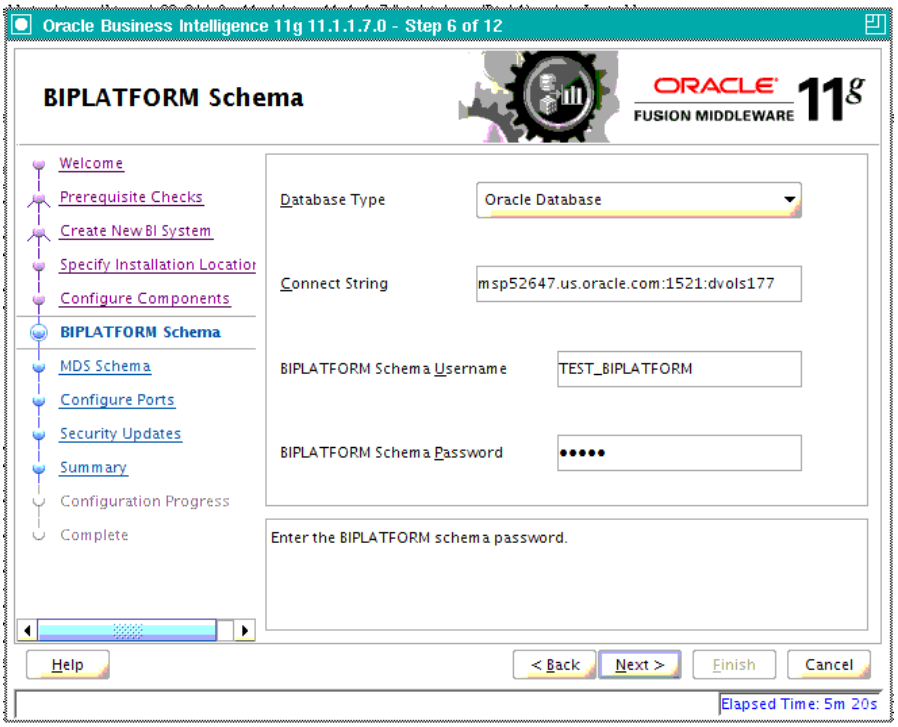
your WebLogic password and accept the recommended Domain Name; "bifoundation domain"



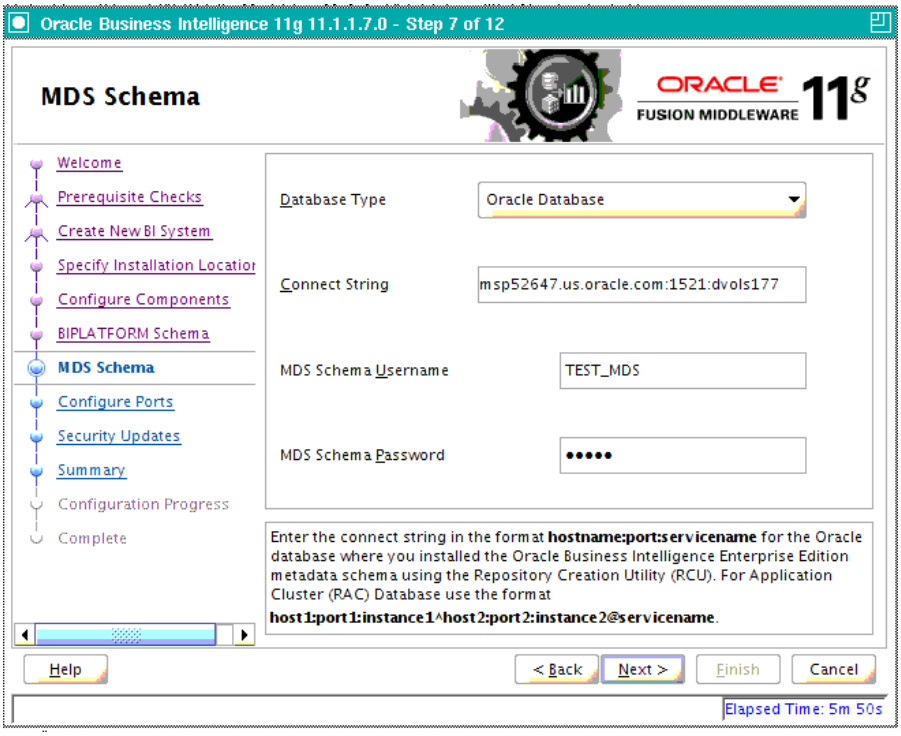
31. Installation locations are recommended by default. Accept these unless you need the installation location to be different from that is suggested. Click Next.



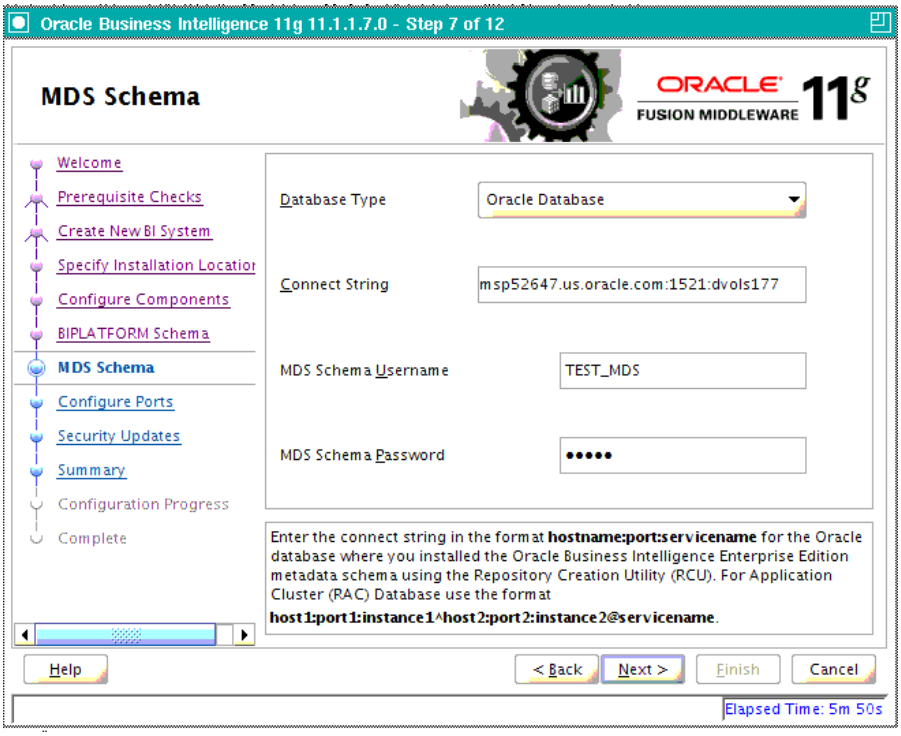
32. Select "Business Intelligence Publisher" and click Next.



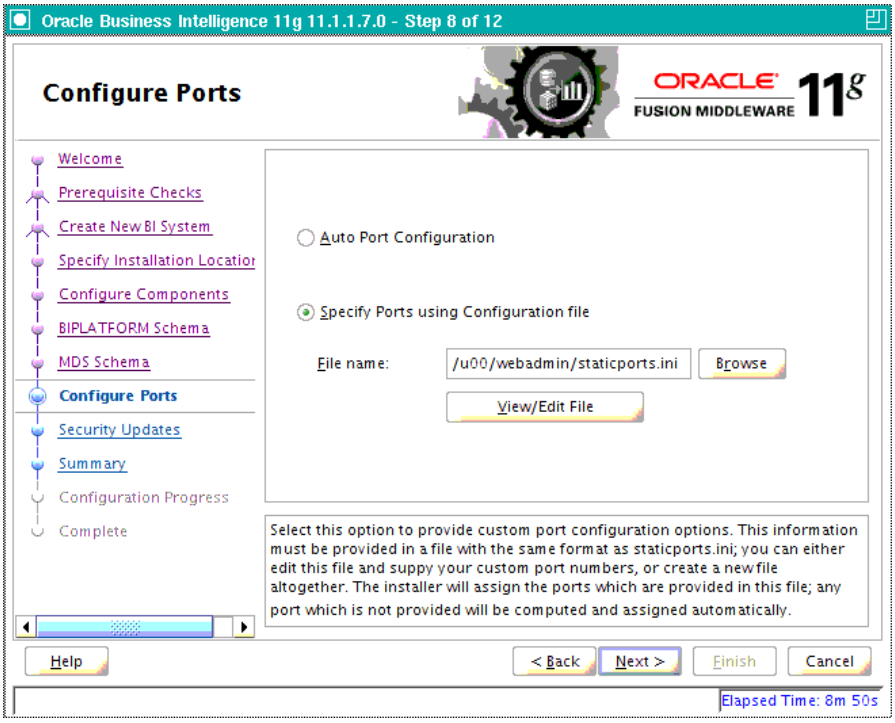
33. Provide the connect string details in the format dbserver:dbport:sid. Provide the details of the BIPLATFORM schema created in the previous steps using RCU. Click Next.



34. MDS schema details are populated by default. Click Next.



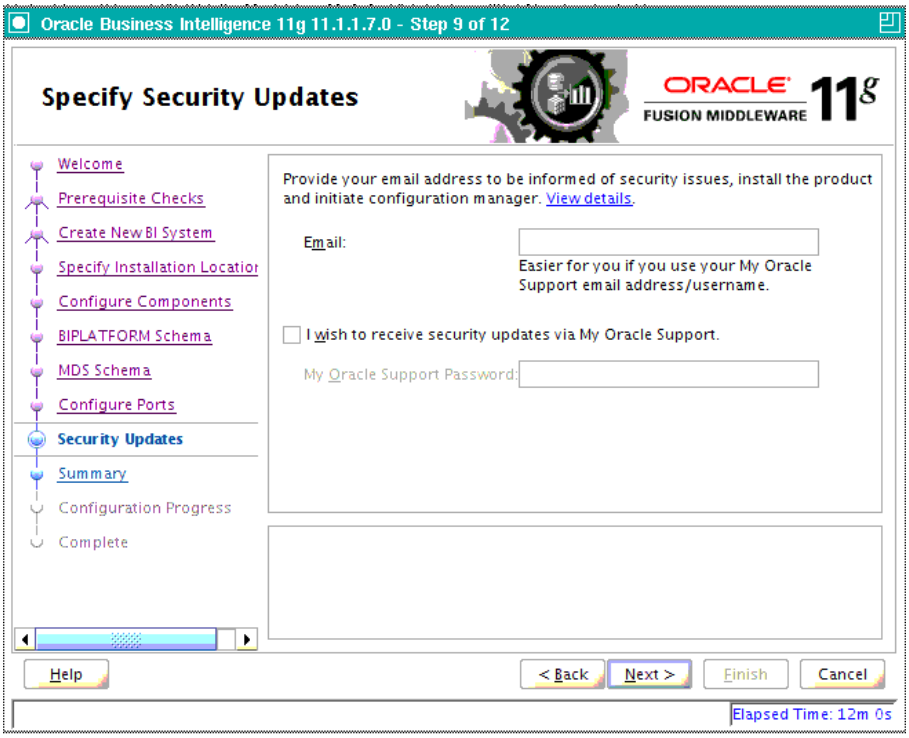
35. Click Next.



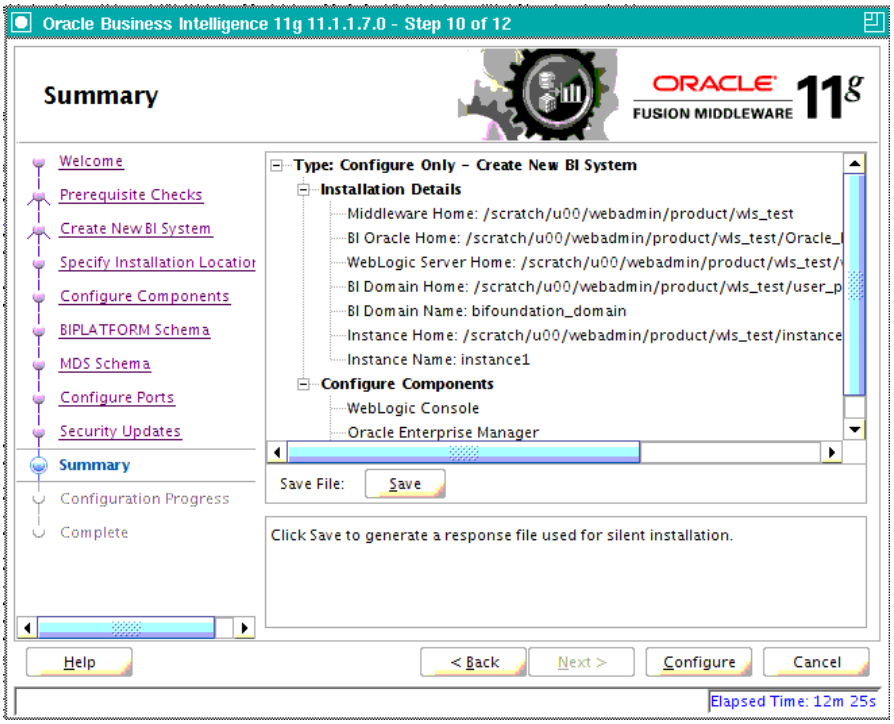
36. Configure your BI ports. This screen allows you to assign Oracle BI EE ports from the staticports.ini file.

This file is located in the Oracle BI EE software at:
/obiee11.1.1.7/bishiphome/Disk1/stage/Response/staticports.ini

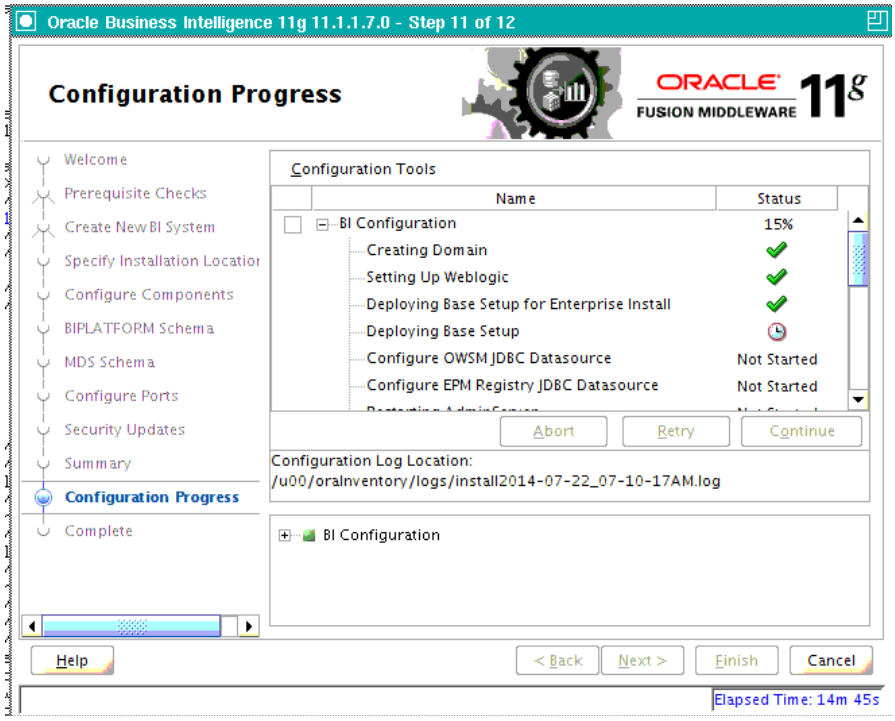
37. Edit this file to make sure you will have the ports you want for your BI Publisher components. Otherwise the installer will assign default port numbers.



38. Uncheck security updates.



39. Summary is displayed. Verify the specifications provided before you click Configure.

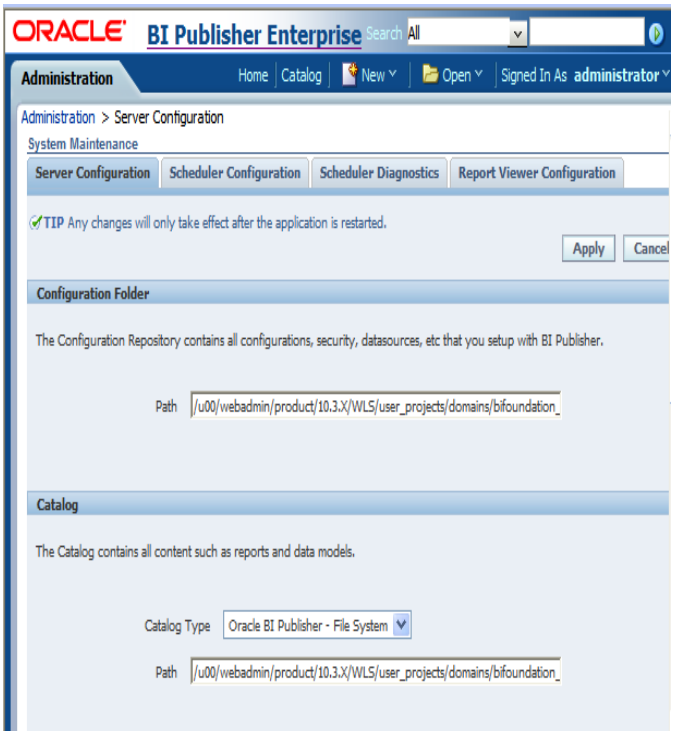


40. Click Finish after the configuration process is finished.
41. Document and test the URLs that are created.

    This screen contains the URL's for the components that got installed.
42. Save this screen, so that you know the right URL's for your installation.
43. To test your BI Publisher installation, launch xmlpserver. Login with the credentials you entered in your Oracle BI EE configuration (weblogic / password).

44. Post install steps: Configure the BI Publisher repository. After signon, select "Administration".

45. On the System Maintenance Section, press Server Configuration
46. Navigate to the Configuration Screen.



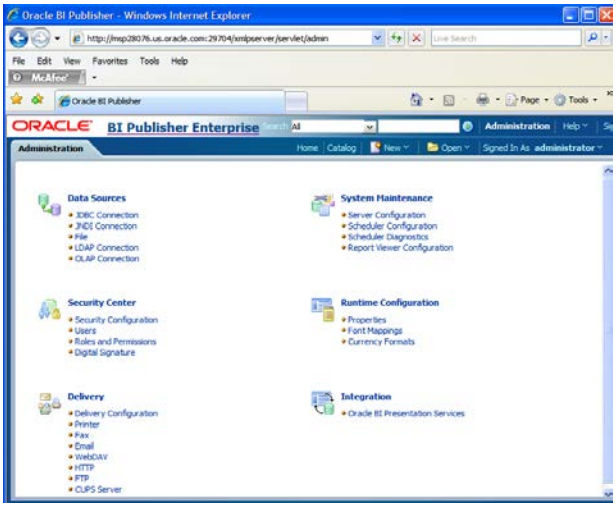
47. On this screen on the Configuration Folder section, enter the path to your repository. On the Catalog section enter Catalog Type: Oracle BI Publisher – File System from the drop down menu.

    This is the path you entered in the Configuration Section and Catalog Section:

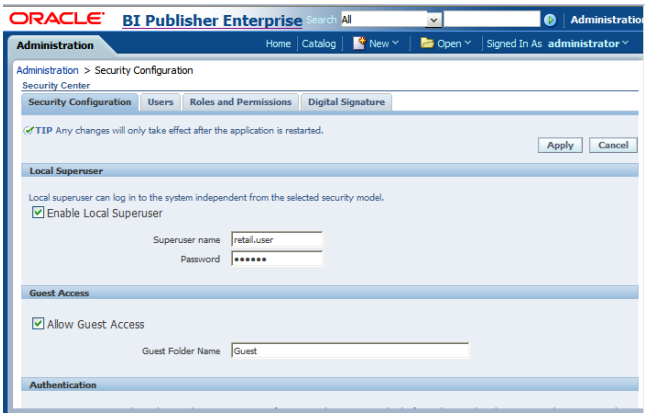
    `$OBIEE_HOME/WLS/user_projects/domains/bifoundation_domain/config/bipublisher/repository`

48. Restart the BI Publisher after this change.

49. Post install step: Set BI Publisher security model



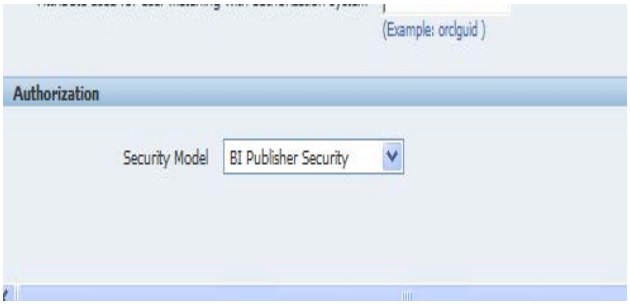
a. On the BI Publisher 11g Administration Screen, click Security Configuration from the Security Center.



b. Enable a superuser by checking the "Enable Local SuperUser" box and by entering name and password on the corresponding fields on this screen.

c. Mark "Allow Guest Access" check box. Enter "Guest" as Guest Folder Name

d. Scroll down the screen and locate the Authorization section:



e. Select BI Publisher Security from the Security Model list.

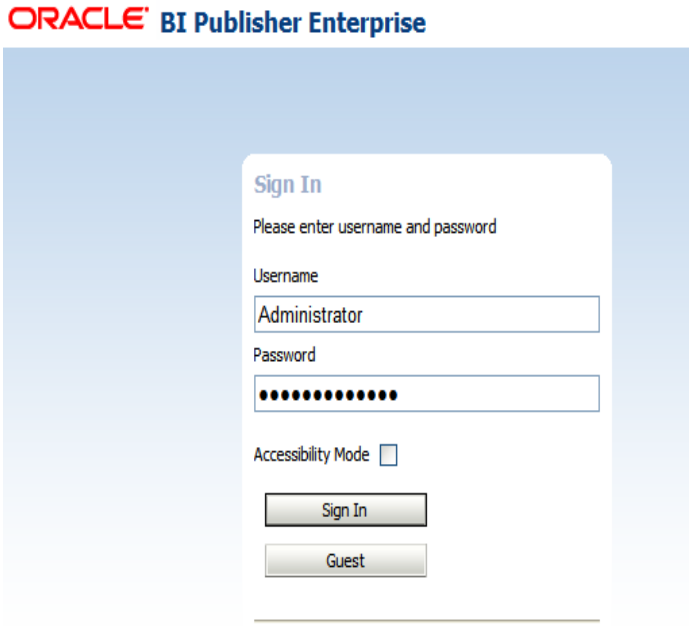
f. The default user name for the BI Publisher Security Model is Administrator

g. On the password text field, enter a value that you can remember. It is going to be the password for Login to xmlpserver.

h. Save the changes and re-start the BI Publisher server.

i. Launch xmlpserver. To Login you must use the new credentials that you set up in the former step: Username: Administrator Password: password.

> **Note:** You will not be able to login to xmlpserver as weblogic any more because we have already changed the Security Model.



