Oracle® Retail Store Inventory Management

Installation Guide
Release 12.0.10IN
E22537-02

March 2011

ORACLE

Oracle® Store Inventory Management Installation Guide, Release 12.0.10IN

Primary Author: Donna Linde

Contributors: Anshuman Accanoor

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 Store Inventory Management, Installation Guide, Release 12.0.10IN

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 Online Documentation available on the Oracle Technology Network Web site. 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

For more information, see the following documents in the Oracle Retail Store Inventory Management Release 12.0.10IN documentation set:

- *Oracle Retail Store Inventory Management Release Notes*
- *Oracle Retail Store Inventory Management Operations Guide*
- *Oracle Retail Store Inventory Management Data Model*
- *Oracle Retail Store Inventory Management User Guide Addendum*
- *Oracle Retail Store Inventory Management Online Help*

Customer Support

https://metalink.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

If you are installing the application for the first time, you install either a base release (for example, 12.0) or a later patch release (for example, 12.0.9). If you are installing a software version other than the base release, be sure to read the documentation for each patch release (since the base release) before you begin installation. Patch documentation can contain critical information related to the base release and code changes that have been made since the base release.

Oracle Retail Documentation on the Oracle Technology Network

In addition to being packaged with each product release (on the base or patch level), all Oracle Retail documentation is available on the following Web site (with the exception of the Data Model which is only available with the release packaged code):

http://www.oracle.com/technology/documentation/oracle_retail.html

Documentation should be available on this Web site within a month after a product release. Note that documentation is always available with the packaged code on the release date.

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."

> **Note:** This is a note. It is used to call out information that is important, but not necessarily part of the procedure.

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

A hyperlink appears like this.

1

Preinstallation Tasks

Check Database Server Requirements

General Requirements for a database server running SIM include:

Supported on 10gR2	Versions Supported:
Database Server OS	OS certified with Oracle Database 10gR2 Enterprise Edition. Options are: ▪ Sun Sparc 5.9 and 5.10 ▪ AIX 5.2 and 5.3 ▪ HP-UX 11.23 (PARISC) ▪ OEL 4 Update 4 (OEL 4.4) for x86-64
Database Server	Oracle Database 10g Release 2 Enterprise Edition (10.2.0.4 patchset required) with the following components: ▪ Oracle Database 10g ▪ Oracle Partitioning ▪ Oracle Net Services ▪ Oracle Call Interface (OCI) ▪ Oracle Programmer ▪ Oracle XML Development Kit ▪ Companion CD ▪ ANSI compliant C compiler (certified with OS and database version) **Patches:** ▪ 10.2.0.4 patchset: 6810189 **Others components:** ▪ Perl compiler 5.0 or later ▪ x-Windows interface

Check Application Server Requirements

General requirements for an application server capable of running SIM include:

– Supported on:	– Versions Supported:
Application Server OS	OS certified with Oracle Application Server 10g 10.1.3.4. ▪ Sun Sparc 5.9 and 5.10 ▪ AIX 5.2 and 5.3 ▪ HP-UX 11.23 (PARISC) ▪ OEL 4 Update 4 (OEL 4.4) for x86-64
Application Server	Oracle Application Server 10g 10.1.3.4 with the following patches: ▪ 5632264 (NEED UPDATED TIMEZONE FILES (VERSION 4) FOR MORE DST RULE CHANGES)

Note: This release of SIM is only supported in a managed OC4J instance as part of OracleAS 10g. It is not supported on OC4J standalone

Check Directory Server Requirements

SIM uses directory server based user authentication and searching. For LDAP, SIM is certified with the following directory servers:

- Oracle Internet Directory 10.1.2.0.2*
- OpenLDAP version 2.x**

There are no known limitations that would prevent SIM from running against any LDAP 3.0-compliant directory server.

Check Third-Party Software Dependencies

- Oracle Retail Wireless Foundation Server – provided by Wavelink 4.x

Check Client PC and Web Browser Requirements

– Requirement	– Version
Operating system	Windows 2000 or XP
Display resolution	1024x768
Processor	1GHz or higher.
Memory	minimum of 512MBytes
Sun JRE	5.0 Update 11 or newer (1.5.0_11)
Microsoft Internet Explorer	version 5.5 or higher The browser is used to launch the Java WebStart client.

Oracle Retail Dependencies

The following Oracle Retail products can be integrated with SIM. Next to each product is an indication of whether it is required or optional for SIM to function properly:

- Retail Integration Bus (RIB) 12.0.9 and all subsequent patches and hot fixes – Required
  - RIBforSIM is a separately-packaged component that connects SIM to the RIB. Although typically used to integrate SIM with RMS, RIB can also be used to integrate SIM with other merchandising systems.

    Note: RIB requires custom modifications to use a merchandising system other than RMS

- Retail Merchandising System (RMS) 12.0.10 India Localized – Optional
- Retail Service Layer for RMS (RSLforRMS) 12.0.10 – Optional
- Retail Price Management (RPM) 12.0.10 India Localized – Optional

The above products can be installed before or after SIM. However, it is helpful to know the connection details for the other products ahead of time so that you can provide them to the SIM application installer, which will configure the connection points for you.

2

Database Installation Tasks

Before you apply the SIM 12.0.10 India Localized patch:

- Make a backup of all your objects and database schema.
- Check that SIM 12.0.9 is installed.
- Review the enclosed SIM 12.0.10 Patch Release Notes (sim-12010-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 module (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.

> **Note:** These instructions refer to SIM12DEV as the Oracle owning schema.