50. Post install step: Set the repository path.

Example:
/u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain/config/bipublisher/repository In the Oracle BI EE file system you will find the repository in the following location:

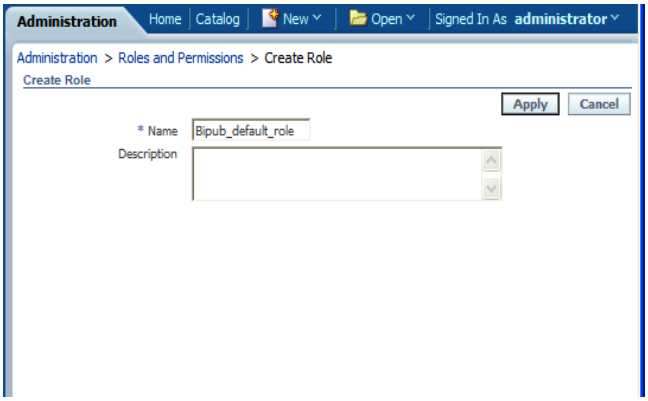
```
$OBIEE/wls/user_projects/domains/bifoundation_domain/config/bipublisher/repository
```

In the repository you will see the following directories:

- Admin
- DemoFiles
- Reports
- Tools
- Users

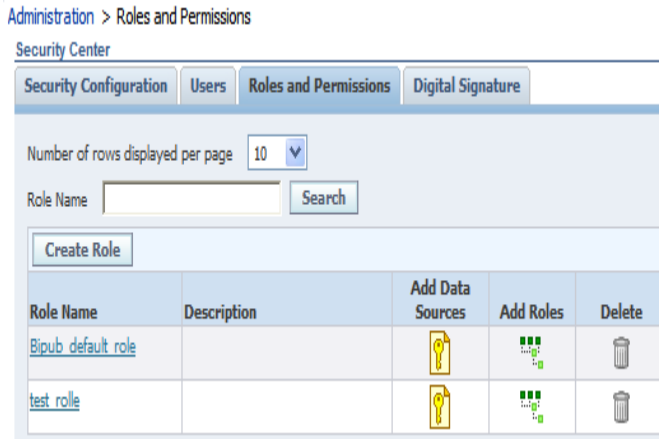
51. Post install step: Create role Bipub_default_role.
    a. From the xmlpserver Administration screen, scroll down to Security Center and click Roles and Permissions.

    b. On the Roles and Permissions screen, click the Create Role button.

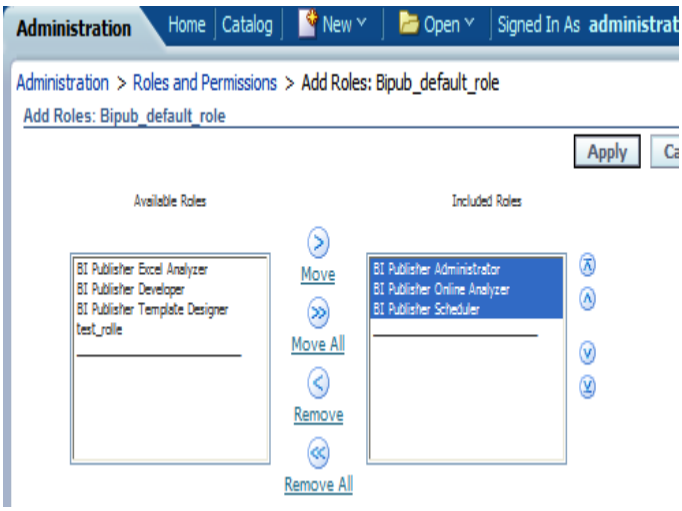


    c. Create the Bipub_default_role. Enter in Create Role Section name of the role.
    d. When the information has been entered press Apply changes.

52. Post install step: Assign BiPub system roles to the newly created Bipub_default_role.
    a. To assign BiPub system roles to the newly create Bipub_default_role, go to Security Center section and navigate to the Roles and Permissions screen:



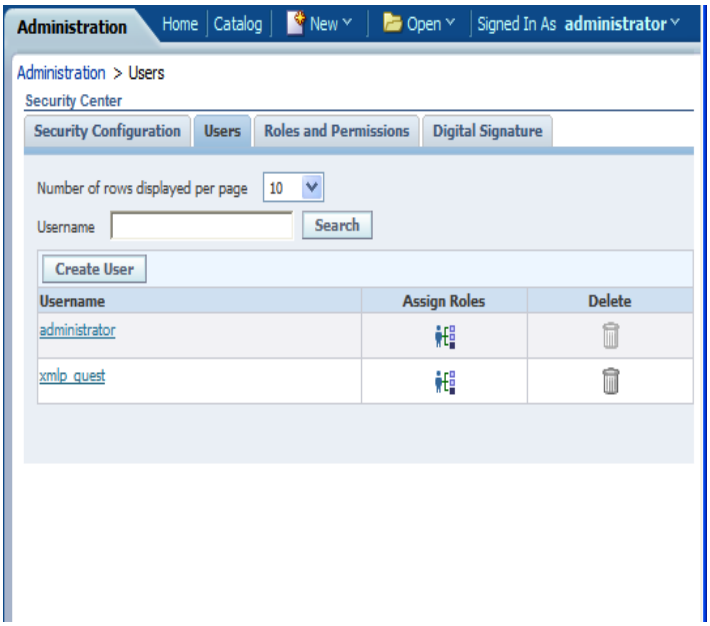
    b. On the Roles and Permissions screen you should see the new role created: "Bipub_default_role". Add multiple roles to the Bipub_Default_Role by pressing the corresponding green icon on the Add Roles column



    c. From the "Available Roles" panel, select the ones needed for your reports and move them to the "Included Roles" panel
    d. Press the Apply button to save your changes.

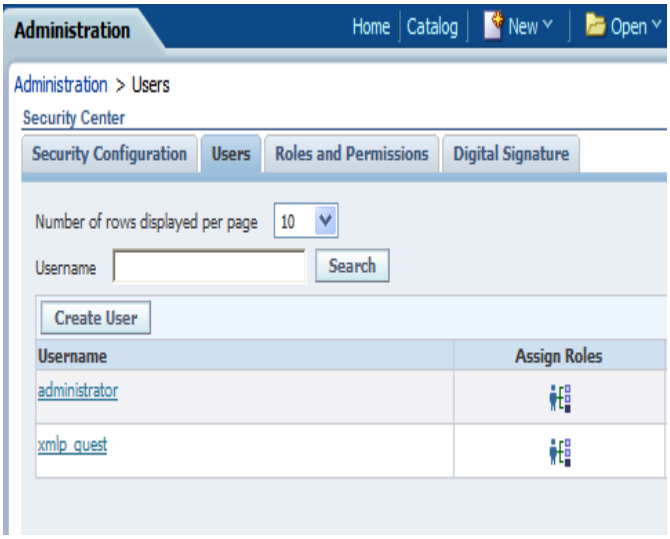
53. Post install step: create Guest (XMLP_GUEST) user.

a. From the xmlpserver Administration screen scroll down to Security Center section and press Users to navigate to the next screen

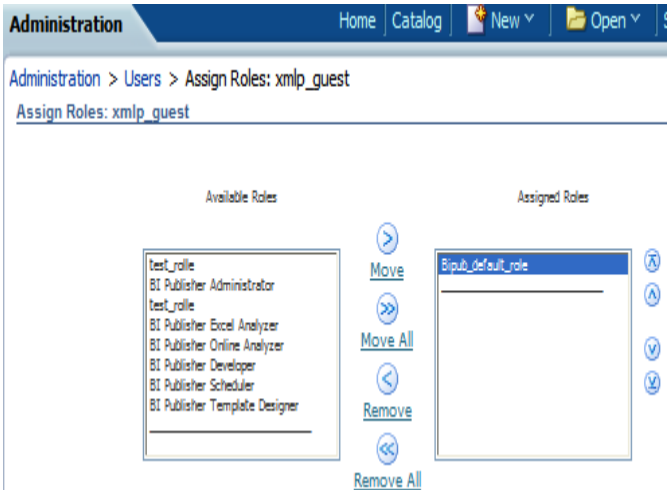


b. Select the "Create User" button to create the "xmlp_guest" user and save the changes

54. Post install step: Adding the Bipub_default_role to XMLP_GUEST user.
    a. Open the Users section:



    b. For xmlp_guest user, press on the "Assign Roles" icon to navigate to the next screen:



    c. On the Assign Roles screen, select the BiPub_default_role from the Available Roles panel to the "Assigned Roles" panel and click Apply to save your changes.

55. Post install step: create folders. Complete the following steps.
    a. Create the "Guest" and "RMS13" directories on the server and change directory into this directory and make sure the permission to these new folders are 755. Example assuming that /u00/webadmin is the root of the installation:

```
cd
/u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain
/config/bipublisher/repository/Reports
mkdir
/u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain
/config/bipublisher/repository/Reports/Guest
cd Guest
mkdir
/u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain
/config/bipublisher/repository/Reports/Guest/RMS13
cd RMS
```

BI Publisher 11g – Manually Copy Reports to Install Directory

If you followed the instructions under "Option 1: Use Application Installer to Patch" in the chapter, "Application Server Installation Tasks—Patch, you can skip to the next section ("Installing the RMS BI Publisher Templates"). If you followed "Option 2: Compile RMS Toolset and Forms Directly," you must manually copy the reports to INSTALL_DIR.

1. If the directories INSTALL_DIR/base/reports/10g or INSTALL_DIR/base/reports/11g exist follow these steps. Otherwise, skip to step 2:
    a. Change directories to INSTALL_DIR/base/reports
    b. Remove the entire 10g directory if it exists. We will no longer ship 10g reports in future releases.
    c. Move all of the directories in INSTALL_DIR/base/reports/11g to INSTALL_DIR/base/reports
    d. Remove the empty INSTALL_DIR/base/reports/11g directory.
2. If the directories INSTALL_DIR/base/reports/10g or INSTALL_DIR/base/reports/11g do not exist, and the reports under INSTALL_DIR/base/reports/ are the 13.2.4 set of reports or earlier, delete all the directories under INSTALL_DIR/base/reports/. All reports that were a part of 13.2.4 and earlier releases are 10g reports that will be completely replaced by 11g reports included in the 13.2.5 and future releases.
3. Copy the reports from the RMS application patch APP_PATCH_DIR/app-patch/<version>/reports to INSTALL_DIR/base/reports/.

BI Publisher 11g Installing the RMS BI Publisher Templates

This section describes how the RMS report templates are installed into the appropriate BI server repositories. BI_REPOSITORY refers to the BI Publisher reports repository.

example:
/u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain/config/bipublisher/repository/

Report files are placed in the directory, INSTALL_DIR/base/reports, and must be copied into the BI repository directory.

1. Change directory to the proper directory under INSTALL_DIR/base/reports/11g. This directory contains subdirectories whose names reflect the names of report templates provided with RMS.

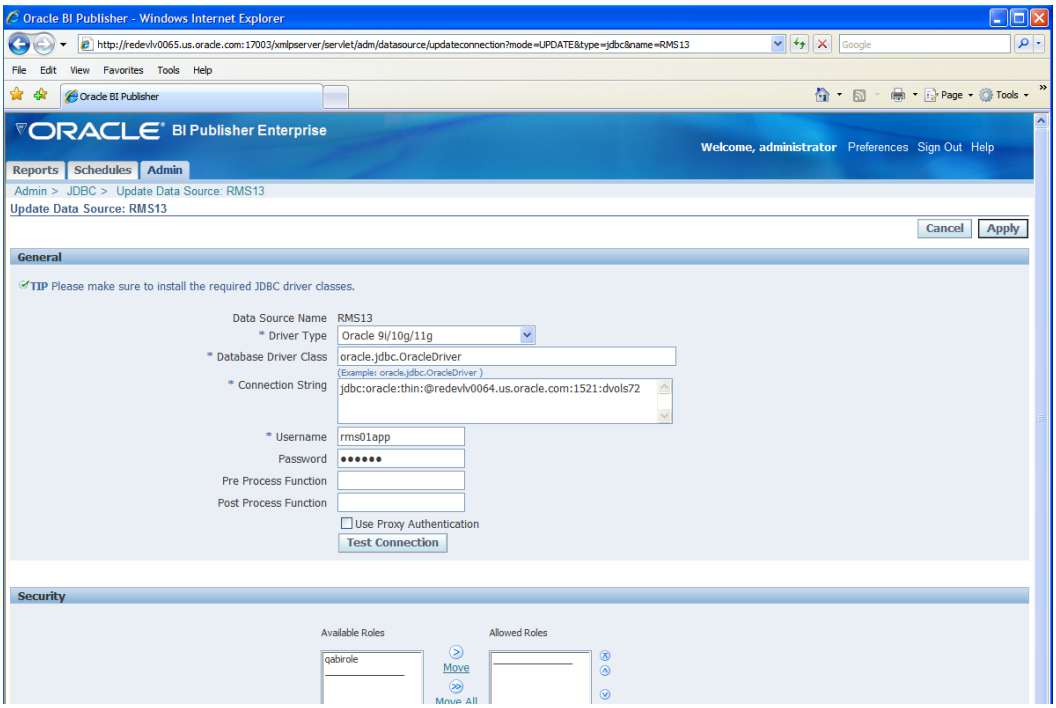
2. Copy each report directory into the directory created above

```
Example,
cp –R *
/u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain
/config/bipublisher/repository/Reports/Guest/RMS13
```

BI Publisher 11g – Configuring the RMS JDBC connection

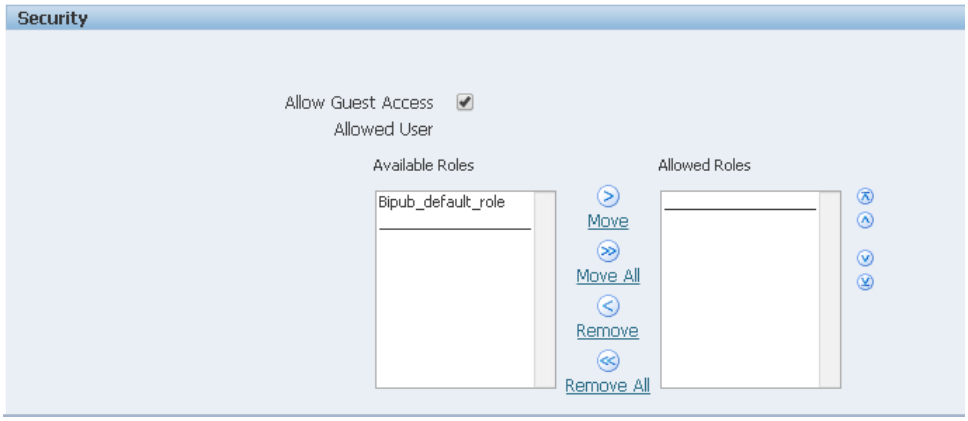
Follow the below steps to configure JDBC connection for RMS Data Source name. This is the data source that RMS uses for RMS reports.

1. Log on with the default user ID and passwords for BI Publisher using the administrative user and password configured previously.
2. Click the **Admin** tab and select the **JDBC Connection** hyperlink in the Data Sources lists. The following screen is displayed.



3. Enter "RMS13" for the datasource name, and enter the appropriate details for the RMS data source. Once the data is entered, click Test Connection to test the connection. Connection string is similar to this example: jdbc:oracle:thin:@redevlv0064.us.oracle.com:1521:dvols72 syntax is jdbc:oracle:thin:@<hostname>:<port>:<dbsid>

4. Select **Allow Guest access** checkbox

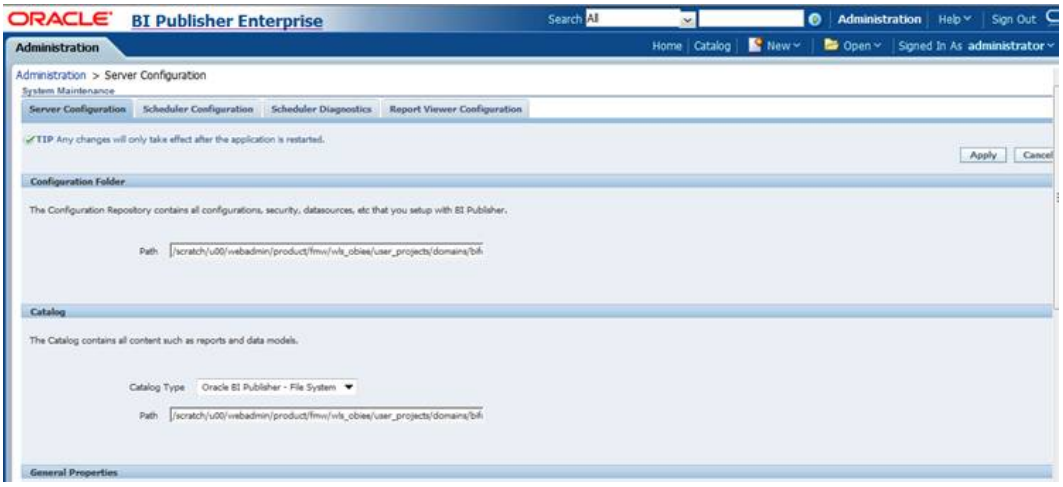


5. Click **Apply** to save the information.

BI Publisher 11g – Verify Oracle BI Publisher Set Up for RMS Reports

Verify that Oracle BI Publisher has been set up correctly as follows:

1. Click the **Server Configuration** tab under the Administration menu. Under the Catalog section, the type should be set to: Oracle BI Publisher- File System and the path set to where the reports are located; REPORTS_DIR.



2. Change the following values in the <installation name>.env file located here: $WLS_HOME/user_projects/domains/<domain name>/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/<installation name>/<installation name>.env
   - ORACLE_RMS_REPORTS_HOST=http://<server>:<port>/

     For example, ORACLE_RMS_REPORTS_HOST=http://redevlv0072.us.oracle.com:7003/

     ORACLE_RMS_RWSERVER=/<location to RMS directory>/

     Example, ORACLE_RMS_RWSERVER=xmlpserver/Guest/RMS13/

8

Data Migration

Included in the 13.2.8 release is a tool responsible for upgrading preexisting data in the RMS schema once 13.2.2 database upgrades are executed. If upgrading from 13.2.1, or earlier, you will need to run this tool to upgrade your data after completing the 13.2.2 Database patch. Running the tool against schemas that have been patched to a version later than 13.2.2 may have unexpected results.

Note: If you already ran the Data Migration tool during or after the 13.2 release, you do not need to run it again.

Note: High volume environments may require multiple days for data migration.

Before running the RMS 13.2 Data Migration Tool:

- Make a backup of all your objects and database schema.
- Check that RMS has 13.2.2 installed.
- Review the enclosed RMS 13.2.8 Patch Release Notes (rms-1328-rn.pdf).

Create Staging Directory for RMS Data Migration Files

To create a staging directory for RMS data migration files, complete the following steps.

1. Log in to the database server as a user that can connect to the RMS database.
2. Create a staging directory for the RMS database schema installation software.
3. Copy the rms1328datamigration.zip file from the RMS 13.2 release to the staging directory. This is referred to as STAGING_DIR when running the data migration tool.
4. Change directories to STAGING_DIR and extract the rms1328datamigration.zip file. This creates a "master_controller" subdirectory under STAGING_DIR.

Configure RMS Data Migration Tool

To configure the RMS data migration tool, complete the following steps.

1. Change directories to STAGING_DIR/master_controller/rms.
1. Source the oraenv script to set up the Oracle environment variables (ORACLE_HOME, ORACLE_SID, PATH, etc).

Example: prompt$. oraenv

ORACLE_SID = [] ? mydb

prompt$

3. Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example: prompt$ echo $ORACLE_HOME
   /u00/oracle/product/mydbversion
   prompt$ echo $ORACLE_SID
   mydb

4. Set and export the NLS_LANG environment variable.

   Example: NLS_LANG=AMERICAN_AMERICA.UTF8
   export NLS_LANG

5. Set and export the TNS_ADMIN environment variable.

   Example: TNS_ADMIN=/path/to/wallet/files/dir/
   export TNS_ADMIN

6. Open the controller.cfg file and replace the values for the following variables with the appropriate values.
   a. export PATCH_DIR=STAGING_DIR/master_controller/rms
   b. export SCHEMA_OWNER=<The name of the RMS schema>
   c. export MMUSER=/@< Schema Owner Wallet Alias >

   Note: See "Appendix: Setting Up Password Stores with Oracle Wallet" for how to set up the database wallet.

   Note: Verify that TNS is set up correctly by using the UP variable to successfully log in to the RMS 13 schema.

   Example: /u00/oracle> sqlplus $UP

7. Configure the following files in the STAGING_DIR/master_controller/rms/files directory with data from your existing RMS schema for the migration. Use the existing files as templates for how this data should be formatted. For descriptions of this data, refer to the RMS 13.2.8 Data Model document (rms-1328-dm.pdf).
   - state.dat

     state.dat is used to update the country for a state in the STATE table. This file is required if there are no stores/warehouses in the system or stores/warehouses are in more than one country. This check is done in the PREVALIDATION routine. Replace the default values in the template state.dat file with the correct values for your schema.
   - country_tax_jurisdiction.dat

     country_tax_jurisdiction.dat is used to populate the jurisdiction codes in the COUNTRY_TAX_JURISDICTION table. Replace the default values in the template country_tax_jurisdiction.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- addr.dat

  addr.dat is used to update the jurisdiction code for a state/country in the ADDR table. Replace the default values in the template addr.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- comp_store.dat

  comp_store.dat is used to update the jurisdiction code for a state/country in the COMP_STORE table. Replace the default values in the template comp_store.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- competitor.dat

  competitor.dat is used to update the jurisdiction code for a state/country in the COMPETITOR table. Replace the default values in the template competitor.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- comphead.dat

  comphead.dat is used to update the jurisdiction code for a state/country in the COMPHEAD table. Replace the default values in the template comphead.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- customer.dat

  customer.dat is used to update the jurisdiction code for a state/country in the CUSTOMER table. Replace the default values in the template customer.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- ordcust.dat

  ordcust.dat is used to update the jurisdiction code for a state/country in the ORDCUST table. Replace the default values in the template ordcust.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- outloc.dat

  outloc.dat is used to update the jurisdiction code for a state/country in the OUTLOC table. Replace the default values in the template outloc.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

- rtv_head.dat

  rtv_head.dat is used to update the jurisdiction code for a state/country in the RTV_HEAD table. Replace the default values in the template rtv_head.dat file with the correct values for your schema. This is optional and required only if jurisdiction codes need to be loaded into the schema.

8. Run the following insert statement into your RMS schema manually. You can modify the default values if necessary:

```
insert into upg_item_supp_manu_country select item,supplier,origin_country_id,
'Y' from item_supp_country;
```

Run the RMS Data Migration Tool

To run the RMS data migration tool, complete the following steps.

Change directories to STAGING_DIR/master_controller/rms.

1. If rerunning the data migration process, clear the contents of the "processed" directory.
2. Run the prevalidation tool. This ensures that the input files for the data migration tool is up to date:
   ```
   $ ./rms132_upgrade.ksh PREVALIDATION
   ```
3. Run migration tool.
   ```
   $ ./rms132_upgrade.ksh UPGRADE
   ```
4. Run the migration cleanup tool. This removes temporary data migration objects from the database.
   ```
   $ ./rms132_upgrade.ksh CLEANUP
   ```
5. Refer to the files in the log and error directory for details if there are problems during migration.
6. You will need to rebuild synonyms for any additional RMS users. Create synonyms to the owner schema for all tables, views, sequences, functions, procedures, packages and types to which the user has access.