Copy from CD Directory

1. Copy the sim-db-patch.zip file from the CD /dbserverunix directory to a newly created staging directory on your UNIX server.
2. Unzip the file by entering:
   ```
   unzip sim-db-patch.zip
   ```
3. Export the NLS_LANG variable with an appropriate language and locale, and the UTF8 encoding. For example:
   ```
   export NLS_LANG=AMERICAN_AMERICA.UTF8;
   ```

Apply SIM Patch Files

1. Change directories to STAGING_AREA/sim/dbschema/patch.
2. Log in to SQL*Plus as SIM12DEV and run the following commands:
3. Change directories to STAGING_AREA/sim/dbschema/patch/data.

> **Note:** This documentation correction does not affect the installation of this bundled hot fix release; see "Installing the Bundled Hot Fix Release" for installation instructions for SIM 12.0.11.4.

4. Log in to SQL*Plus as SIM12DEV and run the following commands:
   ```
   SQL> @0073_sim_translation_keys_delete.sql
   SQL> @0074_sim_translation_keys_other.sql
   SQL> @0075_sim_translation_lang_en.sql
   ```

Update SIM Stored Procedures

1. Change directories to STAGING_AREA/sim/dbschema/patch/stored_procedures.
2. Log into SQL*Plus as SIM12DEV and run the following commands:
   ```
   SQL> @stockCountss.pls
   SQL> @stockCountsb.pls
   ```

3

Application Installation

Before proceeding you must install Oracle Application Server 10g 10.1.3 plus the patches listed in the Chapter 1 of this document. The SIM application will be deployed to an OC4J instance within the OracleAS 10g installation. It is assumed Oracle Database has already been configured and loaded with the appropriate SIM schema for your installation.

Create a New OC4J Instance for SIM

You can skip this section if you are redeploying to an existing OC4J instance.

The SIM application must be deployed to its own dedicated OC4J instance. For instructions on how to create a new OC4J instance, see *Adding and Deleting OC4J Instances* in the *Reconfiguring Application Server Instances* chapter of the *Oracle Application Server Administrator's Guide*.

1. Log into the server which is running your OracleAS 10g installation. Set your ORACLE_HOME environment variable to point to this installation.
2. Choose a name for the new OC4J instance.

   Example: sim-oc4j-instance

   Create this OC4J instance as documented in the *Oracle Application Server Administrator's Guide*.

   Example:
   $ORACLE_HOME/bin/createinstance
   –instanceName sim-oc4j-instance

   When prompted for the oc4jadmin password, provide the same administrative password you gave for the Oracle Application Server installation. All OC4J instances running Oracle Retail applications must have the same oc4jadmin password.
3. **(Linux only)** Increase memory for the new OC4J instance by modifying $ORACLE_HOME/opmn/conf/opmn.xml. Locate the OC4J instance you just created, and add the **-XX:PermSize=256m -XX:MaxPermSize=512m -Xms256m -Xmx256m** options to the start-parameters section.

   Example:

   ```
   <process-type id=”orco-inst” module-id=”OC4J”
   status=”enabled”>
      <module-data>
         <category id=”start-parameters”>
            <data id=”java-options” value=”-server
   -XX:PermSize=256m -XX:MaxPermSize=512m -Xms256m -
   Xmx256m -
   Djava.security.policy=$ORACLE_HOME/j2ee/orco-
   inst/config/java2.policy –Djava.awt.headless=true
   –Dhttp.webdir.enabled=false”/>
         </category>
   ```
4. Force OPMN to reload the configuration file.

Example: $ORACLE_HOME/opmn/bin/opmnctl reload

5. Start the OC4J instance. You can do this through the Enterprise Manager web interface, or on the command line using the opmnctl utility:

   Example: $ORACLE_HOME/opmn/bin/opmnctl startproc process-type=sim-oc4j-instance

6. Verify that the OC4J instance was fully started. If you are using the Enterprise Manager web interface, the instance should have a green arrow indicating that it is running. On the command line, verify that the instance has a status of "Alive".

   Example: $ORACLE_HOME/opmn/bin/opmnctl status

   If you are unable to start the OC4J instance after several attempts, try increasing the startup timeouts in ORACLE_HOME/opmn/conf/opmn.xml. If that does not help, consult the Oracle Application Server documentation for further assistance.

Configure Apache for JNLP Files

If this is the first WebStart application that is being installed in the HTTP server, you need to configure the **mime.types** file with the jnlp file type. If you are using the Apache distribution that is included with OracleAS, this file can be found under ORACLE_HOME/Apache/Apache/conf. Add the following line to the file:

```
application/x-java-jnlp-file          jnlp
```

Restart the Apache server for this change to take effect. If you do not add this line then jnlp files are served as plain text and you are not able to launch the application.

Example: $ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server

Set the LANG Environment Variable

The LANG environment variable must be set in the profile of the UNIX user who owns the application server ORACLE_HOME files. If you change the value of LANG or set the value for the first time, you must restart the Application Server in order for the change to take effect.

Example:

LANG=en_US

export LANG

For instructions on how to restart the Application Server, see the *opmnctl Commands* chapter of the *Oracle® Process Manager and Notification Server Administrator's Guide*.

Example:

$ORACLE_HOME/opmn/bin/opmnctl stopall

$ORACLE_HOME/opmn/bin/opmnctl startall

Expand the SIM Application Distribution

1. Log into the UNIX server as the user who owns the OracleAS 10g installation. Create a new staging directory for the SIM application distribution (sim12application.zip). There should be a minimum of 300 MB disk space available for the application installation files.

   Example: $ORACLE_HOME/j2ee/sim-oc4j-instance/sim-staging

   This location is referred to as INSTALL_DIR for the remainder of this chapter.
2. Copy sim12application.zip to <INSTALL_DIR> and extract its contents.