Configure ReIM Data Migration Tool

If you choose to migrate ReIM data, follow these steps.

1. Change directories to STAGING_DIR/master_controller/reim.
2. Source the oraenv script to set up the Oracle environment variables (ORACLE_HOME, ORACLE_SID, PATH, etc).

   Example: prompt$. oraenv
   ORACLE_SID = [] ? mydb
   prompt$

3. Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example: prompt$ echo $ORACLE_HOME
   /u00/oracle/product/mydbversion
   prompt$ echo $ORACLE_SID
   mydb

4. Set and export the NLS_LANG environment variable.

   Example: NLS_LANG=AMERICAN_AMERICA.UTF8
   export NLS_LANG

5. Set and export the TNS_ADMIN environment variable.

   Example: TNS_ADMIN=/path/to/wallet/files/dir/

   export TNS_ADMIN

6. Open the controller.cfg file and replace the values for the following variables with the appropriate values:
   - export PATCH_DIR=STAGING_DIR/master_controller/reim
   - export SCHEMA_OWNER=<The name of the RMS schema>
   - export MMUSER=/@< Schema Owner Wallet Alias >

   Notes:

   See "Appendix: Setting Up Password Stores with Oracle Wallet" for how to set up database wallet.

   Verify that TNS is set up correctly by using the UP variable to successfully log in to the RMS 13 schema.

   Example: /u00/oracle> sqlplus $UP

Run the ReIM Data Migration Tool

To run the ReIM data migration tool, complete the following steps.

1. Change directories to STAGING_DIR/master_controller/reim.
2. If rerunning the data migration process, clear the contents of the processed directory.
3. Run migration tool.
   ```
   $ ./reim132_upgrade.ksh UPGRADE
   ```
4. Run migration cleanup tool. This removes temporary data migration objects from the database.
   ```
   $ ./reim132_upgrade.ksh CLEANUP
   ```
5. Refer to the files in the log and error directory for details if there are problems during migration.
6. Rebuild synonyms for any additional RMS users.

Configure Allocation Data Migration Tool

If you choose to migrate Allocation data, follow these steps:

1. Change directories to STAGING_DIR/master_controller/alloc-rms.
2. Source the oraenv script to set up the Oracle environment variables (ORACLE_HOME, ORACLE_SID, PATH, etc).

   Example: prompt$. oraenv

   ORACLE_SID = [] ? mydb

   prompt$

3. Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example: prompt$ echo $ORACLE_HOME

   /u00/oracle/product/mydbversion

   prompt$ echo $ORACLE_SID

mydb

4. Set and export the NLS_LANG environment variable.

   Example: NLS_LANG=AMERICAN_AMERICA.UTF8

   export NLS_LANG

5. Set and export the TNS_ADMIN environment variable.

   Example: TNS_ADMIN=/path/to/wallet/files/dir/

   export TNS_ADMIN

6. Open the controller.cfg file and replace the values for the following variables with the appropriate values :
   - export PATCH_DIR=STAGING_DIR/master_controller/alloc-rms
   - export SCHEMA_OWNER=<The name of the RMS schema>
   - export MMUSER=/@< Schema Owner Wallet Alias >

   Note: See "Appendix: Setting Up Password Stores with Oracle Wallet" for how to set up database wallet.

   Note: Verify that TNS is set up correctly by using the UP variable to successfully log in to the RMS 13 schema.

   Example: /u00/oracle> sqlplus $UP

Run the Allocation Data Migration Tool

To run the allocation data migration tool, complete the following steps.

1. Change directories to STAGING_DIR/master_controller/alloc-rms.
2. If rerunning the data migration process, clear the contents of the processed directory.
3. Run prevalidation tool. This ensures that the input files for the data migration tool is up to date:

   ```
   $ ./allocation132_upgrade.ksh PREVALIDATION
   ```

4. Run migration tool.

   ```
   $ ./allocation132_upgrade.ksh UPGRADE
   ```

5. Run migration cleanup tool. This removes temporary data migration objects from the database.

   ```
   $ ./allocation132_upgrade.ksh CLEANUP
   ```

6. Refer to the files in the log and error directory for details if there are problems during migration.
7. Rebuild synonyms for any additional RMS users.

9

Web Services Installation

Some Oracle Retail applications; <app> (for example, RMS) use Oracle Objects for the PL/SQL API's. The tool generates a Web Service Provider layer between the external clients and the <app> API's to provide the Web Service functionality, such as faults, logging, and security, as well as the conversion from xml payloads to Oracle Objects. The Retail Service Enabler (RSE) tool creates the appropriate Provider web service endpoints as well as templates for the PL/SQL APIs.

Note: Depending on your business needs, you may not need to install web services.

Note: If you are utilizing the AIA 2.5 solution for PeopleSoft you must skip this section. Please contact customer support for details

Extract Web Services Files

To extract Web Services files, do the following:

1. Create a directory under the Batch INSTALL_DIR to hold the web services files. This will be referred to as Web Service Objects.
2. Copy the rms1328webservices.zip file from the RMS 13.2.8 release to INSTALL_DIR/'Web Service Objects'.
3. Unzip rms1328webservices.zip from INSTALL_DIR/'Web Service Objects'.

Set up Environment

To set up the environment, do the following:

1. Source the oraenv script to set up the Oracle environment variables (ORACLE_HOME, ORACLE_SID, PATH, etc)

   Example:
   ```
   prompt$ . oraenv
   ORACLE_SID = [] ? mydb
   prompt$
   ```

2. Verify the ORACLE_HOME and ORACLE_SID variables after running this script.

   Example:
   ```
   prompt$ echo $ORACLE_HOME
           /u00/oracle/product/mydbversion
   prompt$ echo $ORACLE_SID
           mydb
   ```

3. export TNS_ADMIN=/path/to/wallet/files/dir/
4. export UP=/@<Schema Owner Wallet Alias>

Note: See "Appendix: Setting Up Password Stores with Oracle Wallet" for how to set up database wallet.

5. Verify that TNS is set up correctly by using the UP variable to successfully log in to the RMS 13 schema.

   Example: /u00/oracle> sqlplus $UP

Drop Existing Java Code from RMS Database Schema

If you have an existing Web service installation in the RMS schema, you will need to drop the Java code from it before continuing.

```
dropjava -u $UP -v missing11g.jar dbwsclientws.jar dbwsclientdb11.jar

dropjava -u $UP -v GetDrillBackForwardURLConsumer.jar

dropjava -u $UP -v GlAccountValidationServiceConsumer.jar

dropjava -u $UP -v ProcessGLAccountValidationRetailReqABCSImplConsumer.jar
```

If you encounter errors while dropping the Java, drop the individual classes..

Example: dropjava –u $UP oracle/retail/integration/services/consumer/GetURLWS/runtime/SourceSystem_Encoder

Loading Java Code to the RMS Database Schema

Web service installation involves loading java code to the RMS database schema itself. Perform the following procedures to load java code to the RMS database schema..

1. Increase ORACLE initialization parameter JAVA_POOL_SIZE. 150MB is recommended.
2. Run the following commands:

   ```
   cd INSTALL_DIR/'Web Service Objects'/Consumer/lib

   loadjava -u $UP -v -r -f missing11g.jar dbwsclientws.jar dbwsclientdb11.jar
   ```

3. Make sure the step above completes with 0 errors. If you encounter errors, run the following command, correct the error, and then repeat the steps above.

   ```
   dropjava -u $UP -v missing11g.jar dbwsclientws.jar dbwsclientdb11.jar
   ```

4. Change the contents of the following files to your RMS schema owner name when seeing the value <USER>.
   - INSTALL_DIR/'Web Service Objects'/Consumer/sql/GetDrillBackForwardURLConsumer_grant.sql
   - INSTALL_DIR/'Web Service Objects'/Consumer/sql/GlAccountValidationServiceConsumer_grant.sql

     Example: Change all occurrence of <USER> to RMS schema owner RMS01 in the files:

     ```
     dbms_java.grant_permission( '<USER>',
     'SYS:java.lang.RuntimePermission', 'setFactory', '' )
     ```

     to

     ```
     dbms_java.grant_permission( 'RMS01',
     'SYS:java.lang.RuntimePermission', 'setFactory', '' )
     ```

5. Run the above files as the database sys user.

6. Perform the following commands to load java to the database:
   ```
   cd ../jars
   ```
   ```
   loadjava -u $UP -v -r -f GetDrillBackForwardURLConsumer.jar
   ```
7. Make sure the step above completes with 0 errors. If you encounter errors, run the following command, correct the error, and then repeat the steps above.
   ```
   dropjava -u $UP -v GetDrillBackForwardURLConsumer.jar
   ```
8. Perform the following commands to continue loading java to the database:
   ```
   loadjava -u $UP -v -r -f GlAccountValidationServiceConsumer.jar
   ```
9. Make sure the step above completes with 0 errors. If you encounter errors, run the following command, correct the error, and then repeat the steps above.
   ```
   dropjava -u $UP -v GlAccountValidationServiceConsumer.jar
   ```
10. You do NOT create synonyms to each java object loaded as the synonyms were created in packages previously loaded pointing to the exposed java objects.

Create a Managed Server

Create a managed server for the RMS Web services app to be deployed per the *WebLogic Installation Guide*.

Create a Datasource

Create a datasource for RMS Webservices to point to the RMS schema as follows.

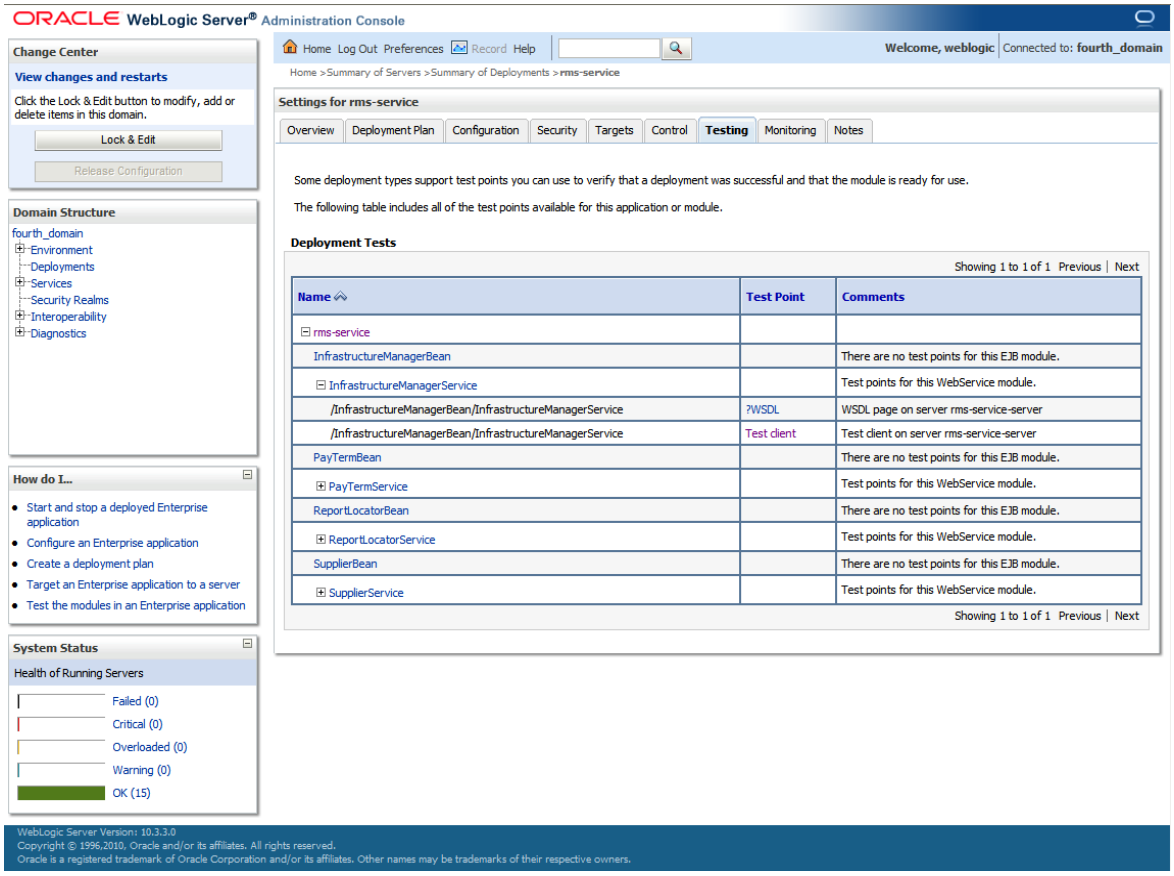
- Name can be anything you want.
- JNDI Name must be jdbc/RetailWebServiceDs.
- Set database type and driver for your environment (use non-XA jdbc driver).
- Set connection properties for the database using the rms user (rms01user). Be sure to test the configuration before moving on.
- Point the data source to the server created in the Create a Managed Server section above.

Deploy RMS Service EAR File

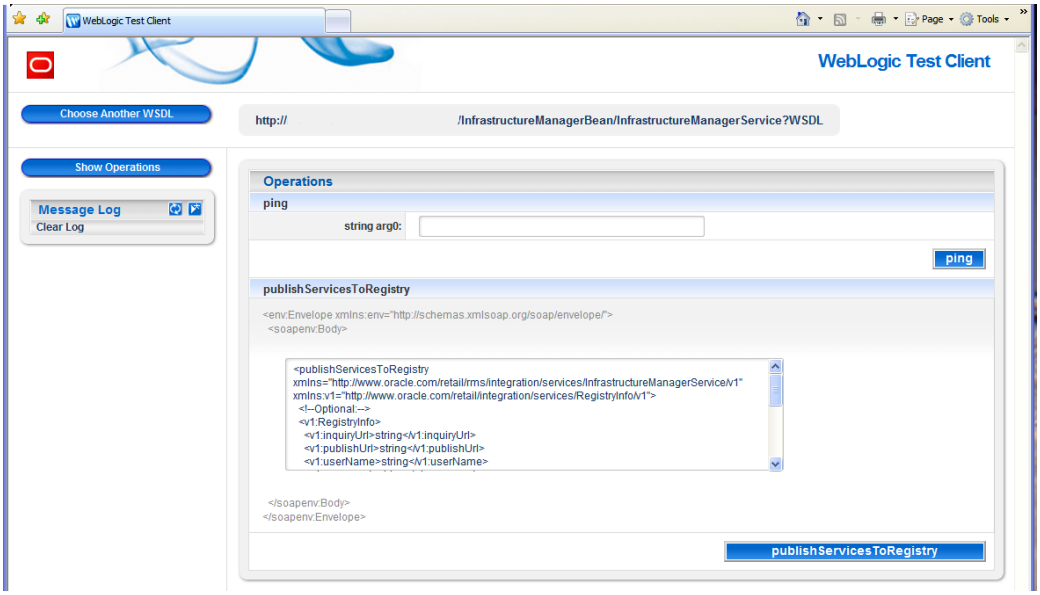
To deploy the RMS Service .ear file, do the following.

1. Make sure that the managed server created in Step 2, where this application will be deployed, is up and running.
2. In the left Domain Structure window, click Environment > Deployments.
3. Click Lock and Edit in the change center to install the ear file. It will enable the install button on the deployments screen.
4. Click Install.
5. Click the upload your file(s) link.
6. Click the Deployment Archive browse button.
7. Select the rms-service.ear file from local machine.
8. Click Next. Make sure that the radio button for rms-services.ear is selected.
9. Click Next again. Make sure that Install this deployment as an application is selected.
10. Click Next again and select the server created in Step 2.
11. Click Next. Click Finish to return to the deployments page. You should see rms-service in the list of deployments.

12. Click Activate Changes in the change center. The state of the application may be shown as prepared. If so, select the check box next to rms-service to will enable the Start button. Click Start. Select servicing all requests.
13. To test the deployment, click on the application. Click the testing tab.



14. Expand one of the four web services. Click the ?WSDL and Test Client links to test. For the test client you should see a screen similar to the following:



Configure Web Service Security

> **Note:** If you are utilizing the ORFI solution for PeopleSoft or EBS, you must skip this section.

This section details how to configure the web service deployment to use the WS-Security Username Profile. Configuring this policy will force all incoming requests to contain WS Security headers to authenticate the requestor based on a user name and password elements. The use of this profile does not provide any confidentiality protection on web service requests: data contained within the Web service messages will not be encrypted. However, using a secure message transport, such as SSL/TLS, will provide confidentiality for the message as it traverses the network. For more information on using SSL/TLS see the section, "Configuring SSL" found in the WebLogic document, "Securing the WebLogic Server, 10g Release 3 (10.3)".

Additional WS Security policies may also be available depending on the configuration of the WebLogic server. Using these policies will require appropriate changes to web service requests created by applications consuming the web service. Many of these policies also require additional steps for correct keystore and truststore file configuration.

> **Note**: The ORFI solution for PeopleSoft and EBS does not support the WS-Security Username profile.

When a web service uses the WS-Security Username profile, all web service consumers must specify a user name configured within the current WebLogic domain. This user name must also have the appropriate role(s) associated with it. Using this profile is thus a two-step process:

1. Attach the WS-Security Username policy to the web service
2. Create roles and users who can access the web services

These steps are explained below.

Attach Policy File to the Web Service

The ear file contains **usernametoken.xml** in the **META-INF/policies** folder. This file contains the policy which is used by the web service. Follow the below steps to attach it to a web service:

1. Click on the application in the deployments screen.
2. The overview page of the application shows all the modules and components which are installed as part of the application:
3. Click on the web service for which you want to enable security.
4. Click the Configuration > WS-Policy tab of the web service. You should see the Web Service port (for example, PayTermPort) in the list of service endpoints.
5. Click the plus sign next to the Web Service to show all the operations of the Web Service.
6. Secure all the operations of the Web Service or selected operations of the Web Service. If you click on the Web Service port, it will show the next page, where you can attach policy file to the web service.
7. In the Available Endpoint Policies list, select the policy:usernametoken.xml option and move it to the list of Chosen Endpoint Policies.
8. Click OK. Leave all default values as they are.
9. Click OK again.
10. Verify the policy details have been added by clicking on the Testing tab of the web service page, and then selecting the WSDL. The WSDL should contain content similar to that shown below, along with additional service specific details:

```
<?xml version='1.0' encoding='UTF-8'?>
 <definitions xmlns:wssutil="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-utility-1.0.xsd"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://www.oracle.com/retail/rms/integration/services/PayTermServic
e/v1" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://www.oracle.com/retail/rms/integration/services/PayTerm
Service/v1" name="PayTermService">
<wsp:UsingPolicy wssutil:Required="true" />
    <wsp:Policy wssutil:Id="usernametoken">
    <ns1:SupportingTokens xmlns:ns1="http://docs.oasis-open.org/ws-sx/ws-
    securitypolicy/200512">
    <wsp:Policy>
<ns1:UsernameToken ns1:IncludeToken="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200512/IncludeToken/AlwaysToRecipient">
<wsp:Policy>
<ns1:WssUsernameToken10 />
</wsp:Policy>
</ns1:UsernameToken>
</wsp:Policy>
</ns1:SupportingTokens>
</wsp:Policy>
```

Create Roles and Users who can Access the Web Services

The second step is to create roles and users who can access the Web services, as follows.

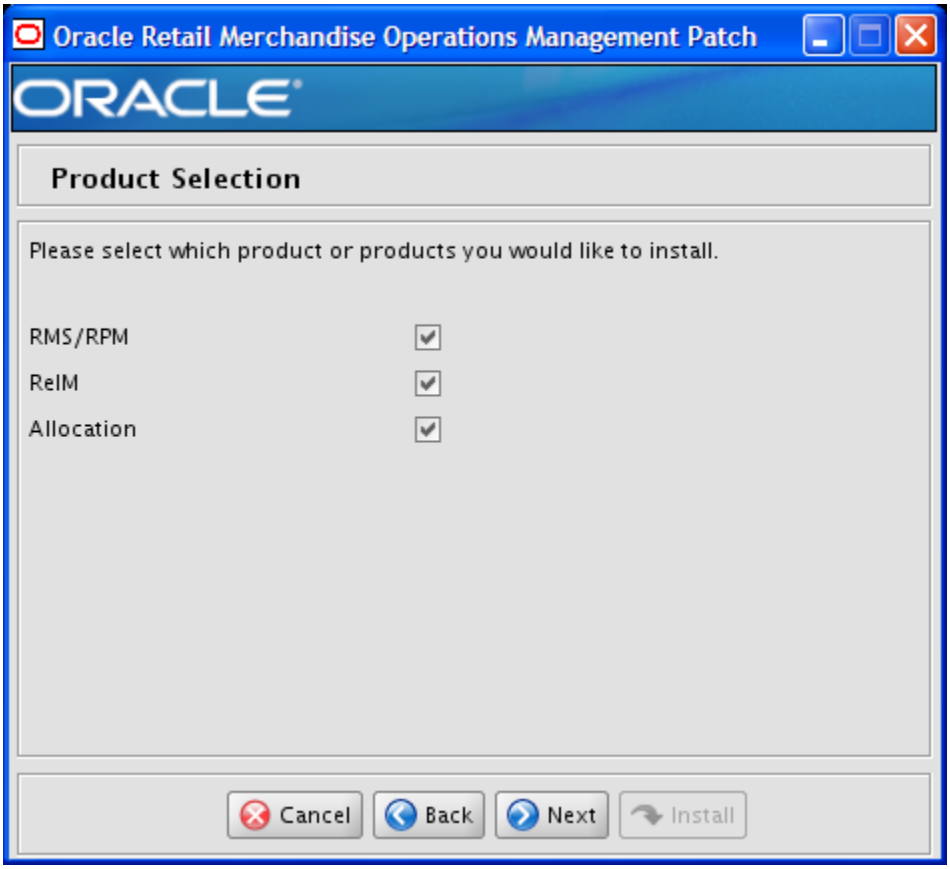
1. Add users to the security realm. In the Domain Structure window, click the Security Realms link. The default realm is shown.
2. Click the link on the realm.
3. Click the Users and Groups tab.
4. Click New. And enter user name and password details on the next screen. Leave the default value for Provider.
5. Click OK to save the changes. The new user is shown in the list of users
6. Add roles either from the Roles and Policies tab of the security realm, or through the Security tab of the Web Service. This example shows how to create a role from the security tab of the Web Service.
7. Navigate to the Security tab of the Web Service to which you are adding security.
8. In the Roles tab, click New.
9. In the Name field, enter the role name. For example, rmsrole. Leave the Provider Name to default value. Click OK. The newly created role is shown in the role tab.
10. Add the user to the role. Click on the newly created role.
11. Click Add Conditions.
12. Select User in the Predicate List drop down. Click Next.
13. Enter the user name that was created in the security realm. Click Add. It is added to the list below the text box.
14. Click Finish.
15. Click Save.
16. Navigate back to the Security > Policies tab of the Web Service.
17. Click Add Conditions.
18. Select Role in the Predicate List drop down. Click Next.
19. Enter the role name that was created earlier and click Add. The role is added in the list below the text box.
20. Click Finish.
21. Click Save.

The process of adding security to Web Services is complete. Open the test page of the Web Service and verify that the Web Service is secured.