Provide Third-Party Jar File

The SIM application requires the hibernate2.jar file to be installed. This file should be downloaded from http://www.hibernate.org and placed in the <INSTALL_DIR>/sim/application/hibernate folder before the installer is launched. For SIM 12, Hibernate 2.1.8 should be used. You need to download the Hibernate distribution and extract the hibernate2.jar file from it.

The SIM application installer verifies that hibernate2.jar has been provided and that it is the correct version. If hibernate2.jar is missing or incorrect, the installer will not proceed.

The installer applies hibernate2.jar to the SIM application by placing it under the ORACLE_HOME/j2ee/<oc4j-instance-name>/sim-home/library/ent directory.

Run the SIM Application Installer

Once you have an OC4J instance that is started, you can run the SIM application installer. This installer configures and deploys the SIM application and Java WebStart client files.

Note: Appendix C contains details on every screen and field in the application installer.

1. Expand the sim12application.zip distribution into <INSTALL_DIR>.
2. Set the ORACLE_HOME and JAVA_HOME environment variables. ORACLE_HOME should point to your OracleAS installation. JAVA_HOME should point to $ORACLE_HOME/jdk.
3. If you are using an X server such as Exceed, set the DISPLAY environment variable so that you can run the installer in GUI mode (recommended). If you are not using an X server, or the GUI is too slow over your network, unset DISPLAY for text mode.
4. Verify that the required third-party jar file is in place: <INSTALL_DIR>/sim/application/hibernate/hibernate2.jar
5. Run the install.sh script. This launches the installer. After installation is completed, a detailed installation log file is created: <INSTALL_DIR>/ sim/application/logs/sim-install-app.<timestamp>.log.
6. Sign the sim-config.jar file. (See instructions below)
7. Copy the sim-home directory if you wish to run batch scripts from a location outside of the ORACLE_HOME. This step is optional. (See instructions below)

Sign the SIM Client Configuration Jar File

There is some client-side configuration that the installer performs which results in a modified sim-config.jar file after installation. Because of this, the jar file cannot be pre-signed by Oracle. The user must sign this jar file after the installer has completed.

To create an example key called "foo", the following command can be run:

$JAVA_HOME/bin/keytool –genkey –alias foo

This command prompts you for a keystore password along with organizational info.

Once complete, the keystore alias resides in the default location in the user's home directory (ie ~/.keystore). If you get an error message saying that the keystore has been tampered with, try renaming or deleting the ~/.keystore file and running the keytool command again.

The sim-config.jar is located within the deployed client application: $ORACLE_HOME/j2ee/<oc4j-instance-name>/applications/sim-client/sim-client/lib/sim-config.jar

To sign the sim-config.jar file using your alias and keystore, run the jarsigner utility.

Example: $JAVA_HOME/bin/jarsigner sim-config.jar foo

Consult the "jarsigner" documentation from Sun for further information on the JAR signing process.

SIM Batch Scripts

The SIM application installer places the SIM batch programs with the rest of the SIM application files under $ORACLE_HOME/j2ee/<oc4j-instance-name>/sim-home.

The batch programs can be run from a different location if you cannot run them from under the application server ORACLE_HOME. To install the batch files in a different location just copy the entire $ORACLE_HOME/j2ee/<oc4j-instance-name>/sim-home directory to the appropriate destination.

The sim-home is assumed to be located on the same server as the application server. If you copy the sim-home to a location on a different server, then you need to configure the file path to the sim-batch.log file, which is defined in sim-home/batch-config/log4j.xml.

Resolving Errors Encountered During Application Installation

If the application installer encounters any errors, it halts execution immediately. You can run the installer in silent mode so that you don't have to retype the settings for your environment. See Appendix D of this document for instructions on silent mode.

See Appendix F of this document for a list of common installation errors.

Since the application installation is a full reinstall every time, any previous partial installs will be overwritten by the successful installation.

Manual Deployment Option

Skip this section if you chose the default option of allowing the installer to complete installation to the application server.

The installer includes the option to configure the application locally and skip deployment to the application server. If this option is chosen, the installer will make the configured application files available under <INSTALL_DIR>/sim/application/sim12/configured-output/.

If you chose this installer option, you can complete the installation by following these steps:

1. Inspect and then overlay files from <INSTALL_DIR>/sim/application/sim12/configured-output into your application server installation.
2. Deploy the SIM EAR file using the Enterprise Manager web interface. The configured EAR file is located at <INSTALL_DIR>/sim/application/sim12/configured-output/sim.ear. When deploying the EAR file, you should provide the same application name you gave to the installer. This value was stored in the <INSTALL_DIR>/sim/application/ant.install.properties file by the installer for later reference.
3. Deploy the client WAR file to the application server using the Enterprise Manager web interface. The configured WAR file is located at <INSTALL_DIR>/sim/application/sim12/configured-output/sim-client.war.
4. Start the Wavelink server. The start file for Wavelink is located at: $ORACLE_HOME/j2ee/<oc4j-instance-name>/sim-home/wavelink/bin/wavelink-startup.sh

Backups Created by Installer

The SIM application installer will back up the sim-home directory and data_sources.xml file if it finds a previous installation of SIM. The backups are made by suffixing the file or directory with a timestamp. This is done to prevent the removal of any custom changes you might have. These backup directories can be safely removed without affecting the current installation.

Example: sim-home-backup-200708171550

Example: data-sources.xml.200708171550

Test the SIM Application

After the application installer completes and you have signed the sim-config.jar you should have a working SIM application installation. To launch the application client, open a web browser and go to the client URL. You can find the URL in the *next-steps* section of the log file that was produced by the installer.

Example: http://myhost:7777/sim-client/launch?template=sim_jnlp_template.vm

If after you log in you receive an error message that the timezone is not properly configured for your store, please refer to Configuring SIM Across Time Zones in the *SIM Operations Guide*.

Help Files

The application installer automatically copies the help files to the proper location. They are accessible from the help links within the application.

Starting and Stopping SIM

The startup and shutdown scripts for SIM can be found with the SIM batch scripts in:

ORACLE_HOME/j2ee/<oc4j-instance-name>/sim-home/bin/startup.sh

ORACLE_HOME/j2ee/<oc4j-instance-name>/sim-home/bin/shutdown.sh

SIM can also be restarted by using the Enterprise Manager to restart the OC4J instance that contains SIM. However, if you use the Enterprise Manager to restart SIM, the Wavelink server needs to be restarted manually.

Starting and Stopping the Wavelink Server

In order to use handheld wireless devices with SIM, the Wavelink server must be running. The SIM application installer installs, configures, and starts the Wavelink server for you, so once the SIM application install is complete, the Wavelink server is ready to be used.

> **Note:** If you use the Enterprise Manager to restart SIM, then you will need to restart the Wavelink server manually.

If you use SIM's startup and shutdown scripts to restart SIM on the command line, then the Wavelink server must also be restarted along with SIM. However, if you use the Enterprise Manager to restart SIM, the Wavelink server is not affected. Thus it needs to be restarted manually once SIM is running again.

The Wavelink server scripts can be found here:

ORACLE_HOME/ j2ee/<oc4j-instance-name>/sim-home/wavelink/bin/wavelink-startup.sh

ORACLE_HOME/ j2ee/<oc4j-instance-name>/sim-home/wavelink/bin/wavelink-shutdown.sh

> **Note:** Contact Wavelink for requirements, recommendations, and evaluations of currently deployed equipment for operating SIM on wireless handheld devices and printers. The SIM -- Wireless Foundation™ has a component that runs on the handheld device and a corresponding component that runs on the application server. Wireless component installation and configuration is not covered in this installation guide.

4

LDAP Configuration

Getting SIM to Work With OpenLDAP

Setup of LDAP

There are a number of commercial LDAP servers available on the market – a commonly used one is OpenLDAP (available at www.openldap.org).

> **Note:** Development of LDAP functionality in the SIM product was carried out by using OpenLDAP 2.1.12 server with a Berkeley DB 4.1.25 back-end on UNIX

Once an LDAP server has been selected and installed, the SIM data schema (SIM.schema) must be loaded on top of the default LDAP core schema (core.schema) supplied by the server. The following sample configuration files and scripts are included in this release at SIM_INSTALL_DIR/server<Platform>/retek/sim/files/prod/database/ldap for use with OpenLDAP and Berkeley DB installations:

> **Note:** The following scripts and configuration files are provided as examples only. Variations may be necessary based on the LDAP server that is chosen and installed.

- slapd.conf

  An example OpenLDAP server configuration file.

- start_ldap.sh (start_ldap.bat)

  An example Start up script that starts just the LDAP stand-alone server. <LDAPServerName> and <LDAPServerPort> will have to be set to fit your environment.

- loadnStart.sh (loadnStart.bat)

An example script that removes any LDAP databases, recreates the data directory, starts the LDAP stand-alone server, and loads a sample LDIF file. The sample LDIF files are discussed in the next section. This script needs to be modified to conform to your environment's directory structure and LDAP server.

> **Note:** Running this script completely deletes any data in the target LDAP repository and insert the test data contained in the ldif file the script references.

- stop_ldap.sh

  Stops the LDAP standalone server by killing the process.

> **Note:** A stop script for Windows was not created – simply use ctl-c to stop the server process in the CMD window in which it is running

- sim.schema

  Contains the SIM LDAP schema that is loaded over the core.schema provided with the LDAP server.
- Several .ldif files that contain sample data are also included. They are explained further in the next section.

To Configure SIM in an OpenLDAP Environment LDAP Server:

1. Customize the stop_ldap.sh script:
   - Change the argument of the kill command to the location of the slapd.pid file as specified by the 'pidfile' key in the LDAP configuration file (slapd.conf).
2. Customize the start_ldap.sh and the loadnStart.ldap scripts:
   - Change the LD_LIBRARY_PATH key to the location of the BerkeleyDB libraries (this depends on the distribution of OpenLDAP used, some are static-linked against the BerkeleyDB libraries and do not need this).
   - In loadnStart.sh, change the lines that delete and recreate the LDAP database to reference your specific setup.
   - Change line that launches the LDAP server. The format of the line is given in the script.
   - In loadnStart.sh, change the line that loads the sample data into the LDAP server. The format of the line is given in the script.
3. Customize the slapd.conf file:
   - Ensure that the SM.schema file is referenced correctly near the top of the slpad.conf file.
4. Execute stop_ldap.sh (if the LDAP server is already running)
5. Execute start_ldap.sh or loadnStart.sh to start the stand-alone LDAP server.

   If loadnStart.sh is run, a scrolling list of data inserts into the LDAP repository should be displayed. If you get an error starting the server - check to make sure the server was not started anyway.

   For connection errors, double check that the rootdn name and password specified in slapd.conf and start_ldap.sh/loadnStart.sh match.

SIM Data Schema Loads

A Lightweight Directory Access Protocol (LDAP) Server handles user authentication in SIM. In order to have SIM setup correctly and have users login to the application, SIM needs to communicate with a LDAP server. Once the LDAP server is configured and installed, the SIM data schema must be loaded on top of the default LDAP core schema (core.schema) supplied by the server. This is done by the example script loadnStart.sh above.

Loading the data consists of creating three primary objects that SIM uses:

- Users
- Roles
- Stores

Several sample data entry files are available in the SIM_INSTALL_DIR/server<Platform>/retek/sim/files/prod/database/ldap directory and illustrate the formatting of the required data. The file sampleData.ldif contains a sample entry for the SIM schema. The other sample data files, testData.ldif and superLDIF.ldif contain varying amounts of sample data.