A

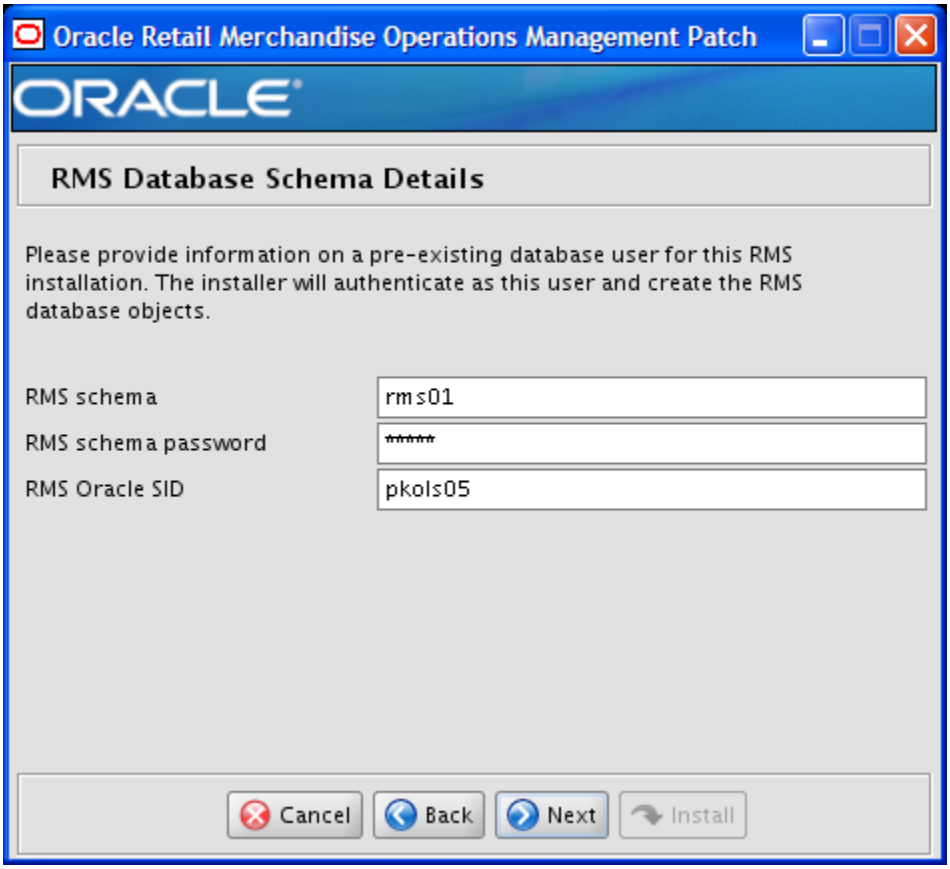
Appendix: RMS Database Patch Installer Screens

Screen: Product Selection



Field Title	Product Selection
Field Description	By default the RMS database schema patch installer creates the database objects for RMS/ReSA/RTM and RPM. As an option, the database objects for ReIM and/or Allocation may be installed at the same time or later.
Example	RMS/RPM, ReIM, Allocation

Screen: RMS Database Schema Details



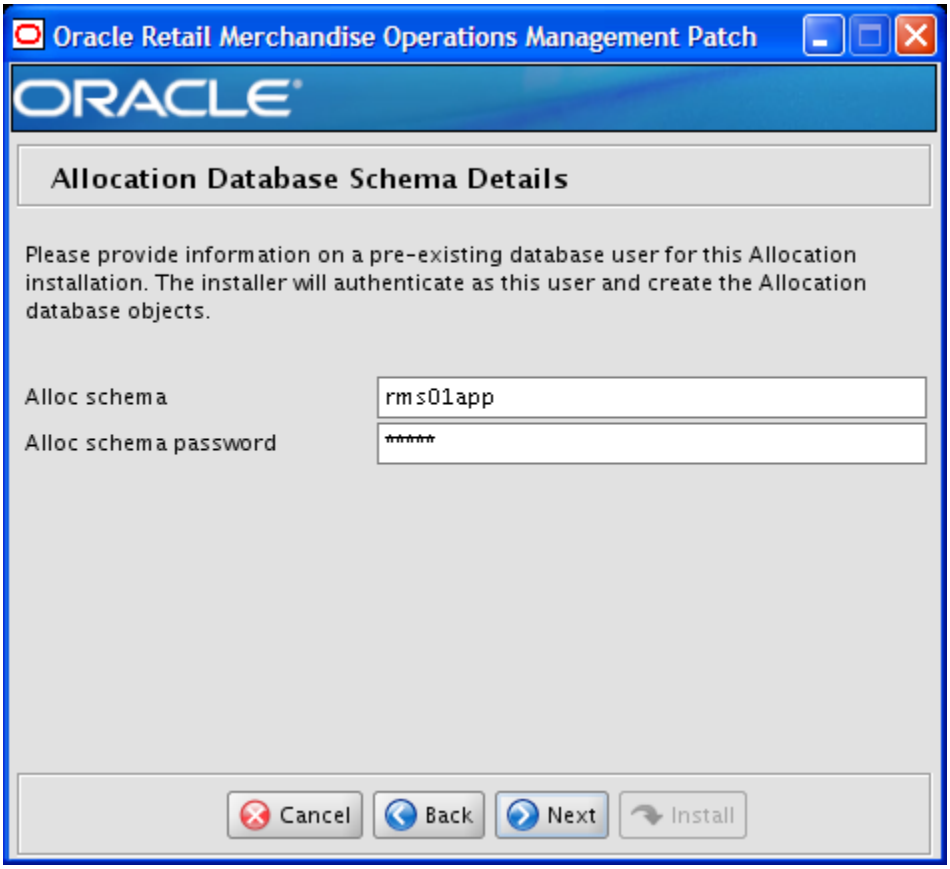
Field Title	RMS schema
Field Description	Provide the RMS database user here. The installer logs into the database as this user to patch the RMS schema. This user must already exist in the database when the RMS database schema patch installer is run.
Example	rms01

Field Title	RMS schema password
Field Description	Database password for the RMS schema Owner.

Field Title	RMS Oracle SID
Field Description	Oracle system identifier for the database where the RMS patch will be applied.
Example	pkols05

The database settings provided are validated by the installer when you advance to the next screen.

Screen: Allocation Database Schema Details

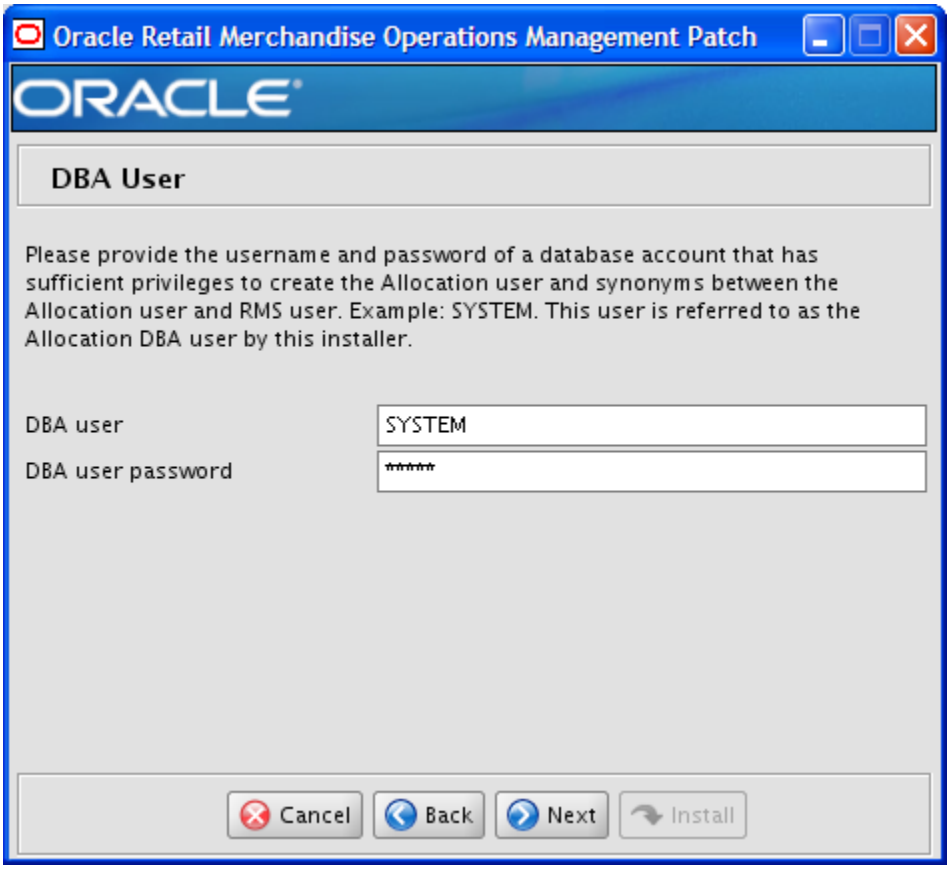


Field Title	Alloc schema
Field Description	Provide the Allocation database user here. The installer logs into the database as this user to patch the Allocschema. This user must already exist in the database when the RMS database schema patch installer is run.
Example	rms01app

Field Title	Alloc schema password
Field Description	Database password for the Allocation user schema.

The database settings provided are validated by the installer when you advance to the next screen.

Screen: DBA User

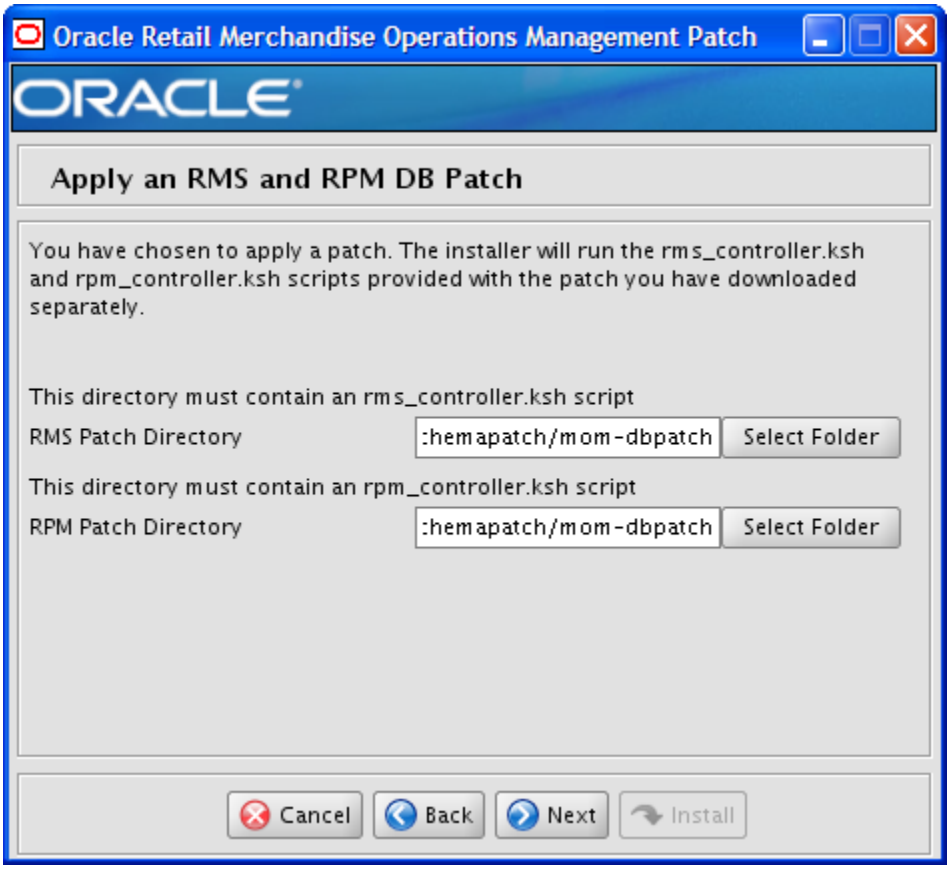


Field Title	DBA user
Field Description	Provide a database user with sufficient privileges to create synonyms between other users. The installer logs into the database using this account and creates the synonyms needed between the RMS and Allocation users.
Example	SYSTEM

Field Title	DBA user password
Field Description	Database password for the DBA user.

The database settings provided are validated by the installer when you advance to the next screen.

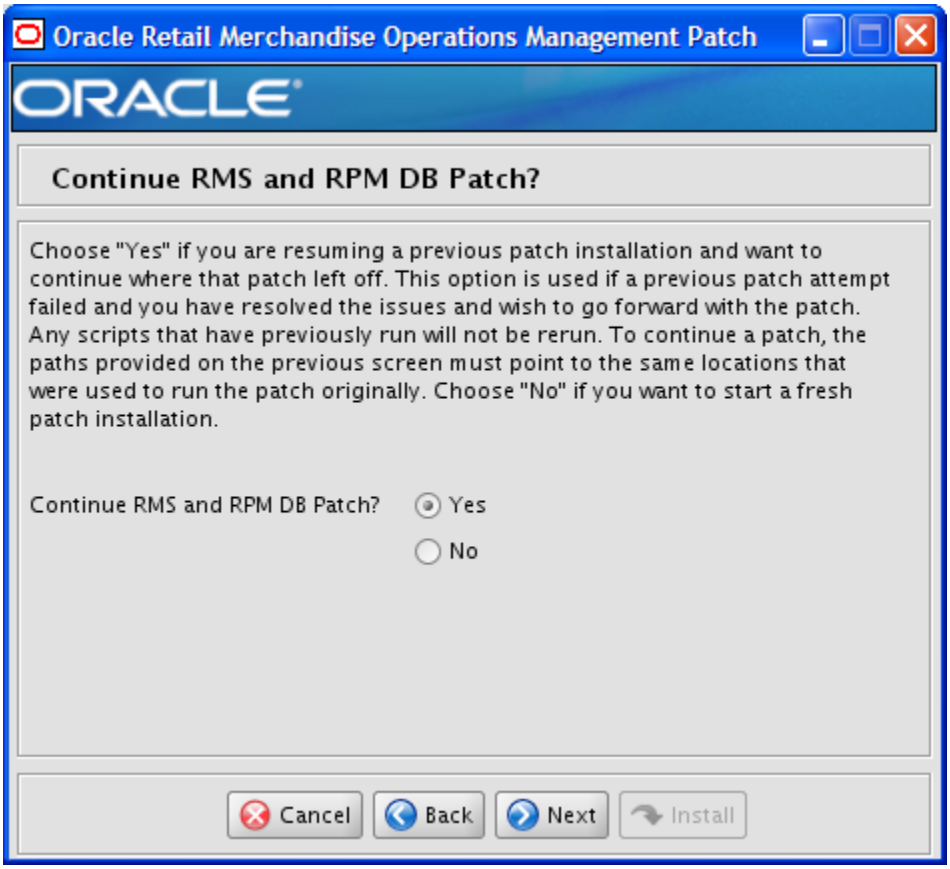
Screen: Apply an RMS and RPM DB Patch



Field Title	RMS Patch Directory
Field Description	Provide the directory path to the RMS patch you want to install. The installer runs only the patch you provide. **Note:** The directory you choose must contain an rms_controller.ksh file.
Example	/path/to/rms/dbschemapatch/mom-dbpatch for all 13.2.x patches **Note:** The patch option is intended for patches starting with 13.2.

Field Title	RPM Patch Directory
Field Description	Provide the directory path to the RPM patch you want to install. The installer runs only the patch you provide. **Note:** The directory you choose must contain an rpm_controller.ksh file.
Example	/path/to/rms/dbschemapatch/mom-dbpatch for all 13.2.x patches **Note:** The patch option is intended for patches starting with 13.2.

Screen: Continue RMS and RPM DB Patch

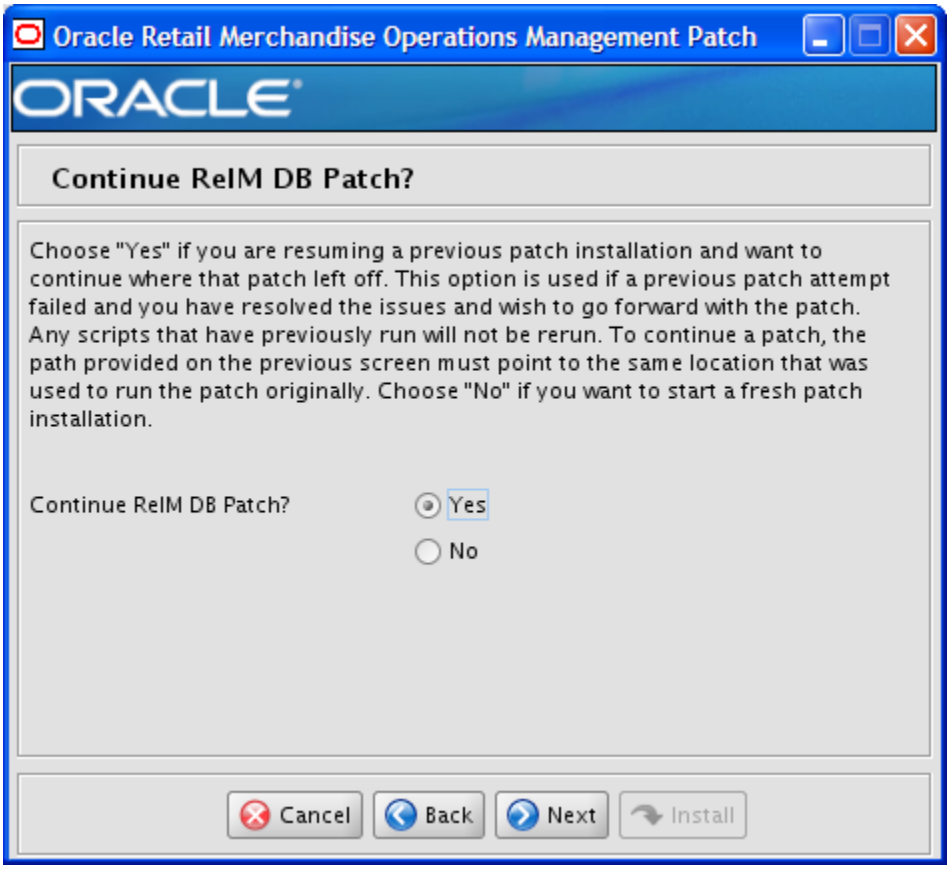


Field Title	Continue RMS and RPM DB Patch?
Field Description	The patch process allows you to continue a previously run patch if it stopped before completion or failed. If Yes is selected, any scripts that were previously run for the RMS and RPM patch will be skipped. If No is selected, the patch will start from the beginning. **Note:** To continue a patch, the content of the "processed" directories in the RMS Patch Directory and RPM Patch Directory chosen on the previous screen must be the same as it was after the previous patch was stopped. If you choose No, this directory will be cleared, and you will not be able to continue this patch in the future.

Screen: Apply RelM DB Patch

Field Title	Patch Directory
Field Description	Provide the directory path to the ReIM patch you want to install. The installer runs only the patch you provide. **Note:** The directory you choose must contain a reim_controller.ksh file.
Example	/path/to/rms/dbschemapatch/mom-dbpatch for all 13.2.x patches **Note:** The patch option is intended for patches starting with 13.2.

Screen: Continue RelM DB Patch



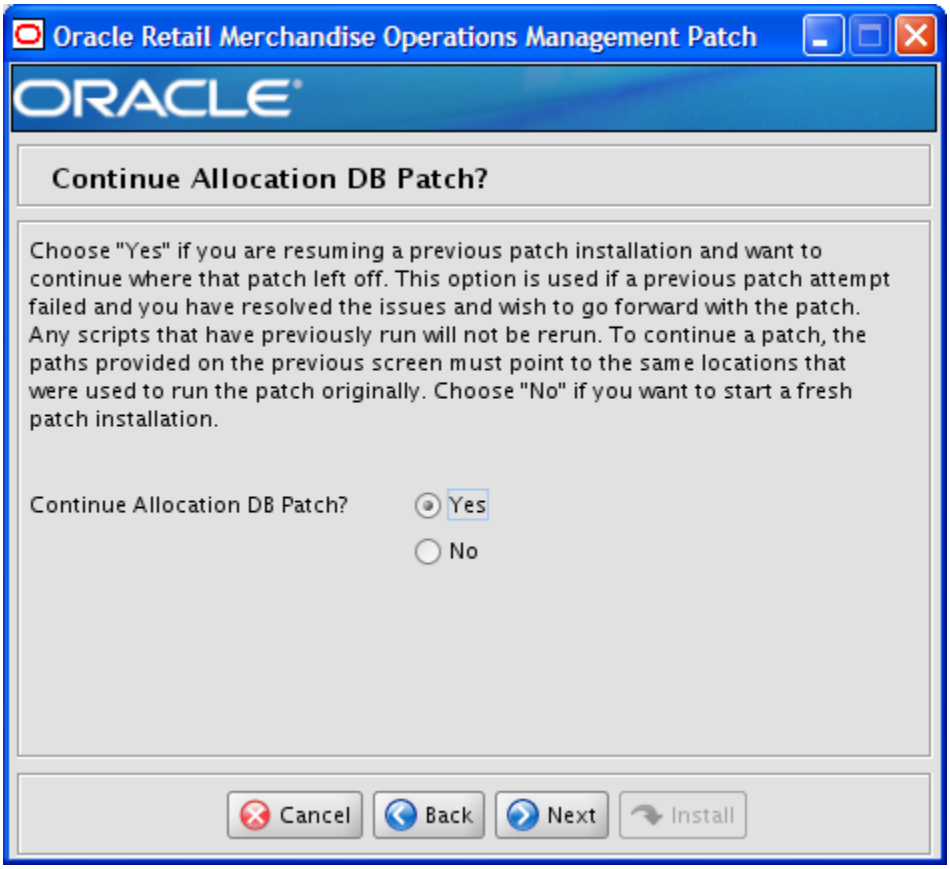
Field Title	Continue ReIM DB Patch?
Field Description	The patch process allows you to continue a previously run patch if it stopped before completion or failed. If Yes is selected, any scripts that were previously run for the ReIM patch will be skipped. If No is selected, the patch will start from the beginning. **Note:** To continue a patch, the content of the "processed" directory in the Patch Directory chosen on the previous screen must be the same as it was after the previous patch was stopped. If you choose No, this directory will be cleared, and you will not be able to continue this patch in the future.

Screen: Apply Allocation DB Patch

Field Title	Alloc Patch Directory
Field Description	Provide the directory path to the patch for the Allocation schema you want to install. The installer runs only the patch you provide. The Alloc controller is used to apply the necessary Allocation patches to the Allocation schema. **Note:** The directory you choose must contain an alloc_controller.ksh file.
Example	/path/to/rms/dbschemapatch/mom-dbpatch for all 13.2.x patches **Note:** The patch option is intended for patches starting with 13.2.

Field Title	Alloc in RMS Patch Directory
Field Description	Provide the directory path to the Allocation patch for the RMS Schema you want to install. The installer runs only the patch you provide. The Alloc_RMS controller is used to apply the necessary Allocation patches to the RMS schema **Note:** The directory you choose must contain an alloc_rms_controller.ksh file.
Example	/path/to/rms/dbschemapatch/mom-dbpatch for all 13.2.x patches **Note:** The patch option is intended for patches starting with 13.2.