Note: You can have more than one rsimStoreID by simply repeating the userStore line, but should only have one homeStore.

Note: Any user store entry for the user object must have a corresponding Store data populated in the SIM Oracle database to allow a successful login (table PA_STR_RTL).

Note: SIM does not currently use/validate against the employmentStatus field, but may at some future release. Valid types are below.

0 = active
1 = terminated
2 = onleave
3 = oncall

User roles contain various privileges that users assigned this rsimRoleName can access. If a role is set to TRUE in isStoreSuperUser, that role can perform all privileges in any store they are assigned to as long as that task is available in that store. If a user has TRUE in isSuperUser, they can perform any task in any store as long as that task is available in that store.

The privileges available in SIM are listed below.

- Create/View Stock Count (my store) - 1
- Create/View Stock Count (all stores) - 2
- Authorize Count - 4
- Item Lookup - 8
- Transfer Receive - 16
- Transfer Create/Save - 32
- Supplier Lookup - 64
- DSD - 128
- Return Stock - 256
- Warehouse Delivery - 512
- Container Lookup - 1024
- Inventory Adjustment - 2048
- Pricing - 4096
- View/Perform Stock Count - 8192
- Store Admin/Configuration – 16384
- Transfer Requests – 32768
- Item Requests – 65536
- Sequencing - 131072

LDAP store data must match the location data maintained in the SIM database. Stores also have privileges assigned to them. It needs to be noted that store privileges override user privileges. For example, if privilege 512 is not available to a store, users logged into that store are not able to perform function 512 even if their role specifically allows it.

Once an LDAP user is correctly set up for a store that is present in your SIM database you are able to log in to the SIM client.

A

Appendix: SIM Configuration Files

This section documents which files are configured by the installers and where you can find them to do manual configuration later.

OC4J Instance Name in startup.sh and shutdown.sh

Example: opmnctl startproc process-type=<oc4j-instance-name>

Example: opmnctl stopproc process-type=<oc4j-instance-name>

Client Codebase and Provider URL in JnlpLaunch.properties

- The token.sim_provider_url property contains the JNDI provider URL. The URL should have the following format:

  token.sim_provider_url=opmn:ormi://<host>:<opmn-req-port>:<oc4j-instance-name>/<sim-app-name>

- The token.sim_download_url property contains the client codebase. The client codebase should have the following format:

  token. sim_download_url=http://<host>:<http-port>/<client-context-root>

Client Codebase in sim_config.jnlp

The client codebase specified in the sim_config.jnlp file should have the following format:

codebase="http://<host>:<http-port>/<client-context-root>"

Client Codebase in client_master.cfg

The client codebase is used to form the WebHelp URL in the client_master.cfg file.

Example: HELP_BASE_DIR= http://<host>:<http-port>/<client-context-root>/WebHelp

JNDI Details in jndi.cfg

The JNDI properties should have the following format:

NAMING_SERVER_URL=opmn:ormi://<host>:>:<opmn-req-port>:<oc4j-instance-name>/<sim-app-name>

SECURITY_PRINCIPAL=oc4jadmin

SECURITY_CREDENTIALS=<oc4jadmin-password>

JNDI Provider URLs for Other Oracle Retail Applications in jndi_providers.xml

If SIM is integrated with other Oracle Retail applications such as RPM, RMS, or RIB, then the JNDI providers for those applications must be provided in the jndi_providers.xml file. The format of each URL should be:

Example: url=" opmn:ormi://<host>:<opmn-req-port>:<rpm-oc4j-instance-name>/<rpm-app-name>"

Example: url=" opmn:ormi://<host>:<opmn-req-port>:<rms-oc4j-instance-name>/<rms-app-name>"

Example: url=" opmn:ormi://<host>:<opmn-req-port>:<ribforsim-oc4j-instance-name>/<ribforsim-app-name>"

Context Roots for Web Modules in application.xml

The context roots for SIM's WAR file and Web Services WAR file are located in the application.xml inside SIM's EAR file.

```
<application>
    <module>
        <web>
            <web-uri>sim.war</web-uri>
            <context-root>/simweb</context-root>
        </web>
    </module>
    <module>
        <web>
            <web-uri>sim-ws.war</web-uri>
            <context-root>/sim-ws</context-root>
        </web>
    </module>
</application>
```

Database Information in data-sources.xml

The <connection-pool> and <managed-data-source> elements define the data sources:

```
<connection-pool name="SIM Connection Pool">
    <connection-factory factory-class="oracle.jdbc.pool.OracleDataSource"
          user="sim-schema-user" password="sim-schema-password"
          url="jdbc:oracle:thin:@host:port:sid"/>
</connection-pool>
<managed-data-source login-timeout="30"
          connection-pool-name="SIM Connection Pool"
          jndi-name="jdbc/SimDataSource" name="jdbc/SimDataSource"/>
```

Database Information in batch_db.cfg

SIM's batch scripts use the properties in the batch_db.cfg file to connect to the database. The database properties should have the following format:

URL=jdbc:oracle:thin:@<host>:<port>:<sid>

USER_NAME=<sim-schema-user>

PASSWORD=<sim-schema-password>

RIB and RSL Configuration in integration.cfg

RIB message publishing and RSL calls can be enabled or disabled by setting the `ribMessagePublishEnabled` and `rslCallsEnabled` properties respectively.