Screen: Continue Allocation DB Patch



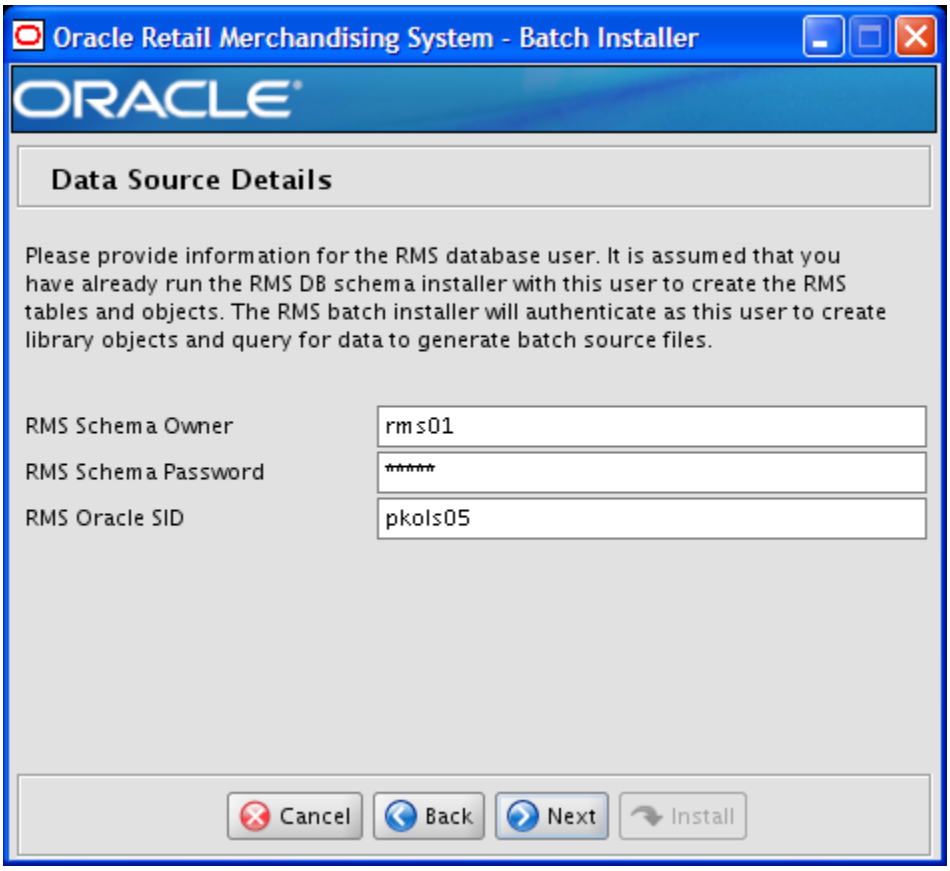
Field Title	Continue Allocation DB Patch?
Field Description	The patch process allows you to continue a previously run patch if it stopped before completion or failed. If Yes is selected, any scripts that were previously run for the Allocation patch in the Allocation and RMS schemas will be skipped. If No is selected, the patch will start from the beginning. **Note:** To continue a patch, the content of the "processed" directories in the Alloc Patch Directory and Alloc in RMS Patch Directory chosen on the previous screen must be the same as it was after the previous patch was stopped. If you choose No, this directory will be cleared, and you will not be able to continue this patch in the future.

B

Appendix: RMS Batch Installer Screens

You need the following details about your environment for the installer to successfully compile and install the RMS batch programs. Depending on the options you select, you may not see some screens or fields.

Screen: DataSource Details

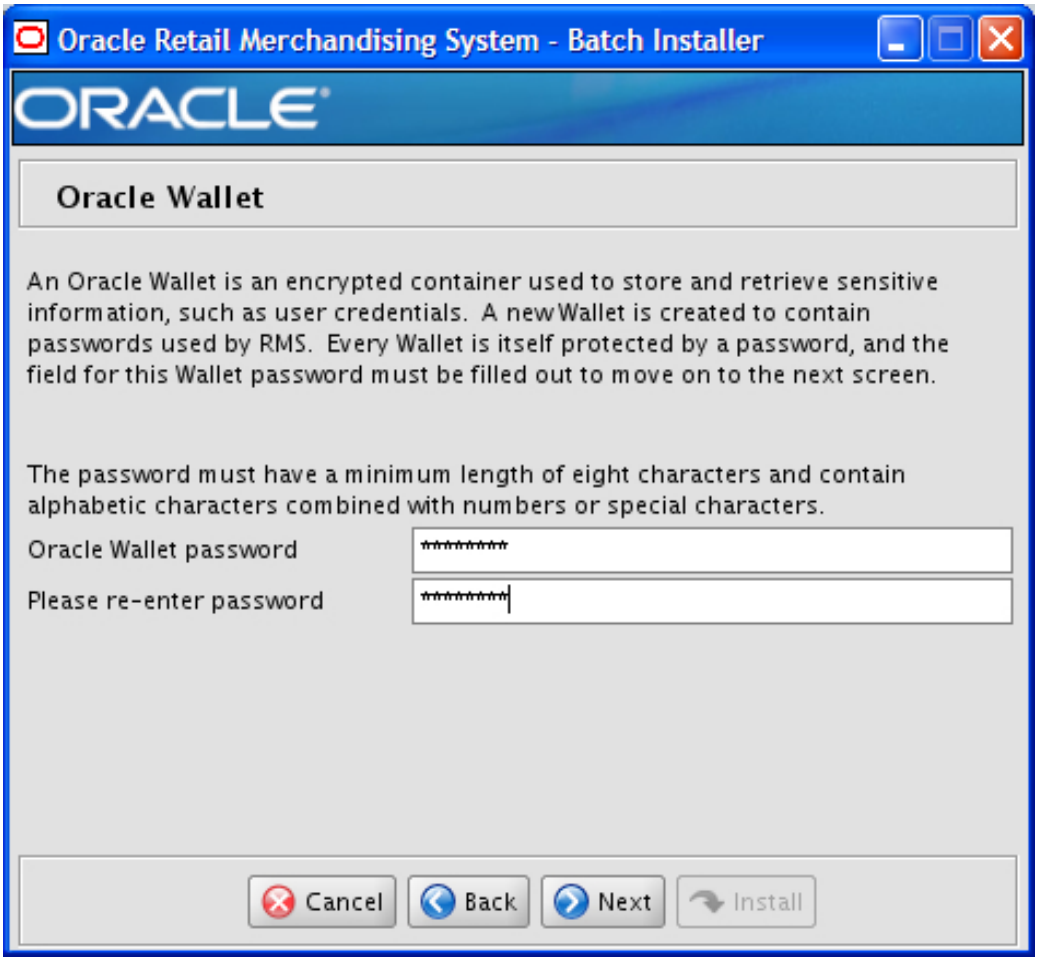


Field Title	RMS Schema Owner
Field Description	Provide the RMS database user here. The installer will log into the database as this user to create RMS library objects and query for data to generate batch source files. This user must already exist in the database and have the RMS tables installed.
Example	rms01

Field Title	RMS Schema Password
Field Description	Database password for the RMS Schema Owner.

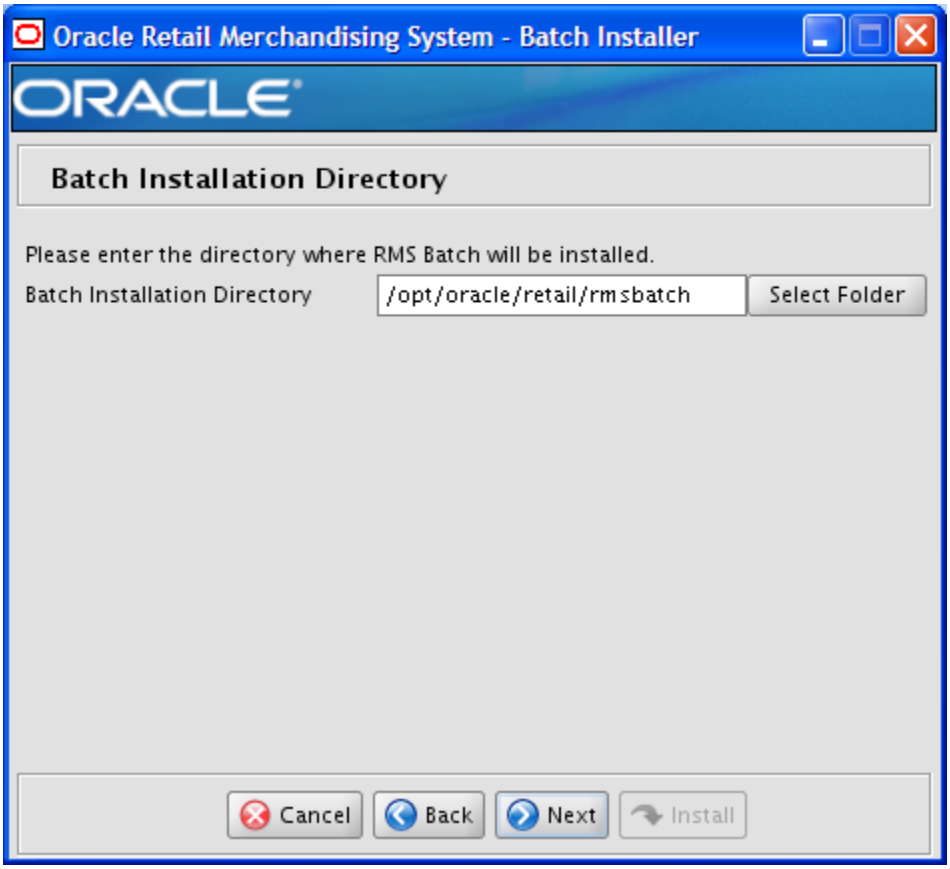
Field Title	RMS Oracle SID
Field Description	Oracle system identifier for the database where RMS will be installed
Example	pkols05

Screen: Oracle Wallet



Field Title	Oracle Wallet password
Field Description	This is the password for the wallet that will store the database credentials that were supplied in the previous screen.

Screen: Batch Installation Directory



Field Title	Batch Installation Directory
Field Description	Location where the installer will install the batch source and then compile it. This is the permanent location for the RMS batch programs.
Example	/opt/oracle/retail/rmsbatch

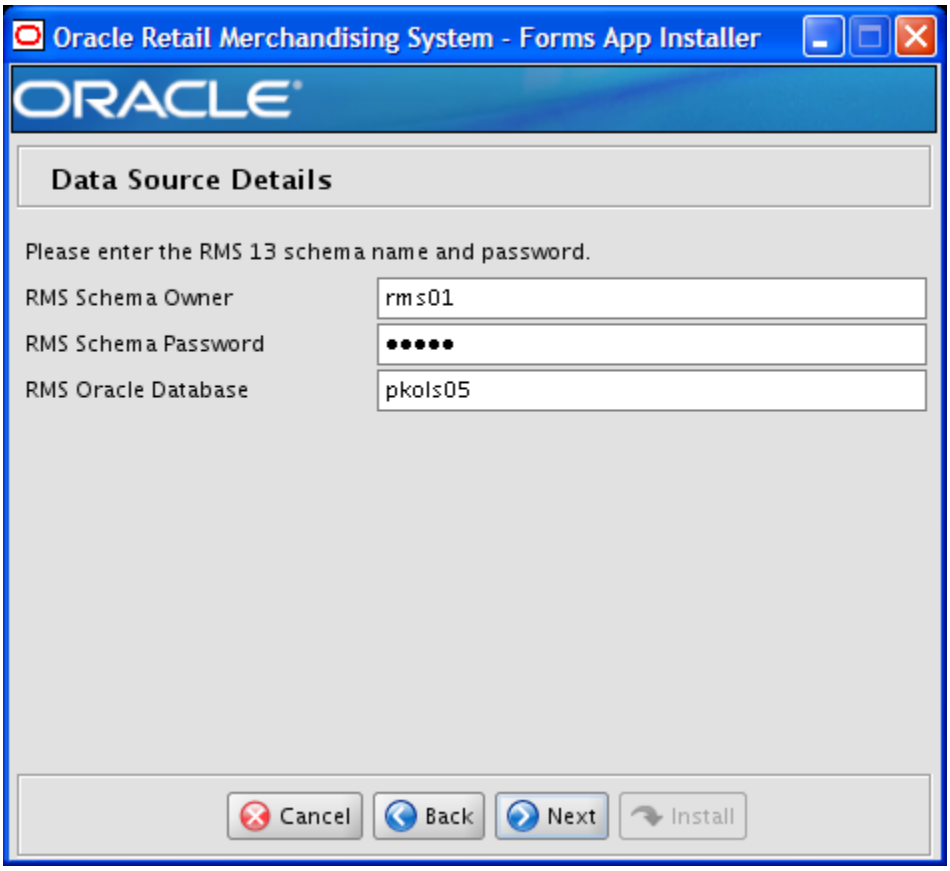
C

Appendix: RMS Application Installer Screens

Screen: Oracle Customer Information

For information about this screen, see the "Oracle Configuration Manager" section in the *Oracle Configuration Manager Installer Guide*.

Screen: Data Source Details



Field Title	RMS Schema Owner
Field Description	This is the same username that was used during the RMS Database Schema Installer.
Example	rms01

Field Title	RMS Schema Password
Field Description	This is the same password that was used during the RMS Database Schema Installer.

Field Title	RMS Oracle SID
Field Description	This is the same Oracle SID that was used during the RMS Database Schema Installer.
Example	pkols05

Screen: Oracle Wallet

Field Title	Oracle Wallet password
Field Description	This is the password for the wallet that will store the database credentials that were supplied in the previous screen.

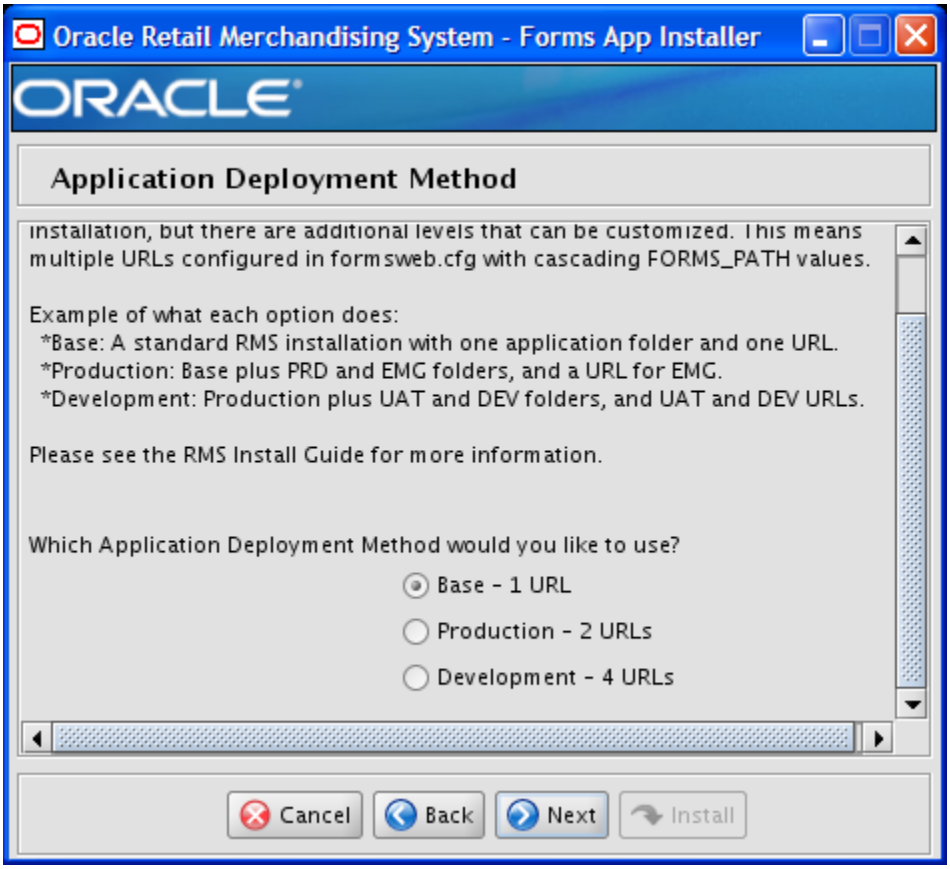
Screen: Installation Name

Field Title	Installation Name
Field Description	This value is used in conjunction the Oracle Configuration Manager (OCM). It gives the installation a unique name so the OCM can identify different installations of RMS in the same WebLogic instance.
Example	rms13inst

Screen: Application Installation Directory

Field Title	Application Installation Directory
Field Description	The location where the RMS Application (toolset, forms and reports) will be installed. The RMS $MMHOME path will be a subdirectory of this directory, named base.
Example	/u00/webadmin/rms13inst

Screen: Application Deployment Method

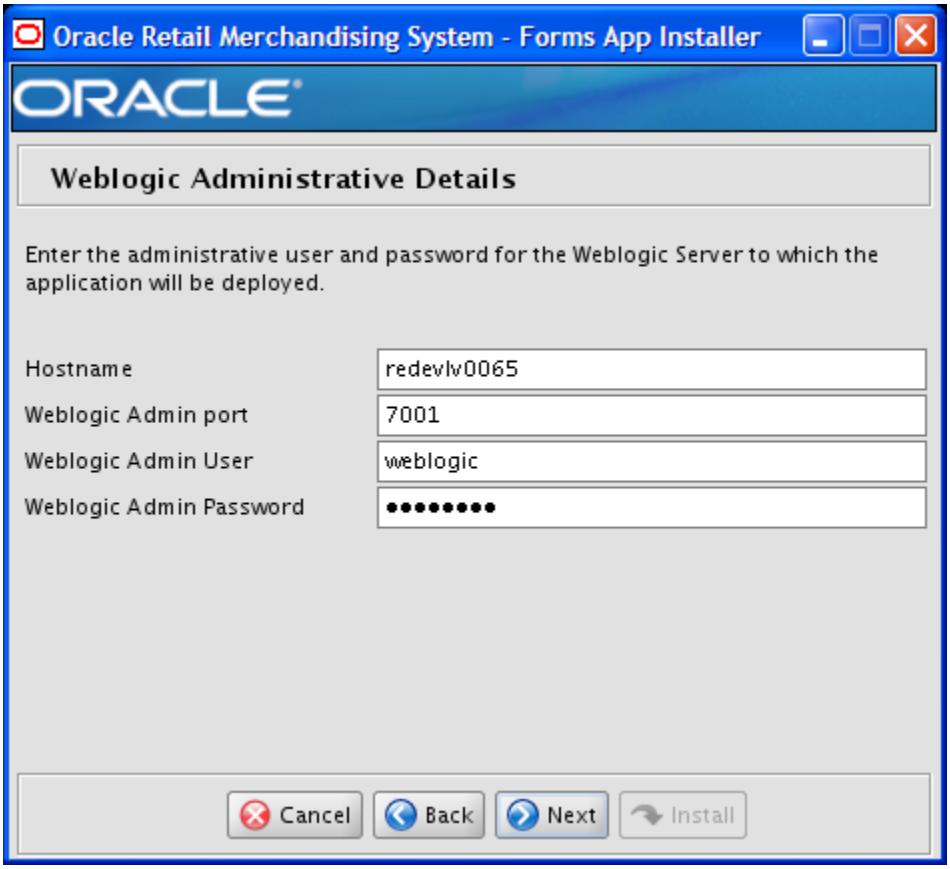


Field Title	Which Environment Deployment Method would you like to use
Field Description	Select the Application Deployment Method you would like. See "Appendix: Application Deployment Method."
Example	Base – 1 URL

Screen: WebLogic Configuration

Field Title	Configure WebLogic
Field Description	Make the necessary configurations to the WebLogic server to be able to run RMS forms. If you choose No, these configurations should be done manually. **Note:** If you rerun the installer, and choose to check the box in the installer screens, Configure WebLogic, you may need to clean up duplicate entries in the WebLogic formsweb.cfg file.

Screen: WebLogic Administrative Details



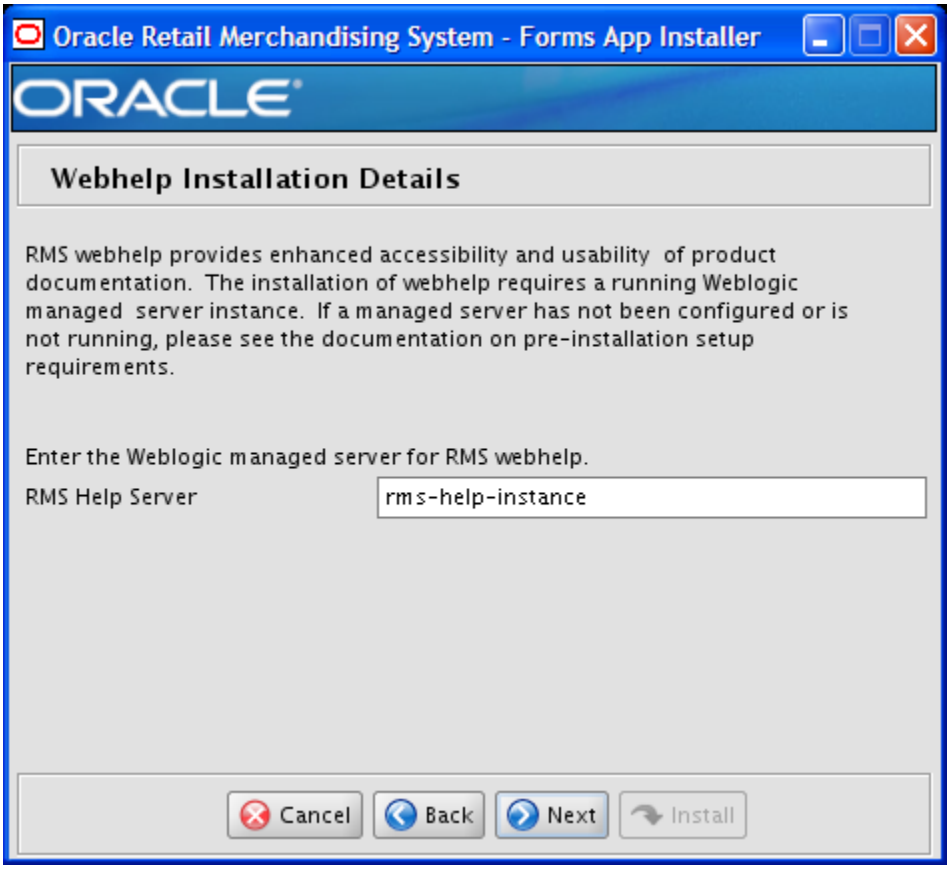
Field Title	Hostname
Field Description	Hostname of the application server
Example	redevlv0065

Field Title	WebLogic Admin port
Field Description	Port number of the WebLogic AdminServer
Example	7001

Field Title	WebLogic Admin User
Field Description	Username of the admin user for WebLogic instance to which the RMS Webhelp application is being deployed.
Example	weblogic

Field Title	WebLogic Admin Password
Field Description	Password for the WebLogic admin user. You chose this password when you created the WebLogic instance.

Screen: Webhelp Installation Details



Field Title	WebLogic Help Server
Field Description	The WebLogic managed server that was created for the RMS Webhelp application.
Example	rms-help-instance

D

Appendix: Installer Silent Mode

In addition to the GUI and text interfaces of the RMS installer, there is a silent mode that can be run. This mode is useful if you wish to run a repeat installation without retyping the settings you provided in the previous installation. It is also useful if you encounter errors in the middle of an installation and wish to continue.

The installer runs in two distinct phases. The first phase involves gathering settings from the user. At the end of the first phase, a properties file named ant.install.properties is created with the settings that were provided. Then the second phase begins, where this properties file is used to provide your settings for the installation.

To skip the first phase and re-use the ant.install.properties file from a previous run, follow these instructions:

1. Edit the ant.install.properties file and correct any invalid settings that may have caused the installer to fail in its previous run.
2. Look for duplicate properties in the ant.install.properties file. Some properties are set on multiple pages to ensure default values when a page is only displayed under certain conditions. For example, if there are two instances of input.property.name, remove all but the last one.
3. Run the installer again with the **silent** argument.

> **Example:** install.sh silent

E

Appendix: URL Reference

This appendix is a URL reference.

JDBC URL for a Database

Used by the Java application and by the installer to connect to the database.

Thick Client Syntax: jdbc:oracle:oci:@<sid>

<sid>: system identifier for the database

Example: jdbc:oracle:oci:@mysid

Thin Client Syntax: jdbc:oracle:thin:@<host>:<port>:<sid>

<host>: hostname of the database server

<port>: database listener port

<sid>: system identifier for the database

Example: jdbc:oracle:thin:@myhost:1521:mysid

LDAP Server URL

Used by the Java application to connect to the LDAP directory.

Syntax: ldap://<host>:<port>

<host>: hostname of the directory server

<port>: LDAP server port

Example: ldap://myhost:389

JNDI Provider URL for an Application

Used by the application client to access the application running in the server. Also used by other applications for server-to-server calls.

WebLogic:

Syntax: t3://<host>:<port>:/<app>

- <host>: hostname of the WebLogic environment
- <port>: Port of the managed server to which rpm has been deployed. This can be found in the <WEBLOGIC_DOMAIN_HOME>/config/config.xml file.
- <app>: Deployment name for the application.

Example: t3://myhost:8001/rpm13

Note: The JNDI provider URL can have a different format depending on your cluster topology. Consult the WebLogic documentation

WebSphere:
Syntax: iiop://<host>:<port>
<host>: hostname of the WebSphere environment
<port>: BOOTSTRAP port of the WebSphere server that is running the application.
Example: iiop://myhost:2809

F

Appendix: Common Installation Errors

This appendix provides some common errors encountered during installation of RMS.

GUI Mode Crashes when Installing on AIX 7

Symptom:

There is a known issue with the installer on AIX7.1. The installer in GUI mode will crash when it communicates with Oracle database, and produces two binary dump files (core.<timestamp>.dmp, Snap.<timestamp>.trc) and a javacore text file (javacore.<timestamp>.txt).

Solution:

As a workaround, please run the installer in text mode (ksh install.sh text) or silent mode (ksh install.sh silent).

Database Installer Hangs on Startup

Symptom:

When the database schema installer is run, the following is written to the console and the installer hangs indefinitely:

```
Running pre-install checks
Running tnsping to get listener port
```

Solution:

The installer startup script is waiting for control to return from the **tnsping** command, but tnsping is hanging. Type Control+C to cancel the installer, and investigate and solve the problem that is causing the **tnsping <sid>** command to hang. This can be caused by duplicate database listeners running.

Unreadable Buttons in the Installer

If you are unable to read the text within the installer buttons, it probably means that your JAVA_HOME is pointed to a pre-1.4.2 JRE or JDK.

Set JAVA_HOME with the appropriate JDK (the same JDK used by the WebLogic server.

Warning: Could not create system preferences directory

Symptom:

The following text appears in the installer Errors tab:

```
May 22, 2006 11:16:39 AM java.util.prefs.FileSystemPreferences$3 run
WARNING: Could not create system preferences directory. System preferences are
unusable.
May 22, 2006 11:17:09 AM java.util.prefs.FileSystemPreferences
checkLockFile0ErrorCode
WARNING: Could not lock System prefs. Unix error code -264946424.
```

Solution:

This is related to Java bug 4838770. The /etc/.java/.systemPrefs directory may not have been created on your system. See http://bugs.sun.com for details.

This is an issue with your installation of Java and does not affect the Oracle Retail product installation.

Warning: Couldn't find X Input Context

Symptom:

The following text appears in the console window during execution of the installer in GUI mode:

```
Couldn't find X Input Context
```

Solution:

This message is harmless and can be ignored.

Unresponsive Country and Currency Drop-Downs

Symptom:

In GUI mode, when you click on the drop-down list selection for the primary country or currency, the list does not appear, and this message appears in the console window:

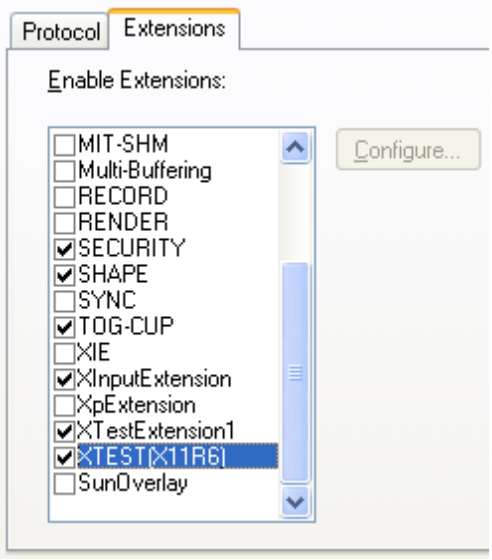
```
XTEST extension not installed on this X server: Error 0
```

Solution:

To run the RMS installer in GUI mode you must have the XTEST extension enabled in your X server.

To enabling XTEST in Exceed, do the following:

1. Open Xconfig to edit Exceed configuration.
2. Go to the X Server Protocol settings.
3. Click the Extensions tab.
4. Make sure that the XTEST extension is selected:



5. Restart the X Server and re-run the RMS installer.

Could not execl robot child process: Permission denied

Symptom:

When opening a drop-down list in GUI mode of the RMS installer, the installer freezes up and displays the following message in the console:

```
Couldn't execl robot child process: Permission denied
```

Solution:

As the owner of the database ORACLE_HOME (i.e. ***oracle***), grant execute permissions to the awt_robot* files under $ORACLE_HOME/jdk/jre/lib. The database schema installer uses $ORACLE_HOME/jdk for its JAVA_HOME.

Example (using SUN Solaris):

```
chmod a+x $ORACLE_HOME/jdk/jre/lib/sparc/awt_robot
chmod a+x $ORACLE_HOME/jdk/jre/lib/sparcv9/awt_robot
```

ConcurrentModificationException in Installer GUI

Symptom:

In GUI mode, the errors tab shows the following error:

```
java.util.ConcurrentModificationException
           at
java.util.AbstractList$Itr.checkForComodification(AbstractList.java:448)
           at java.util.AbstractList$Itr.next(AbstractList.java:419)
… etc
```

Solution:

You can ignore this error. It is related to third-party Java Swing code for rendering of the installer GUI and does not affect the retail product installation.

FRM-30064: Unable to parse statement select while compiling fm_ituda.fmb

Symptom:

When running the application installer you get the following error:

```
FRM-30064: Unable to parse statement select vu.uda_desc,  vu.uda_id  from v_uda vu
where get_primary_lang = get_user_lang  and vu.display_type = 'LV' union all
select nvl(t.translated_value, vu.uda_desc),  vu.uda_id  from tl_shadow t,  v_uda
vu  where get_primary_lang != get_user_lang and upper(vu.uda_desc) = t.key(+) and
get_user_lang = t.lang(+) and vu.display_type = 'LV' order by 1.
ORA-28112: failed to execute policy function
Record Group RG_UDA_LOV
Form: FM_ITUDALST

FRM-30085: Unable to adjust form for output.

Form not created
```

Solution:

Disable the database filter policies by running drop_filter_policy.sql, run the application installer again and then run add_filter_policy.sql. Both files can be located with the database installer.

ORA-04031 (unable to allocate memory) error during database schema installation

Symptom:

When running the database schema installer you get the following error one or more times:

```
[ora:sqlplus] alter package
[ora:sqlplus] *
[ora:sqlplus] ERROR at line 1:
[ora:sqlplus] ORA-04031: unable to allocate 92120 bytes of shared memory ("shared
[ora:sqlplus] pool","unknown object","PL/SQL MPCODE","BAMIMA: Bam Buffer")
```

Solution:

There was not enough available memory in the shared pool on the database at the time of compilation. There are several choices to get past this error:

- Log into the database and attempt to recompile invalid objects in the database schema. Subsequent attempts to compile the same object(s) can be successful.
- Have a DBA increase the shared pool size on the database and re-run the installer from scratch on a new schema user.

X Error of failed request: BadWindow (invalid Window parameter)

Symptom:

When compiling forms during the application installation you receive this error one or more times:

```
X Error of failed request:  BadWindow (invalid Window parameter)
  Major opcode of failed request:  18 (X_ChangeProperty)
  Resource id in failed request:  0x1800002
  Serial number of failed request:  432
  Current serial number in output stream:  437
```

Solution:

This error occurs when there are too many requests made to the X server. If this error occurs manually recompile the form.

Example:

```
frmpcmp.sh userid=$UP module_type=form module=FORM_OR_MENU
```

RIB Errors

At random times, the RIB will get certain errors such as GETNXT(?,?,?,?,?,?,?) and/or ORA-21700 object does not exist or is marked for delete. This is very confusing because you may research and find that the object exists and is valid.

You must re-initialize the reference to reference an existing object as follows:

1. Bring down the RIB WebLogic in question
2. Run <STAGING_DIR>/mom-dbpatch/13.2.8/rms/objects/compile_all_user_objects.sql to recompile all objects in the RMS schema.
3. Run another object validate script (ex: inv_obj_comp.sql) until all the invalids have been resolved
4. Bring up the RIB WebLogic in question.

Error Connecting to Database URL

Symptom:

After entering database credentials in the installer screens and hitting next, a message pops up with an error like this:

```
Error connecting to database URL <url> as user <user>
details...
```

The message prevents you from moving on to the next screen to continue the installation.

Solution:

This error occurs when the installer fails to validate the user credentials you have entered on the screen. Make sure that you have entered the credentials properly. If you receive a message similar to this:

```
Error connecting to database URL <url> as user <user>
java.lang.Exception: UnsatisfiedLinkError encountered when using the Oracle driver.
Please check that the library path is set up properly or switch to the JDBC thin client.
```

It may mean that the installer is using the incorrect library path variables for the platform you are installing on. Open the file <STAGING_DIR>/rms/dbschema/common/preinstall.sh and toggle the variable "use32bit" to "true" if it is set to "false" or vice versa. This setting is dependent on the JRE that is being used.

Multi-Threaded OCI Client Dumps Core after Reconnecting To Database

Symptom

If a multi-threaded Oracle client process that uses OCI to connect to a remote database loses connectivity with the database, it tries to reconnect and the client program continues to run. The program then dumps the core with the following stack trace, when Automatic Diagnostic Repository (ADR) is enabled.

```
skgfqio sdbgrfbibf_io_block_file dbgrfrbf_read_block_file dbgrmflrp_read_page
dbgrmblgmp_get_many_pages dbgrmmdrrmd_read_relation_meta_data
dbgrmmdora_open_record_access_full
dbgriporc_openrel_wcreate dbgrip_open_relation_access dbgrip_start_iterator
dbgrip_relation_iterator dbgruprac_read_adrctl...
```

Solution

Oracle Retail recommended you disable ADR (diag_adr_enabled=OFF, a sqlnet.ora parameter) while using multi-threaded OCI/OCCI application. diag_adr_enabled was introduced in Oracle 11g as a new method of tracing ADR. This will dump additional trace details.

Disabling 'diag_adr_enabled' does not disturb any functionality. Therefore, it can safely be unset by doing diag_adr_enabled=off in sqlnet.ora. However, if you still want tracing, you can have following parameters/variables set in sqlnet.ora:

```
trace_level_server=16 -- for server side NET tracing
trace_level_client=16 -- for client side NET tracing
```

For how to set traditional tracing, see the My Oracle Support document, "SQL*Net, Net8, Oracle Net Services - Tracing and Logging at a Glance" (ID 219968.1).

RMS DB Installer Fails on s11716147a_cost_event_reclass.sql

Symptom

When running the RMS 13.2.1 database patch, it fails on the file s11716147a_cost_event_reclass.sql:

```
[exec] Executing file s11716147a_cost_event_reclass.sql
     [exec] ORA Error while executing s11716147a_cost_event_reclass.sql
   [delete] Deleting directory
/vol.rtk/pkg_mocks/rms1322/rmsdb/rms/dbschemapatch/dblogs/.wallet

BUILD FAILED

/vol.rtk/pkg_mocks/rms1322/rmsdb/rms/dbschemapatch/build.xml:571: The following
error occurred while executing this line:
/vol.rtk/pkg_mocks/rms1322/rmsdb/rms/dbschemapatch/build.xml:367: The following
error occurred while executing this line:
/vol.rtk/pkg_mocks/rms1322/rmsdb/rms/dbschemapatch/build.xml:276:
/vol.rtk/pkg_mocks/rms1322/rmsdb/rms/dbschemapatch/mom-dbpatch/rms_controller.ksh
has failed.
```

If you look in the file s11716147a_cost_event_reclass.err, it reports this error message:

```
Altering table COST_EVENT_RECLASS
ALTER TABLE COST_EVENT_RECLASS MODIFY ITEM VARCHAR2(25) NOT NULL
*
ERROR at line 1:
ORA-02296: cannot enable (RMS132MOCK.) - null values found
```

Solution

You must make sure all the cost events for RECLASS are processed and purged before the 13.2.1 patch is applied. The DBC script s11716147a_cost_event_reclass.sql is altering the table COST_EVENT_RECLASS and adding PK on it. Ensure that before running this DBC the table is empty and the existing data is already processed and purged in nightly batch. The installation can be rerun after this is done. You should choose to continue the previous install when rerunning the installer.

GUI Screens Fail to Open When Running Installer

Symptom

When running the installer in GUI mode, the screens fail to open and the installer ends, returning to the console without an error message. The ant.install.log file contains this error:

```
Fatal exception: Width (0) and height (0) cannot be <= 0
java.lang.IllegalArgumentException: Width (0) and height (0) cannot be <= 0
```

Solution

This is an error encountered when Antinstaller is used in GUI mode with certain X Servers. To workaround this issue, copy ant.install.properties.sample to ant.install.properties and rerun the installer.

Installer fails with sun.security.validator.KeyStores exception

Symptom

Installer first throws the error

```
Exception in thread "main" java.lang.NoClassDefFoundError:
sun.security.validator.KeyStores
```

And then on continuing aborts with the same error.

Solution

OCM does not work on AIX 7.1. The workaround after facing this exception is to recreate the STAGING_DIR and then delete the **"retail-OCM-wls.zip"** file present in STAGING_DIR/rms directory, before running the installer.

Deployed BI Publisher Application fails to start up

Symptom

After deploying the BI Publisher into the WebLogic, the application fails to start up with a "missing CipherException.class" exception.

Solution

Add CLASSPATH to point to the osdt_core3.jar which resides in $WLS_HOME/oracle_common/oui/jlib/lib.

Go to WebLogic Adminconsole > BI Publisher managed server that is used for deploying BI Publisher. Click the Server Start tab. In the Class Path box, add the following (values are examples): /u00/webadmin/product/WLS/oracle_common/oui/jlib/lib

Installer fails because ALLOC_WRAPPER_SQL is invalid after 13.2.8 Patch is applied

If you are patching Allocation, after finishing the database patch installer you will need to manually create some synonyms from the RMS schema to the Allocation schema, and grants from the Allocation schema to the RMS schema. Verify that the following synonyms exist in the RMS schema and that they are pointing to the table with the same name in the Allocation schema:

- ALC_DEFAULT_CHRGS_TEMP
- ALC_ON_HAND_QTY_TEMP
- ALC_RLOH_TEMP

If any of these synonyms do not exist, run the following command as the RMS schema owner for each table to create the missing synonym:

```
create or replace synonym <RMS schema>.<TABLE NAME> for <ALLOC
schema>.<TABLE_NAME>
```

For example:

```
create or replace synonym RMS01.ALC_RLOH_TEMP for ALLOC01.ALC_RLOH_TEMP
```

In addition, grant insert permissions on the alc_on_hand_qty_temp table in the Allocation schema to the RMS schema. Run the following command as the Allocation user to grant the permission:

```
grant insert on ALC_ON_HAND_QTY_TEMP to <RMS schema>
```

For example:

```
grant insert on ALC_ON_HAND_QTY_TEMP to RMS01
```

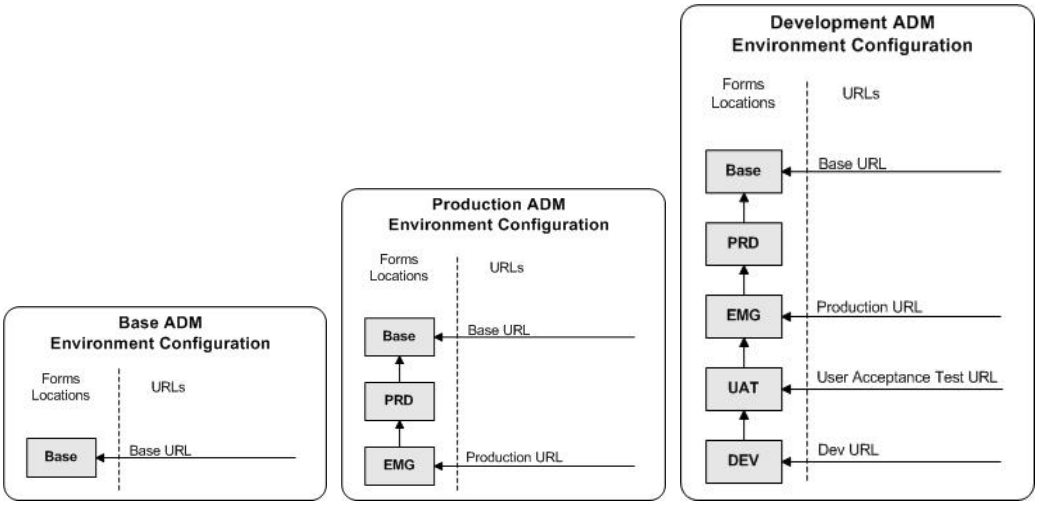
G

Appendix: Application Deployment Method

The RMS installer provides the option to configure multiple application deployment methods. This is a setup where there is still a single primary RMS installation, but there are additional levels where customization can occur. This means multiple URLs configured in formsweb.cfg with cascading FORMS_PATH values.

The installer provides three choices for cascading environment configuration:

- Base: A standard RMS base installation with one application installation folder, and one URL.
- Production: Base plus two additional forms directories for PRD and EMG and an additional URL for EMG.
- Development: Production plus two additional forms directories for UAT and DEV and two additional URLs for UAT and DEV.



The above diagrams show how the application deployment method environment configurations are set up in the forms installation.

The installer creates the set of URLs, and empty directories for the other environments. All forms installed by this installer are placed in the "Base" environment. We are simply laying down the structure for customizations and fixes that the user can make after installation is complete.

H

Appendix: Single Sign-On Resource Access Descriptors

Oracle Forms applications such as RMS use database connections for authentication and authorization purposes. Oracle Single Sign-On, however, uses the Oracle Internet Directory (OID) user ID and password for this purpose. The Forms framework maps OID user IDs to database connections via information stored in Resource Access Descriptors (RADs). A user will have one RAD for each application accessed. RADs may be created by an administrator or by an LDIF script. Depending on the Oracle Internet Directory and/or the formsweb.cfg configuration, RADs may also be created by the user.

A user is prompted for the database connection information whenever formsweb.cfg file specifies ssoMode = true and createDynamicResources = true for an application and no valid RAD exists. RADs may become invalid when passwords have expired or have been changed.

RADs may be created by administrators or users via the Delegated Administration Services application.

> **Note:** Users can create new RADs only if one or more RADs already exist.

RADs may be created and through LDIF scripts as well. For information, see the My Oracle Support document, "How to Create and Copy SSO User Resources (RADs) (ID 244526.1).

I

Appendix: Oracle Single Sign-On for WebLogic

Single Sign-On (SSO) is a term for the ability to sign onto multiple Web applications via a single user ID/Password. There are many implementations of SSO. Oracle currently provides two different implementations: Oracle Single Sign-On (OSSO), and Oracle Access Manager (provides more comprehensive user access capabilities).

Most, if not all, SSO technologies use a session cookie to hold encrypted data passed to each application. The SSO infrastructure has the responsibility to validate these cookies and, possibly, update this information. The user is directed to log on only if the cookie is not present or has become invalid. These session cookies are restricted to a single browser session and are never written to a file.

Another facet of SSO is how these technologies redirect a user's Web browser to various servlets. The SSO implementation determines when and where these redirects occur and what the final screen shown to the user is.

Most SSO implementations are performed in an application's infrastructure and not in the application logic itself. Applications that leverage infrastructure managed authentication (such as deployment specifying Basic or Form authentication) typically have little or no code changes when adapted to work in an SSO environment.

What Do I Need for Oracle Single Sign-On?

The nexus of an Oracle Single Sign-On system is the Oracle Identity Management Infrastructure installation. This consists of the following components:

- An Oracle Internet Directory (OID) LDAP server, used to store user, role, security, and other information. OID uses an Oracle database as the back-end storage of this information.
- An Oracle HTTP Server 11g Release 1 as a front end to the Oracle WebLogic Server. The Oracle HTTP Server is included in the Oracle Web Tier Utilities 11g Release 1 (11.1.1).
- An Oracle Single Sign-On Plug-in, used to authenticate the user and create the OSSO session cookie. This is available in the Oracle Fusion Middleware 11g Web Tier Utilities (11.1.1.7) package. For Oracle Forms applications like RMS and RWMS, HTTP server will be used.
- Oracle Directory Services Manager (ODSM) application in OIM11g, used to administer users and group information. This information may also be loaded or modified via standard LDAP Data Interchange Format (LDIF) scripts.
- Additional administrative scripts for configuring the OSSO system and registering HTTP servers.

Additional WebLogic managed servers will be needed to deploy the business applications leveraging the OSSO technology.

Can Oracle Single Sign-On Work with Other SSO Implementations?

Yes, OSSO has the ability to interoperate with many other SSO implementations, but some restrictions exist.

Oracle Single Sign-on Terms and Definitions

The following terms apply to single sign-on.

Authentication

Authentication is the process of establishing a user's identity. There are many types of authentication. The most common authentication process involves a user ID and password.

Dynamically Protected URLs

A Dynamically Protected URL is a URL whose implementing application is aware of the OSSO environment. The application may allow a user limited access when the user has not been authenticated. Applications that implement dynamic OSSO protection typically display a Login link to provide user authentication and gain greater access to the application's resources.