Examples:

ribMessagePublishEnabled=true

rslCallsEnabled=true

LDAP Details in ldap.cfg

The LDAP settings are found in the ldap.cfg file. They should have the following format:

PRIMARY_LDAP_URL=ldap://<ldap-host>:<ldap-port>

BASE_DN=<ldap-search-base-dn>

APPLICATION_LOGIN=<ldap-search-user-dn>

APPLICATION_PASSWORD=<ldap-search-user-password>

SIM Log File in sim-home/files/prod/config/log4j.xml

The location of SIM's log file is defined in the log4j.xml. Example:

```
<param name="File"
       value="<ORACLE_HOME>/j2ee/<oc4j-instance-name>/sim-home/log/sim.log"/>
```

SIM Batch Script Log File in sim-home/batch-config/log4j.xml

The location of the log file used by SIM batch scripts is defined in the log4j.xml found under the sim-home/batch-config directory. Example:

```
<param name="File"
       value="<ORACLE_HOME>/j2ee/<oc4j-instance-name>/sim-home/log/sim-
batch.log"/>
```

Reporting Tool Details in reporting.cfg

SIM uses the settings in reporting.cfg to access the reporting tool server.

Examples:

REPORTING_TOOL_ADDRESS=http://<report-host>:<report-port>/<bipublisher-instance>/servlet/report

REPORTING_TOOL_REQUEST_URL=http://<report-host>:<report-http-port>/bipublisher_10.1.3.2/servlet/scheduler

REPORTING_TOOL_REQUEST_USERNAME=admin

REPORTING_TOOL_REQUEST_PASSWORD=<password>

Wireless Server Port in wavelink-startup.sh and wireless_services.cfg

The wireless server port is located in both the wavelink-startup.sh and the wireless_services.cfg.

Example: `wireless_port=40002`

Example: `PORT=40002`

DexNex Directories in sim_batch.cfg

The Dexnex file parser imports direct delivery (DSD) information from an EDI flat file produced by a supplier. It uses an input directory to place files for processing and an error directory to place files that fail.

Example:

DEXNEX_INPUT_DIR=<ORACLE_HOME>/j2ee/<ocj-instance-name>/sim-home/files/prod/dexnex/dexnex-input

DEXNEX_ERRORS_DIR=<ORACLE_HOME>/j2ee/<ocj-instance-name>/sim-home/files/prod/dexnex/dexnex-error

Alternate Example:

DEXNEX_INPUT_DIR=<path-to-alternate-sim-home>/files/prod/dexnex/dexnex-input

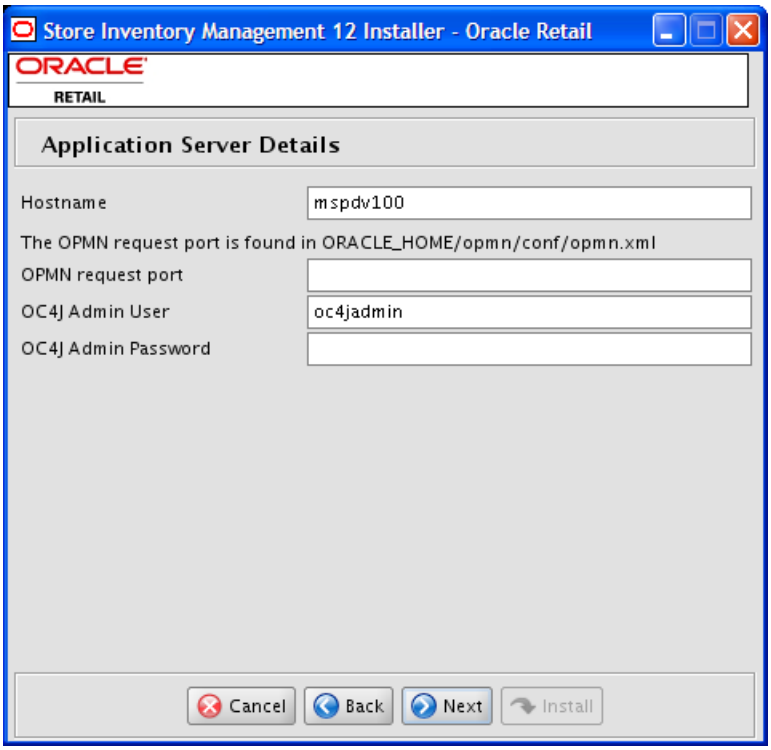
DEXNEX_ERRORS_DIR=<path-to-alternate-sim-home>/files/prod/dexnex/dexnex-error

B

Appendix: SIM Application Installer Screens

You will need the following details about your environment for the installer to successfully deploy the SIM application. Depending on the options you select, you may not see some screens.

Screen: Application Server Details



Fields on this screen:

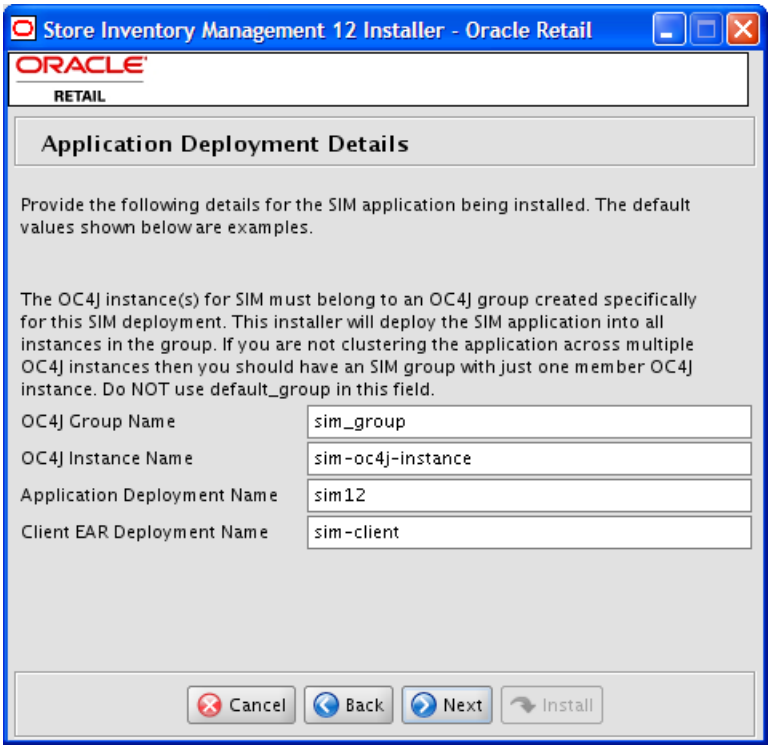
–	**Field Title**	Hostname
–	**Field Description**	The hostname of the server where the application server is installed
–	**Destination**	client_master.cfg
–	**Example**	myhost.us.oracle.com
–	**Notes**	Used by installer scripts to deploy EAR and WAR files and to create default inputs for client codebase and JNDI provider URL