Oracle Identity Management (OIM) and Oracle Access Manager (OAM) Oracle Access Manager (OAM) for 11g

Oracle Identity Management (OIM) 11g includes Oracle Internet Directory and ODSM. Oracle Access Manager (OAM) 11g should be used for SSO using osso agent. Oracle Forms 11g contains Oracle HTTP server and other Retail Applications will use WebTier11g for HTTP.

MOD_OSSO

mod_osso is an Apache Web Server module an Oracle HTTP Server uses to function as a partner application within an Oracle Single Sign-On environment. The Oracle HTTP Server is based on the Apache HTTP Server.

MOD_WEBLOGIC

mod_WebLogic operates as a module within the HTTP server that allows requests to be proxied from the Apache HTTP server to the WebLogic server.

Oracle Internet Directory

Oracle Internet Directory (OID) is an LDAP-compliant directory service. It contains user ids, passwords, group membership, privileges, and other attributes for users who are authenticated using Oracle Single Sign-On.

Partner Application

A partner application is an application that delegates authentication to the Oracle Identity Management Infrastructure. One such partner application is the Oracle HTTP Server (OHS) supplied with Oracle Forms Server or WebTier11g Server if using other Retail Applications other than Oracle Forms Applications. OHS or WebTier uses the MOD_OSSO module to configure this functionality.

All partner applications must be registered with Oracle Access Manager (OAM) 11g if using OAM11g for SSO implementation. An output product of this registration is a configuration file the partner application uses to verify a user has been previously authenticated.

Realm

A Realm is a collection users and groups (roles) managed by a single password policy. This policy controls what may be used for authentication (for example, passwords, X.509 certificates, and biometric devices). A Realm also contains an authorization policy used for controlling access to applications or resources used by one or more applications.

A single OID can contain multiple Realms. This feature can consolidate security for retailers with multiple banners or to consolidate security for multiple development and test environments.

Statically Protected URLs

A URL is considered to be Statically Protected when an Oracle HTTP server is configured to limit access to this URL to only SSO authenticated users. Any attempt to access a Statically Protected URL results in the display of a login page or an error page to the user.

Servlets, static HTML pages, and JSP pages may be statically protected.

> **Note:** Dynamically Protected URL and Statically Protected URL are within the context of the Oracle Software Security Assurance (OSSA). The static protection for URLs is a common JEE feature.

What Single Sign-On is not

Single Sign-On is NOT a user ID/password mapping technology.

However, some applications can store and retrieve user IDs and passwords for non-SSO applications within an OID LDAP server. An example of this is the Oracle Forms Web Application framework, which maps OSSO user IDs to a database logins on a per-application basis.

How Oracle Single Sign-On Works

Oracle Single Sign-On involves a couple of different components. These are:

- The Oracle Single Sign-On (OSSO) servlet, which is responsible for the back-end authentication of the user.
- The Oracle Internet Directory LDAP server, which stores user IDs, passwords, and group (role) membership.
- The Oracle HTTP Server associated with the Web application, which verifies and controls browser redirection to the OSSO servlet.
- If the Web application implements dynamic protection, then the Web application itself is involved with the OSSO system.

Statically Protected URLs

When an unauthenticated user accesses a statically protected URL, the following occurs:

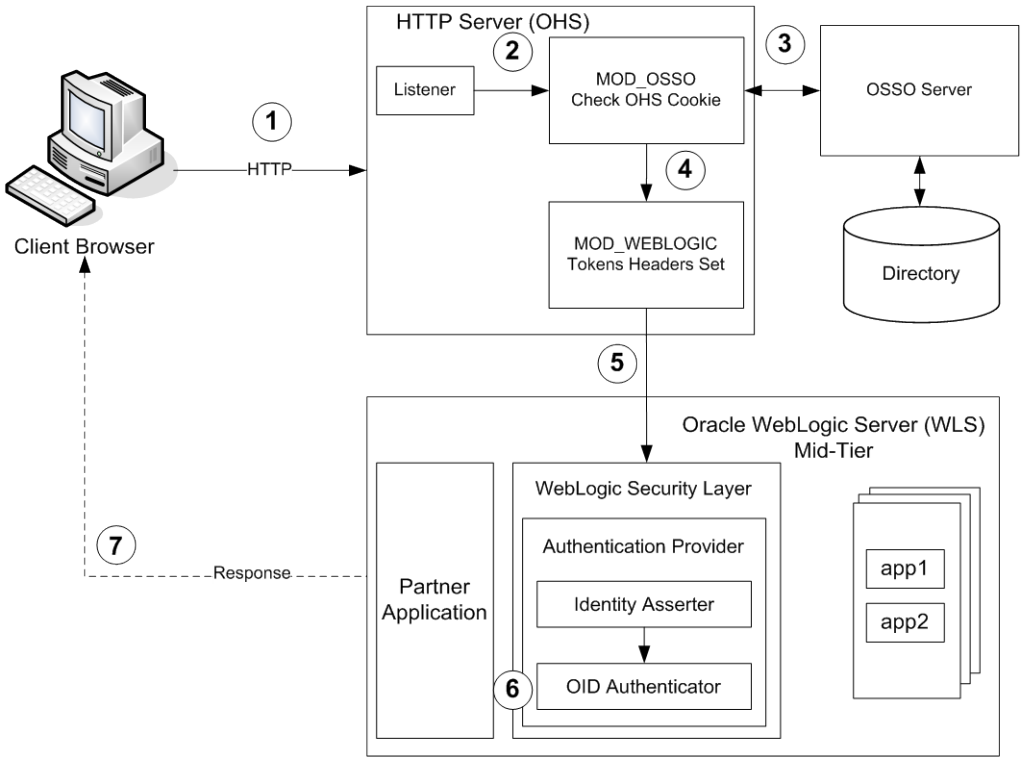
1. The user's Web browser makes an HTTP request to a protected URL serviced by the Oracle HTTP Server (OHS).
2. The Oracle HTTP Server processses the request and routes it to the mod_oss module.
3. This module determines whether the user is already authenticated. If the authentication is required, it directs the browser to the OSSO server. The OSSO server checks for a secure cookie containing the authentication information. If the cookie is not found, the following occurs:
   a. The OSSO servlet determines the user must authenticate, and displays the OSSO login page.
   b. The user must sign in via a valid user ID and password. If the OSSO servlet has been configured to support multiple Realms, a valid realm must also be entered. The user ID, password, and realm information is validated against the Oracle Internet Directory LDAP server. The browser is then redirected back to the Oracle HTTP Server with the encrypted authentication credentials. It does NOT contain the user's password.
4. The mod_osso module then decrypts the user credentials and sets HTTP headers with relevant user attributes, marking the user's session as authenticated.
5. The mod_WebLogic module (within the Oracle HTTP Server) then forwards the request to the Oracle WebLogic Server.
6. The Oracle WebLogic Server then invokes the configured authentication providers that decode the headers and provide the user's role membership. In an OSSO implementation, ensure that the OSSO Identity Asserter is invoked and Oracle Internet Directory (OID) Authenticator is executed to provide the user's role membership.
7. Once the authentication is established, the relevant application logic is initiated and the response is sent back to the user through the Oracle HTTP Server.
   Because the Web browser session is now authenticated, subsequent requests in that session are not redirected to the OSSO server for authentication.

Dynamically Protected URLs

When an unauthenticated user accesses a dynamically protected URL, the following occurs:

1. The user's Web browser makes an HTTP request to a protected URL serviced by the Oracle HTTP Server (OHS). The Oracle HTTP server recognizes the user has not been authenticated, but allows the user to access the URL.
2. The application determines the user must be authenticated and send the Oracle HTTP Server a specific status to begin the authentication process.
3. The Oracle HTTP Server processes the request and routes it to the mod_oss module.
4. This module determines whether the user is already authenticated. If the authentication is required, it directs the browser to the OSSO server. The OSSO server checks for a secure cookie containing the authentication information. If the cookie is not found, the following occurs:
   a. The OSSO servlet determines the user must authenticate, and displays the OSSO login page.
   b. The user must sign in via a valid user ID and password. If the OSSO servlet has been configured to support multiple Realms, a valid realm must also be entered. The user ID, password, and realm information is validated against the Oracle Internet Directory LDAP server. The browser is then redirected back to the Oracle HTTP Server with the encrypted authentication credentials. It does NOT contain the user's password.
5. The mod_osso module then decrypts the user credentials and sets HTTP headers with relevant user attributes, marking the user's session as authenticated.
6. The mod_WebLogic module (within the Oracle HTTP Server) then forwards the request to the Oracle WebLogic Server.
7. The Oracle WebLogic Server then invokes the configured authentication providers that decode the headers and provide the user's role membership. In an OSSO implementation, ensure that the OSSO Identity Asserter is invoked and Oracle Internet Directory (OID) Authenticator is executed to provide the user's role membership.
8. Once the authentication is established, the relevant application logic is initiated and the response is sent back to the user through the Oracle HTTP Server.
   Because the Web browser session is now authenticated, subsequent requests in that session are not redirected to the OSSO server for authentication.

Single Sign-on Topology



Installation Overview

Installing Oracle Single Sign-On using OAM11g requires installation of the following:

1. Oracle Internet Directory (OID) ldap server and the Oracle Directory Services Manager. They are typically installed using the Installer of Oracle Identity Management 11gR1 (11.1.1.7). The ODSM application can be used for user and realm management within OID.
2. Oracle Access Manager 11gR1 (11.1.1.7) has to be installed and configured.
3. Additional midtier instances (such as Oracle Forms 11g) for Oracle Retail applications based on Oracle Forms technologies (such as RMS). These instances must be registered with the OAM installed in step 2.
4. Additional application servers to deploy other Oracle Retail applications and performing application specific initialization and deployment activities must be registered with OAM installed in step 2. For additional information on SSO 11g installation, see the Oracle Access Manager and Single Sign-On Whitepaper (My Oracle Support Doc ID 1492047.1).

Infrastructure Installation and Configuration

The Infrastructure installation for OSSO and Oracle Access Manager (OAM) is dependent on the environment and requirements for its use. Deploying an Infrastructure OAS or Oracle Access Manager (OAM) to be used in a test environment does not have the same availability requirements as for a production environment. Similarly, the Oracle Internet Directory (OID) LDAP server can be deployed in a variety of different configurations. See *Oracle Identity Management Installation Guide11g (if using OAM11).*

OID User Data

Oracle Internet Directory is an LDAP v3 compliant directory server. It provides standards-based user definitions out of the box.

The current version of Oracle Single Sign-On only supports OID as its user storage facility. Customers with existing corporate LDAP implementations may need to synchronize user information between their existing LDAP directory servers and OID. OID supports standard LDIF file formats and provides a JNDI compliant set of Java classes as well. Moreover, OID provides additional synchronization and replication facilities to integrate with other corporate LDAP implementations.

Each user ID stored in OID has a specific record containing user specific information. For role-based access, groups of users can be defined and managed within OID. Applications can thus grant access based on group (role) membership saving administration time and providing a more secure implementation.

OID with Multiple Realms

OID and OSSO can be configured to support multiple user Realms. Each realm is independent from each other and contains its own set of user IDs. As such, creating a new realm is an alternative to installing multiple OID and Infrastructure instances. Hence, a single Infrastructure OAS can be used to support development and test environments by defining one realm for each environment.

Realms may also be used to support multiple groups of external users, such as those from partner companies. For more information on Realms, see the *Oracle Internet Directory Administrators Guide*.

User Management

User Management consists of displaying, creating, updating or removing user information. There are two basic methods of performing user management: LDIF scripts and the Oracle Directory Services Manager (ODSM) available for OID11g.

ODSM

Oracle Directory Services Manager (ODSM) is a Web-based application used in OID11g is designed for both administrators and users which enables you to configure the structure of the directory, define objects in the directory, add and configure users, groups, and other entries. ODSM is the interface you use to manage entries, schema, security, adapters, extensions, and other directory features.

LDIF Scripts

Script based user management can be used to synchronize data between multiple LDAP servers. The standard format for these scripts is the LDAP Data Interchange Format (LDIF). OID supports LDIF script for importing and exporting user information. LDIF scripts may also be used for bulk user load operations.

User Data Synchronization

The user store for Oracle Single Sign-On resides within the Oracle Internet Directory (OID) LDAP server. Oracle Retail applications may require additional information attached to a user name for application-specific purposes and may be stored in an application-specific database. Currently, there are no Oracle Retail tools for synchronizing changes in OID stored information with application-specific user stores. Implementers should plan appropriate time and resources for this process. Oracle Retail strongly suggests that you configure any Oracle Retail application using an LDAP for its user store to point to the same OID server used with Oracle Single Sign-On.

J

Appendix: AIX Shared Library Bug Fix

The env_rdbms.mk file for Oracle 10g and higher includes defect 2143531, which was not fixed because there is a workaround. For the workaround, the following changes in bold and italics must need to be made to the $ORACLE_HOME/rdbms/lib/env_rdbms.mk file. Note that changes are made in both the BUILDLIB_WITH_CONTEXT and BUILDLIB_WITH_NO_CONTEXT functions.

```
-----------------------------------------
BUILDLIB_WITH_CONTEXT=generate_export_list() \
{ \
/bin/nm -X32_64 -B -h -g "$$1" | grep -v ' U ' | awk '{print $$3}' | \
egrep -v '^\.|^TOC' | sort | uniq ; \
}; \
generate_import_list() { \
LIB_NAME=$$1; \
IMP_FILE=$$2; \
\
cat ${ORACLE_HOME}/rdbms/lib/xa.imp | head -1 | awk '{print $$0, "." }' >
$${IMP_FILE}; \
/bin/nm -X32_64 -C -B -h -g $${LIB_NAME} | grep ' U ' | grep -v "::" | grep -v "("
| grep -v "\.cc" | awk '{print $$3}' | sed -e "s/\.//g
" | grep -v "^_" >> $${IMP_FILE}; \
}; \
\
generate_import_list "$(OBJS)" $(SHARED_LIBNAME).imp; \
generate_export_list $(OBJS) > $(SHARED_LIBNAME).exp; \
$(LD) -bnoentry -bM:SRE -bE:$(SHARED_LIBNAME).exp -bI:$(SHARED_LIBNAME).imp \
-o $(SHARED_LIBNAME) $(OBJS) -L$(ORACLE_HOME)/lib -lc_r -lm $(LLIBCLNTSH) $(MATHLIB)
-----------------------------------------
BUILDLIB_NO_CONTEXT=generate_export_list() \
{ \
/bin/nm -X32_64 -B -h -g "$$1" | grep -v ' U ' | awk '{print $$3}' | \
egrep -v '^\.|^TOC' | sort | uniq ; \
}; \
generate_import_list() { \
LIB_NAME=$$1; \
IMP_FILE=$$2; \
\
cat ${ORACLE_HOME}/rdbms/lib/xa.imp | head -1 | awk '{print $$0, "." }' >
$${IMP_FILE}; \
/bin/nm -X32_64 -C -B -h -g $${LIB_NAME} | grep ' U ' | grep -v "::" | grep -v "("
| grep -v "\.cc" | awk '{print $$3}' | sed -e "s/\.//g
" | grep -v "^_" >> $${IMP_FILE}; \
}; \
\
generate_import_list "$(OBJS)" $(SHARED_LIBNAME).imp; \
generate_export_list $(OBJS) > $(SHARED_LIBNAME).exp; \
$(LD) -bnoentry -bM:SRE -bE:$(SHARED_LIBNAME).exp -bI:$(SHARED_LIBNAME).imp \
-o $(SHARED_LIBNAME) $(OBJS) -L$(ORACLE_HOME)/lib -lc_r -lm $(LLIBCLNTSH) $(MATHLIB)
```

K

Appendix: RMS RETL Instructions

This Appendix summarizes the RETL program features utilized in the RMS Extractions (RMS ETL). More information about the RETL tool is available in the *Oracle Retail Extract, Transform, and Load Programmer's Guide* and the *Oracle Retail Extract, Transform, and Load Installation Guide* More information about *Oracle Retail Merchandising System Operations Guide*.

Configuration

The following are configuration instructions.

RETL

Before trying to configure and run RMS ETL, install RETL version 10.3 or later which is required to run RMS ETL. Run the "verify_retl" script (included as part of the RETL installation) to ensure that RETL is working properly before proceeding.

RETL User and Permissions

RMS ETL should be installed and run as the RETL user. Additionally, the permissions should be set up as per the RETL Programmer's Guide. RMS ETL reads data, creates, deletes and updates tables. (This is to ensure that weekly sales data is not pulled multiple times on subsequent extractions.) If these permissions are not set up properly, extractions will fail.

Environment variables

In addition to the RETL environment variables (see the *Oracle Retail Extract, Transfer, and Load Programmer's Guide* for the version of RETL), you need to set MMHOME to the base directory for RMS ETL. This is the top level directory that selected during the RMS Batch installation process. So in .kshrc you should add a line like the following:

```
export MMHOME=<base directory for RMS ETL>
```

rmse_config.env

There are a couple variables that will need to change depending upon local settings:

```
export DBNAME=pkols05
export RMS_OWNER=rms01
export BA_OWNER=rms01
```

Also, you will need to set the environment variable PASSWORD in either the rmse_config.env, .kshrc or some other location that can be included via one of those two means. For example, adding this line to the rmse_config.env will cause the password "bogus" to be used to log into the database: `export PASSWORD=pass1`

L

Appendix: Inserting New Languages

A new language was added for the RMS 13.2 release, Croatian (hr). As the RMS dbschema installer does not support inserting new languages that have not already been installed, this section documents how to manually insert new languages as either primary or secondary languages. These steps should be done after installing the RMS 13.2.8 upgrade.

In this section <lang> represents the two or three-letter code for the language you wish to insert. This is the list of supported codes and the languages they represent:

- de - German
- es – Spanish
- el - Greek
- fr – French
- hu – Hungarian
- hr – Croatian
- it – Italian
- ja – Japanese
- ko – Korean
- nl – Dutch
- pl - Polish
- ptb – Brazilian Portuguese
- ru – Russian
- sv – Swedish
- tr - Turkish
- zhs – Simplified Chinese
- zht – Traditional Chinese

Insert Secondary Language Data

To insert secondary language data, complete the following steps.

> **Note:** These scripts are only for customers who wish to have a primary language of English and a secondary language of any combination of the supported languages. The scripts are UTF-8 encoded. We recommend installing them into a database that has been set to UTF-8.

1. Change directories to DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/13.2.8/rms/lang/.
2. Set the sqlplus session so that the encoding component of the NLS_LANG is UTF8. For example AMERICAN_AMERICA.UTF8.

3. Log into sqlplus with the RMS schema and run the following command:
   ```
   SQL> @rms1328_secondary_<lang>.sql
   ```
4. Check the log file rms1328_secondary_<lang>. log for any errors.

> **Note:** Multiple secondary languages can be added to a primary language install of English.

Insert Primary Language Data

To insert primary language data, complete the following steps.

> **Note:** These scripts are only for customers who wish to have a primary language of one of the non-English supported languages. Once you run one of these primary scripts, you will not be able to revert back to English as your primary language. The scripts are UTF-8 encoded. We recommend installing them into a database that has been set to UTF-8.

1. Change directories to DB_PATCH_DIR/rms/dbschemapatch/mom-dbpatch/13.2.8/rms/lang/.
2. Set the sqlplus session so that the encoding component of the NLS_LANG is UTF8. For example AMERICAN_AMERICA.UTF8.
3. Log into sqlplus with the RMS schema and run the following command:
   ```
   SQL> @rms1328_primary_<lang>.sql
   ```
4. Check the log file rms1328_primary_<lang>. log for any errors.

> **Note:** Only one language can be set as the primary language for the system.

M

Appendix: Setting Up Password Stores with Oracle Wallet

As part of an application installation, administrators must set up password stores for database user accounts using Oracle Wallet. These password stores must be installed on the application database side. While the installer handles much of this process, the administrators must perform some additional steps.

A password store for the application and application server user accounts must also be installed; however, the installer takes care of this entire process.

About Password Stores and Oracle Wallet

Oracle databases have allowed other users on the server to see passwords in case database connect strings (username/password@db) were passed to programs. In the past, users could navigate to `ps -ef|grep <username>` to see the password if the password was supplied in the command line when calling a program.

To make passwords more secure, Oracle Retail has implemented the Oracle Software Security Assurance (OSSA) program. Sensitive information such as user credentials now must be encrypted and stored in a secure location. This location is called password stores or wallets. These password stores are secure software containers that store the encrypted user credentials.

Users can retrieve the credentials using aliases that were set up when encrypting and storing the user credentials in the password store. For example, if `username/password@db` is entered in the command line argument and the alias is called `db_username`, the argument to a program is as follows:

```
sqlplus /@db_username
```

This would connect to the database as it did previously, but it would hide the password from any system user.

After this is configured, as in the example above, the application installation and the other relevant scripts are no longer needed to use embedded usernames and passwords. This reduces any security risks that may exist because usernames and passwords are no longer exposed.

When the installation starts, all the necessary user credentials are retrieved from the Oracle Wallet based on the alias name associated with the user credentials.

There are two different types of password stores or wallets. One type is for database connect strings used in program arguments (such as `sqlplus /@db_username`). The other type is for Java application installation and application use.

Setting Up Password Stores for Database User Accounts

After the database is installed and the default database user accounts are set up, administrators must set up a password store using the Oracle wallet. This involves assigning an alias for the username and associated password for each database user account. The alias is used later during the application installation. This password store must be created on the system where the application server and database client are installed.

This section describes the steps you must take to set up a wallet and the aliases for the database user accounts. For more information on configuring authentication and password stores, see the *Oracle Database Security Guide*.

> **Note:** In this section, <wallet_location> is a placeholder text for illustration purposes. Before running the command, ensure that you specify the path to the location where you want to create and store the wallet.

To set up a password store for the database user accounts, perform the following steps:

1. Create a wallet using the following command:

   ```
   mkstore -wrl <wallet_location> -create
   ```

   After you run the command, a prompt appears. Enter a password for the Oracle Wallet in the prompt.

   > **Note:** The mkstore utility is included in the Oracle Database Client installation.

   The wallet is created with the auto-login feature enabled. This feature enables the database client to access the wallet contents without using the password. For more information, refer to the *Oracle Database Advanced Security Administrator's Guide.*

2. Create the database connection credentials in the wallet using the following command:

   ```
   mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>
   ```

   After you run the command, a prompt appears. Enter the password associated with the database user account in the prompt.

3. Repeat Step 2 for all the database user accounts.

4. Update the sqlnet.ora file to include the following statements:

   ```
   WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA = (DIRECTORY = <wallet_location>)))
   SQLNET.WALLET_OVERRIDE = TRUE
   SSL_CLIENT_AUTHENTICATION = FALSE
   ```

5. Update the tnsnames.ora file to include the following entry for each alias name to be set up.

   ```
   <alias-name> =
       (DESCRIPTION =
        (ADDRESS_LIST =
               (ADDRESS = (PROTOCOL = TCP) (HOST = <host>) (PORT = <port>))
           )
           (CONNECT_DATA =
               (SERVICE_NAME = <service>)
            )
    )
   ```

In the previous example, <alias-name>, <host>, <port>, and <service> are placeholder text for illustration purposes. Ensure that you replace these with the relevant values.

Setting Up Wallets for Database User Accounts

The following examples show how to set up wallets for database user accounts for the following applications:

- For RMS, RWMS, RPM Batch, RETL, RMS, RWMS, and ARI
- For Java Applications (SIM, ReIM, RPM, Alloc, RIB, RSL, AIP, RETL)

For RMS, RPM Plsql Batch, RETL DB, RWMS batch, and ARI

1. Create a new directory called wallet under your folder structure.

```
cd /projects/rms13.2/dev/
mkdir .wallet
```

> **Note:** The default permissions of the wallet allow only the owner to use it, ensuring the connection information is protected. If you want other users to be able to use the connection, you must adjust permissions appropriately to ensure only authorized users have access to the wallet.

2. Create a sqlnet.ora in the wallet directory with the following content.

```
WALLET_LOCATION =   (SOURCE =     (METHOD = FILE)     (METHOD_DATA =
(DIRECTORY =  /projects/rms13.2/dev/.wallet)) )
SQLNET.WALLET_OVERRIDE=TRUE
SSL_CLIENT_AUTHENTICATION=FALSE
```

> **Note**: WALLET_LOCATION must be on line 1 in the file.

3. Setup a tnsnames.ora in the wallet directory. This tnsnames.ora includes the standard tnsnames.ora file. Then, add two custom tns_alias entries that are only for use with the wallet. For example, sqlplus /@dvols29_rms01user.

```
ifile = /u00/oracle/product/11.2.0.4/network/admin/tnsnames.ora

dvols29_rms01user =
  (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = tcp)
  (host = mspdv311.us.oracle.com) (Port = 1521)))
    (CONNECT_DATA = (SID = dvols29) (GLOBAL_NAME = dvols29)))

dvols29_rms01user.world =
  (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = tcp)
  (host = mspdv311.us.oracle.com) (Port = 1521)))
    (CONNECT_DATA = (SID = dvols29) (GLOBAL_NAME = dvols29)))
```

> **Note**: It is important to not just copy the tnsnames.ora file because it can quickly become out of date. The ifile clause (shown above) is key.

4. Create the wallet files. These are empty initially.
   a. Ensure you are in the intended location.
      ```
      $ pwd
      /projects/rms13.2/dev/.wallet
      ```
   b. Create the wallet files.
      ```
      $ mkstore -wrl . –create
      ```
   c. Enter the wallet password you want to use. It is recommended that you use the same password as the UNIX user you are creating the wallet on.
   d. Enter the password again.

      Two wallet files are created from the above command:
      - ewallet.p12
      - cwallet.sso
5. Create the wallet entry that associates the user name and password to the custom tns alias that was setup in the wallet's tnsnames.ora file.

   `mkstore –wrl . –createCredential <tns_alias> <username> <password>`

   Example: `mkstore –wrl . –createCredential dvols29_rms01user rms01user passwd`
6. Test the connectivity. The ORACLE_HOME used with the wallet must be the same version or higher than what the wallet was created with.

   ```
   $ export TNS_ADMIN=/projects/rms13.2/dev/.wallet /* This is very import to use
   wallet to point at the alternate tnsnames.ora created in this example */

   $ sqlplus /@dvols29_rms01user

   SQL*Plus: Release 11

   Connected to:
   Oracle Database 11g

   SQL> show user
   USER is “rms01user”
   ```

   Running batch programs or shell scripts would be similar:

   ```
   Ex: dtesys /@dvols29_rms01user
   script.sh /@dvols29_rms01user

   Set the UP unix variable to help with some compiles :

   export UP=/@dvols29_rms01user
   for use in RMS batch compiles, and RMS, RWMS, and ARI forms compiles.
   ```

   As shown in the example above, users can ensure that passwords remain invisible.

Additional Database Wallet Commands

The following is a list of additional database wallet commands.

- Delete a credential on wallet

  `mkstore –wrl . –deleteCredential dvols29_rms01user`
- Change the password for a credential on wallet

  `mkstore –wrl . –modifyCredential dvols29_rms01user rms01user passwd`

- List the wallet credential entries

  ```
  mkstore –wrl . –list
  ```

  This command returns values such as the following.

  ```
  oracle.security.client.connect_string1
  oracle.security.client.user1
  oracle.security.client.password1
  ```

- View the details of a wallet entry

  ```
  mkstore –wrl . –viewEntry oracle.security.client.connect_string1
  ```

  Returns the value of the entry:

  ```
  dvols29_rms01user
  mkstore –wrl . –viewEntry oracle.security.client.user1
  ```

  Returns value of the entry:

  ```
  rms01user
  ```

  ```
  mkstore –wrl . –viewEntry oracle.security.client.password1
  ```

  Returns value of the entry:

  ```
  passwd
  ```

For Java Applications (SIM, ReIM, RPM, Alloc, RIB, RSL, AIP, RETL)

For Java application, consider the following:

- For database user accounts, ensure that you set up the same alias names between the password stores (database wallet and Java wallet). You can provide the alias name during the installer process.
- Document all aliases that you have set up. During the application installation, you must enter the alias names for the application installer to connect to the database and application server.
- Passwords are not used to update entries in Java wallets. Entries in Java wallets are stored in partitions, or application-level keys. In each retail application that has been installed, the wallet is located in <WEBLOGIC_DOMAIN_HOME>/retail/<appname>/config Example: mspdv351:[103x_WLS] /u00/webadmin/product/10.3.x/WLS/user_projects/domains/132_mck_soa_domain/retail/reim13/config
- Application installers should create the Java wallets for you, but it is good to know how this works for future use and understanding.
- Scripts are located in <WEBLOGIC_DOMAIN_HOME>/retail/<appname>/retail-public-security-api/bin for administering wallet entries.

  Example:

  mspdv351:[103x_WLS] /u00/webadmin/product/10.3.x/WLS/user_projects/domains/132_mck_soa_domain/retail/reim13/retail-public-security-api/bin

- In this directory is a script to help you update each alias entry without having to remember the wallet details. For example, if you set the RPM database alias to rms01user, you will find a script called update-RMS01USER.sh.

  Note: These scripts are available only with application installed by way of an installer.

- Two main scripts are related to this script in the folder for more generic wallet operations: dump_credentials.sh and save_credential.sh.

- If you have not installed the application yet, you can unzip the application zip file and view these scripts in <app>/application/retail-public-security-api/bin.

  Example:

  mspdv351:[103x_WLS] /u00/webadmin/reim/application/retail-public-security-api/bin

update-<ALIAS>.sh

update-<ALIAS>.sh updates the wallet entry for this alias. You can use this script to change the user name and password for this alias. Because the application refers only to the alias, no changes are needed in application properties files.

Usage:

```
update-<username>.sh <myuser>
```

Example:

```
mspdev71:[103x4WLS]
/u00/webadmin/product/10.3.x4/WLS/user_projects/domains/java_domain/retail/rpm
132test/retail-public-security-api/bin> ./update-RMS01USER.sh
usage: update-RMS01USER.sh <username>
<username>: the username to update into this alias.
Example: update-RMS01USER.sh myuser
Note: this script will ask you for the password for the username that you pass
in.
mspdev71:[103x4WLS]
/u00/webadmin/product/10.3.4/WLS/user_projects/domains/java_domain/retail/rpm1
32test/retail-public-security-api/bin>
```

dump_credentials.sh

dump_credentials.sh is used to retrieve information from the wallet. For each entry found in the wallet, the wallet partition, the alias, and the user name are displayed. Note that the password is not displayed. If the value of an entry is uncertain, run save_credential.sh to resave the entry with a known password.

```
dump_credentials.sh <wallet location>
```

Example:

dump_credentials.sh /u00/webadmin/product/10.3.x/WLS/user_projects/domains/132_mck_soa_domain/retail/reim13/config

```
Retail Public Security API Utility
=========================================
```

Below are the credentials found in the wallet at the location:/u00/webadmin/product/10.3.x/WLS/user_projects/domains/132_mck_soa_domain/retail/reim13/config

```
=========================================

Application level key partition name:reim13
User Name Alias:WLS-ALIAS User Name:weblogic
User Name Alias:RETAIL-ALIAS User Name:retail.user
User Name Alias:LDAP-ALIAS User Name:RETAIL.USER
User Name Alias:RMS-ALIAS User Name:rms132mock
User Name Alias:REIMBAT-ALIAS User Name:reimbat
```

save_credential.sh

save_credential.sh is used to update the information in wallet. If you are unsure about the information that is currently in the wallet, use dump_credentials.sh as indicated above. You can add new or update using save_credential.sh as shown below:

```
save_credential.sh -a <alias> -u <user> -p <partition name>  –l <path of the
wallet file location where credentials are stored>
```

Example:

```
mspdv351:[103x_WLS]
/u00/webadmin/mock132_testing/rtil/rtil/application/retail-public-security-
api/bin> save_credential.sh –l
/u00/webadmin/product/10.3.x/WLS/user_projects/domains/132_mck_soa_domain/reta
il/reim13/config
-a RMS-ALIAS -p reim13 -u rms132mock

=========================================
Retail Public Security API Utility
=========================================

Enter password:
Verify password:
```

Note: -p in the above command is for partition name. You must specify the proper partition name used in application code for each Java application.

save_credential.sh and dump_credentials.sh scripts are the same for all applications. If using save_credential.sh to add a wallet entry or to update a wallet entry, bounce the application/managed server so that your changes are visible to the application. Also, save a backup copy of your cwallet.sso file in a location outside of the deployment path, because redeployment or reinstallation of the application will wipe the wallet entries you made after installation of the application. To restore your wallet entries after a redeployment/reinstallation, copy the backed up cwallet.sso file over the cwallet.sso file. Then bounce the application/managed server.

Usage

```
=========================================
Retail Public Security API Utility
=========================================
usage: save_credential.sh -au[plh]
E.g. save_credential.sh -a rms-alias -u rms_user -p rib-rms -l ./
 -a,--userNameAlias <arg>              alias for which the credentials
needs to be stored
 -h,--help                             usage information
 -l,--locationofWalletDir <arg>        location where the wallet file is
created.If not specified, it creates the wallet under secure-credential-wallet
directory which is already present under the retail-public-security-api/
directory.
 -p,--appLevelKeyPartitionName <arg>   application level key partition name
 -u,--userName <arg>                   username to be stored in secure
credential wallet for specified alias*
```

How Does the Wallet Relate to the Application?

The ORACLE Retail Java applications have the wallet alias information you create in an <app-name>.properties file. Below is the reim.properties file. Note the database information and the user are presented as well. The property called datasource.credential.alias=RMS-ALIAS uses the ORACLE wallet with the argument of RMS-ALIAS at the csm.wallet.path and csm.wallet.partition.name = reim13 to retrieve the password for application use.

Reim.properties code sample:

```
datasource.url=jdbc:oracle:thin:@mspdv349.us.oracle.com:1521:pkols07
datasource.schema.owner=rms132mock
datasource.credential.alias=RMS-ALIAS
# ===============================================================
# ossa related Configuration
#
# These settings are for ossa configuration to store credentials.
# ===============================================================

csm.wallet.path=/u00/webadmin/product/10.3.x/WLS/user_projects/domains/132_mck_soa
_domain/retail/reim13/config
csm.wallet.partition.name=reim13
```

How Does the Wallet Relate to Java Batch Program Use?

Some of the ORACLE Retail Java batch applications have an alias to use when running Java batch programs. For example, alias REIMBAT-ALIAS maps through the wallet to REIM app user reimbat, already on the database. To run a ReIM batch program the format would be: reimbatchpgmname REIMBAT-ALIAS <other arguments as needed by the program in question>.

Setting up RETL Wallets

RETL creates a wallet under $RFX_HOME/etc/security, with the following files:

- cwallet.sso
- jazn-data.xml
- jps-config.xml
- README.txt

To set up RETL wallets, perform the following steps:

1. Set the following environment variables:
   - `ORACLE_SID=<retaildb>`
   - `RFX_HOME=/u00/rfx/rfx-13.2.0`
   - `RFX_TMP=/u00/rfx/rfx-13.2.0/tmp`
   - `JAVA_HOME=/usr/jdk1.6.0_12.64bit`
   - `LD_LIBRARY_PATH=$ORACLE_HOME`
   - `PATH=$RFX_HOME/bin:$JAVA_HOME/bin:$PATH`
2. Change directory to `$RFX_HOME/bin`.
3. Run `setup-security-credential.sh`.
   - Enter 1 to add a new database credential.
   - Enter the dbuseralias. For example, `retl_java_rms01user`.
   - Enter the database user name. For example, `rms01user`.
   - Enter the database password.

- Re-enter the database password.
- Enter D to exit the setup script.

4. Update your RETL environment variable script to reflect the names of both the Oracle Networking wallet and the Java wallet.

   For example, to configure RETLforRPAS, modify the following entries in `$MMHOME/RETLforRPAS/rfx/etc/rmse_rpas_config.env`.

   - The RETL_WALLET_ALIAS should point to the Java wallet entry:
     ```
     export RETL_WALLET_ALIAS="retl_java_rms01user"
     ```
   - The ORACLE_WALLET_ALIAS should point to the Oracle network wallet entry:
     ```
     export ORACLE_WALLET_ALIAS="dvols29_rms01user"
     ```
   - The SQLPLUS_LOGON should use the ORACLE_WALLET_ALIAS:
     ```
     export SQLPLUS_LOGON="/@${ORACLE_WALLET_ALIAS}"
     ```

5. To change a password later, run `setup-security-credential.sh`.
   - Enter 2 to update a database credential.
   - Select the credential to update.
   - Enter the database user to update or change.
   - Enter the password of the database user.
   - Re-enter the password.

Quick Guide for Retail Wallets

Retail App	Wallet Type	Wallet Location	Wallet Partition	Alias Name	User Name	Use	Create By	Alias Example	Notes
RMS batch	DB	<RMS batch install dir (MMHOME)>/.wallet	n/a	<Database SID>_<Database schema owner>	<rms schema owner>	Compile, execution	Installer	n/a	Alias hard-coded by installer
RMS forms	DB	<forms install dir>/base/.wallet	n/a	<Database SID>_<Database schema owner>	<rms schema owner>	Compile	Installer	n/a	Alias hard-coded by installer
ARI forms	DB	<forms install dir>/base/.wallet	n/a	<Db_Ari01>	<ari schema owner>	Compile	Manual	ari-alias	
RMWS forms	DB	<forms install dir>/base/.wallet	n/a	<Database SID>_<Database schema owner>	<rwms schema owner>	Compile forms, execute batch	Installer	n/a	Alias hard-coded by installer
RPM app	DB	<RPM batch install dir>/.wallet	n/a	<rms schema owner alias>	<rms schema owner>	Execute batch	Manual	rms-alias	
RWMS auto-login	JAVA	<forms install dir>/base/.javawallet							
			<RWMS Installation name>	<RWMS database user alias>	<RWMS schema owner>	RWMS forms app to avoid dblogin screen	Installer	rwms13inst	

Retail App	Wallet Type	Wallet Location	Wallet Partition	Alias Name	User Name	Use	Create By	Alias Example	Notes
			<RWMS Installation name>	BI_ALIAS	<BI Publisher administrative user>	RWMS forms app to connect to BI Publisher	Installer	n/a	Alias hard-coded by installer
AIP app	JAVA	<weblogic domain home>/retail/<deployed aip app name>/config							Each alias must be unique
			aip13	<AIP weblogic user alias>	<AIP weblogic user name>	App use	Installer	aip-weblogic-alias	
			aip13	<AIP database schema user alias>	<AIP database schema user name>	App use	Installer	aip01user-alias	
			aip13	<rib-aip weblogic user alias>	<rib-aip weblogic user name>	App use	Installer	rib-aip-weblogic-alias	
RPM app	JAVA	<weblogic domain home>/retail/<deployed rpm app name>/config							Each alias must be unique
			rpm13	<rpm weblogic user alias>	<rpm weblogic user name>	App use	Installer	rpm-weblogic-alias	
			rpm13	<rms shema user alias>	<rms shema user name>	App, batch use	Installer	rms01user-alias	

Retail App	Wallet Type	Wallet Location	Wallet Partition	Alias Name	User Name	Use	Create By	Alias Example	Notes
			rpm13	<rpm application user one alias>	<rpm application user one name>	App use	Installer	user1-alias	
			rpm13	<rpm application user two alias>	<rpm application user two name>	App use	Installer	user2-alias	
			rpm13	<rpm batch user alias>	<rpm batch user name>	App, batch use	Installer	rpmbatch-alias	
			rpm13	<rib-rpm weblogic user alias>	<rib-rpm weblogic user name>	App use	Installer	rib-rpm-weblogic-alias	
ReIM app	JAVA	<weblogic domain home>/retail/<deployed reim app name>/config							Each alias must be unique
			<installed app name>	<reim weblogic user alias>	<reim weblogic user name>	App use	Installer	weblogic-alias	
			<installed app name>	<rms shema user alias>	<rms shema user name>	App, batch use	Installer	rms01user-alias	
			<installed app name>	<reim webservice validation user alias>	<reim webservice validation user name>	App use	Installer	reimwebservice-alias	
			<installed app name>	<reim batch user alias>	<reim batch user name>	App, batch use	Installer	reimbatch-alias	

Retail App	Wallet Type	Wallet Location	Wallet Partition	Alias Name	User Name	Use	Create By	Alias Example	Notes
Alloc app	JAVA	<weblogic domain home>/retail/<deployed alloc app name>/config							Each alias must be unique
			<installed app name>	<alloc weblogic user alias>	<alloc weblogic user name>	App use	Installer	weblogic-alias	
			<installed app name>	<rms shema user alias>	<rms shema user name>	App use	Installer	rms01user-alias	
			<installed app name>	<rsl for rms weblogic user alias>	<rsl for rms weblogic user name>	App use	Installer	rsl-rms-weblogic-alias	
RSL app	JAVA	<RSL INSTALL DIR>/rsl-rms/security/config							Each alias must be unique
			rsl-rsm	<rsl weblogic user alias>	<rsl weblogic user name>	App use	Installer	weblogic-alias	
			rsl-rsm	<rms shema user alias>	<rms shema user name>	App use	Installer	rms01user-alias	
SIM app	JAVA	<weblogic domain home>/retail/<deployed sim app name>/config							
			rpm	<rpm weblogic user alias>	<rpm weblogic user name>	App use	Installer	rpm-weblogic-alias	

Retail App	Wallet Type	Wallet Location	Wallet Partition	Alias Name	User Name	Use	Create By	Alias Example	Notes
			rms	<rsl for rms weblogic user alias>	<rsl for rms weblogic user name>	App use	Installer	rsl-rms-weblogic-alias	
			rib-sim	<rib-sim weblogic user alias>	<rib-sim weblogic user name>	App use	Installer	rib-sim-weblogic-alias	
RETL	JAVA	<RETL home>/etc/security	n/a	<target application user alias>	<target application db userid>	App use	Manual	retl_java_rms01user	User may vary depending on RETL flow's target application
RETL	DB	<RETL home>/.wallet	n/a	<target application user alias>	<target application db userid>	App use	Manual	<db>_<user>	User may vary depending on RETL flow's target application
RIB	JAVA	<RIBHOME DIR>/deployment-home/conf/security							<app> is one of aip, rfm, rms, rpm, sim, rwms, tafr
JMS			jms<1-5>	<jms user alias> for jms<1-5>	<jms user name> for jms<1-5>	Integration use	Installer	jms-alias	
WebLogic			rib-<app>-app-server-instance	<rib-app weblogic user alias>	<rib-app weblogic user name>	Integration use	Installer	weblogic-alias	
Admin GUI			rib-<app>#web-app-user-alias	<rib-app admin gui user alias>	<rib-app admin gui user name>	Integration use	Installer	admin-gui-alias	

Retail App	Wallet Type	Wallet Location	Wallet Partition	Alias Name	User Name	Use	Create By	Alias Example	Notes
Application			rib-<app>#user-alias	<app weblogic user alias>	<app weblogic user name>	Integration use	Installer	app-user-alias	Valid only for aip, rpm, sim
DB			rib-<app>#app-db-user-alias	<rib-app database schema user alias>	<rib-app database schema user name>	Integration use	Installer	db-user-alias	Valid only for rfm, rms, rwms, tafr
Error Hospital			rib-<app>#hosp-user-alias	<rib-app error hospital database schema user alias>	<rib-app error hospital database schema user name>	Integration use	Installer	hosp-user-alias	

N

Appendix: Installation Order

This section provides a guideline as to the order in which the Oracle Retail applications should be installed. If a retailer has chosen to use only some of the applications, the order is still valid, less the applications not being installed.

> **Note:** The installation order is not meant to imply integration between products.

Enterprise Installation Order

1. Oracle Retail Merchandising System (RMS), Oracle Retail Trade Management (RTM), Oracle Retail Sales Audit (ReSA). Optional: Oracle Retail Fiscal Management (ORFM)

   > **Note:** ORFM is an optional application for RMS if you are implementing Brazil localization.

2. Oracle Retail Service Layer (RSL)
3. Oracle Retail Extract, Transform, Load (RETL)
4. Oracle Retail Active Retail Intelligence (ARI)
5. Oracle Retail Warehouse Management System (RWMS)
6. Oracle Retail Invoice Matching (ReIM)
7. Oracle Retail Price Management (RPM)

   > **Note:** During installation of RPM, you are asked for the RIBforRPM provider URL. Because RIB is installed after RPM, make a note of the URL you enter. To change the RIBforRPM provider URL after you install RIB, edit the remote_service_locator_info_ribserver.xml file.

8. Oracle Retail Allocation
9. Oracle Retail Central Office (ORCO)
10. Oracle Retail Returns Management (ORRM)
11. Oracle Retail Back Office (ORBO) or Back Office with Labels and Tags (ORLAT)
12. Oracle Retail Store Inventory Management (SIM)

    > **Note:** During installation of SIM, you are asked for the RIB provider URL. Because RIB is installed after SIM, make a note of the URL you enter. To change the RIB provider URL after you install RIB, edit the remote_service_locator_info_ribserver.xml file.

13. Oracle Retail Predictive Application Server (RPAS)
14. Oracle Retail Demand Forecasting (RDF)
15. Oracle Retail Category Management (CM)
16. Oracle Retail Replenishment Optimization (RO)
17. Oracle Retail Analytic Parameter Calculator Replenishment Optimization (APC RO)

18. Oracle Retail Regular Price Optimization (RPO)
19. Oracle Retail Merchandise Financial Planning (MFP)
20. Oracle Retail Size Profile Optimization (SPO)
21. Oracle Retail Assortment Planning (AP)
22. Oracle Retail Item Planning (IP)
23. Oracle Retail Item Planning Configured for COE (IP COE)
24. Oracle Retail Advanced Inventory Planning (AIP)
25. Oracle Retail Integration Bus (RIB)
26. Oracle Retail Point-of-Service (ORPOS)
27. Oracle Retail Markdown Optimization (MDO)
28. Oracle Retail Clearance Optimization Engine (COE)
29. Oracle Retail Analytic Parameter Calculator for Markdown Optimization (APC-MDO)
30. Oracle Retail Analytic Parameter Calculator for Regular Price Optimization (APC-RPO)
31. Oracle Retail Promotion Intelligence and Promotion Planning and Optimization (PI-PPO)
32. Oracle Retail Analytics
33. Oracle Retail Workspace (ORW)