– **Field Title**	OPMN request port
– **Field Description**	The OPMN request port found in $ORACLE_HOME/opmn/conf/opmn.xml <port local="6100" remote="6200" request="6003"/>
– **Example**	6003
– **Notes**	Used by installer scripts to deploy EAR and WAR files and to create default input for JNDI provider URL

– **Field Title**	OC4J Admin User
– **Destination**	jndi.cfg
– **Example**	oc4jadmin
– **Notes**	Used by installer scripts to deploy EAR and WAR files

– **Field Title**	OC4J Admin Password
– **Field Description**	The password of the OC4J Admin User
– **Destination**	jndi.cfg
– **Notes**	Used by installer scripts to deploy EAR and WAR files

Screen: Application Deployment Details



Fields on this screen:

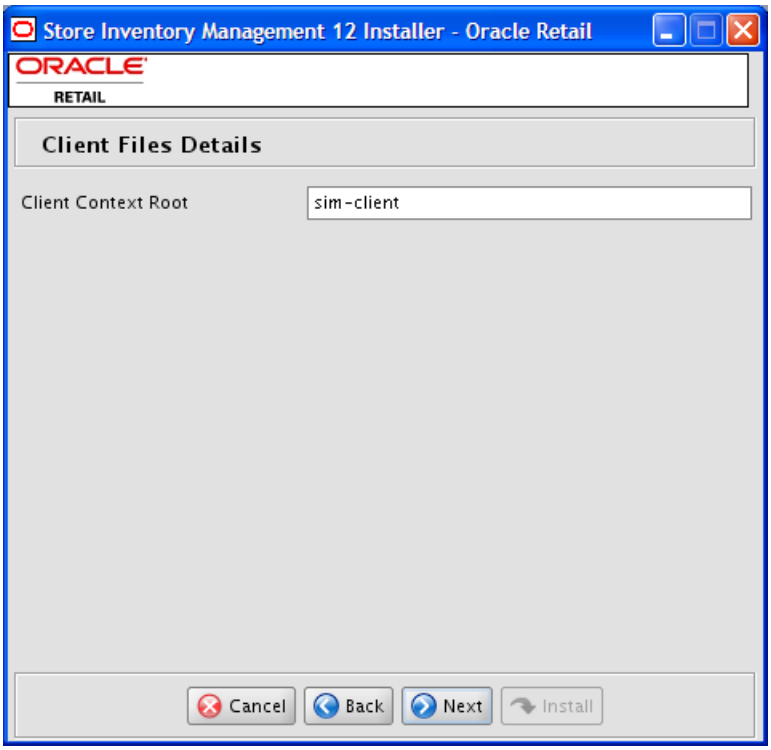
– **Field Title**	OC4J Group Name
– **Field Description**	Name of the OC4J group that was created for this SIM application. The OC4J instance given in the OC4J Instance Name field should be a member of this group. The installer will deploy the SIM application to all OC4J instances which are members of this group. For this reason, you should not use default_group. A new group dedicated to SIM should be created instead.
– **Example**	sim_group

– **Field Title**	OC4J Instance Name
– **Field Description**	The name of the OC4J instance that the SIM application will be deployed to
– **Destination**	log4j.xml, MANIFEST.MF, startup.sh, shutdown.sh,
– **Example**	sim-oc4j-instance

– **Field Title**	Application Deployment Name
– **Field Description**	The name that will be used by the application server to identify the SIM application
– **Example**	sim12
– **Notes**	Used by installer scripts to deploy the application and to create default values for JNDI provider URL

– **Field Title**	Client EAR Deployment Name
– **Field Description**	The name that will be used by the application server to identify the SIM client
– **Example**	sim-client
– **Notes**	Used by installer scripts to deploy the sim-client application and to create default values for JNDI provider URL

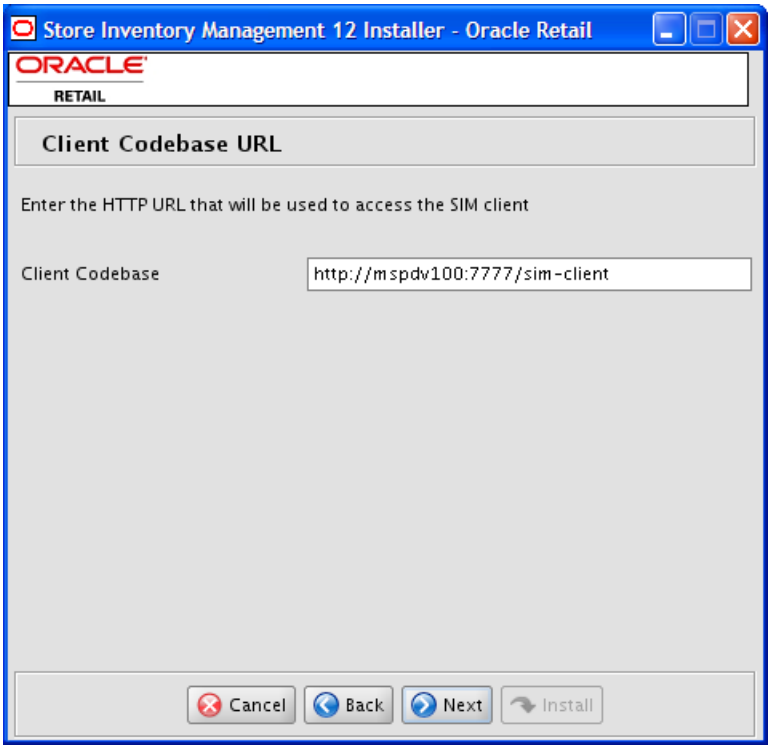
Screen: Client Files Details



Fields on this screen:

–	**Field Title**	Client Context Root
–	**Field Description**	Context root for sim-client.war
–	**Destination**	client_master.cfg
–	**Example**	sim-client
–	**Notes**	Used by installer to create default value for Client Codebase URL

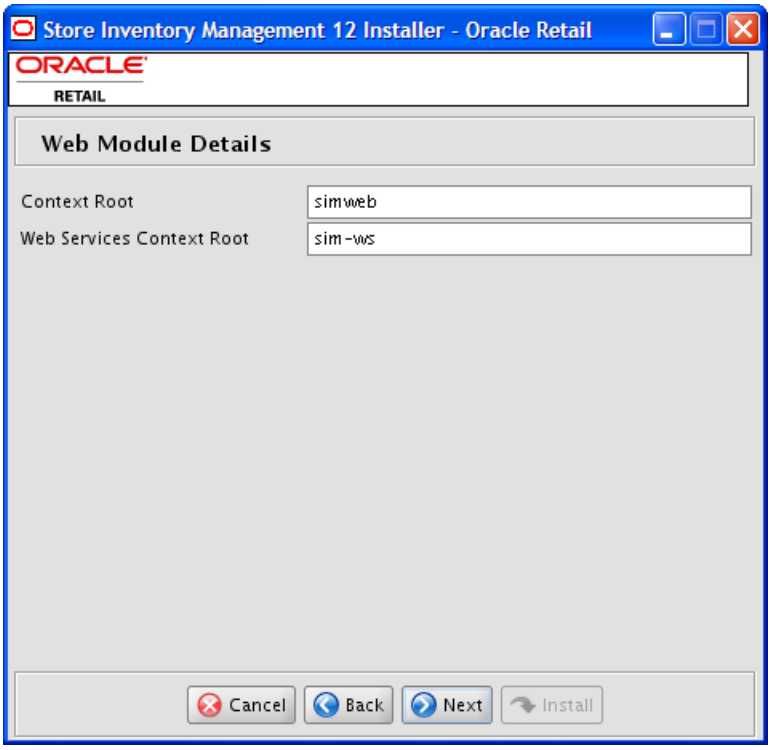
Screen: Client Codebase URL



Fields on this screen:

–	**Field Title**	Client Codebase
–	**Field Description**	The HTTP URL that points to the SIM client installation. The URL is made up of the Hostname, the HTTP port, and the Client Context Root.
–	**Destination**	JNLPLaunch.properties, sim_config.jnlp, client_master.cfg
–	**Example**	http://myhost:7777/sim-client
–	**Notes**	The Client Codebase URL must match the Client Context Root from the previous screen

Screen: Web Module Details

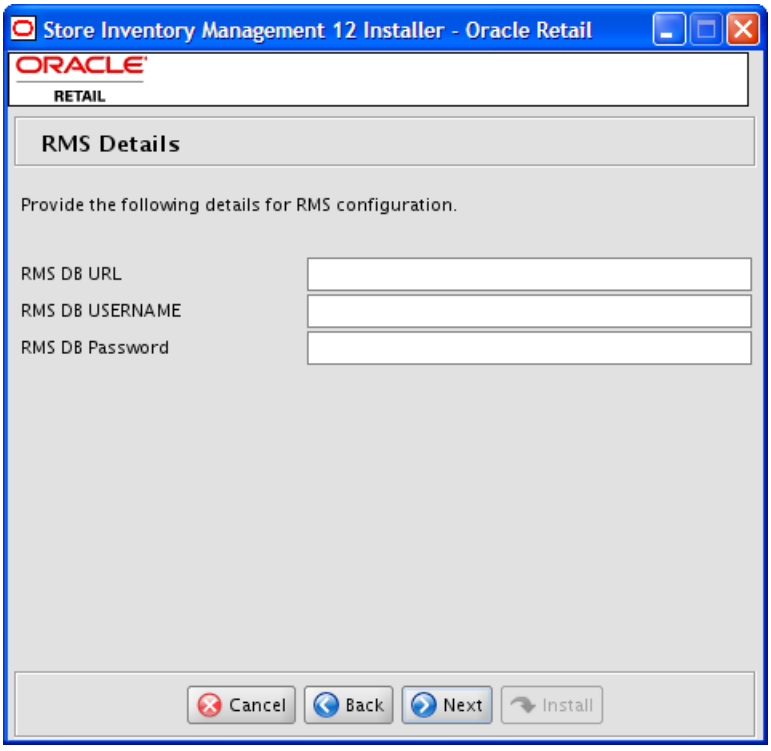


Fields on this screen:

–	**Field Title**	Context Root
–	**Field Description**	The context root for sim.war
–	**Destination**	application.xml
–	**Example**	Simweb

–	**Field Title**	Web Services Context Root
–	**Field Description**	The context root for sim-ws.war
–	**Destination**	application.xml
–	**Example**	sim-ws

Screen: RPM Details



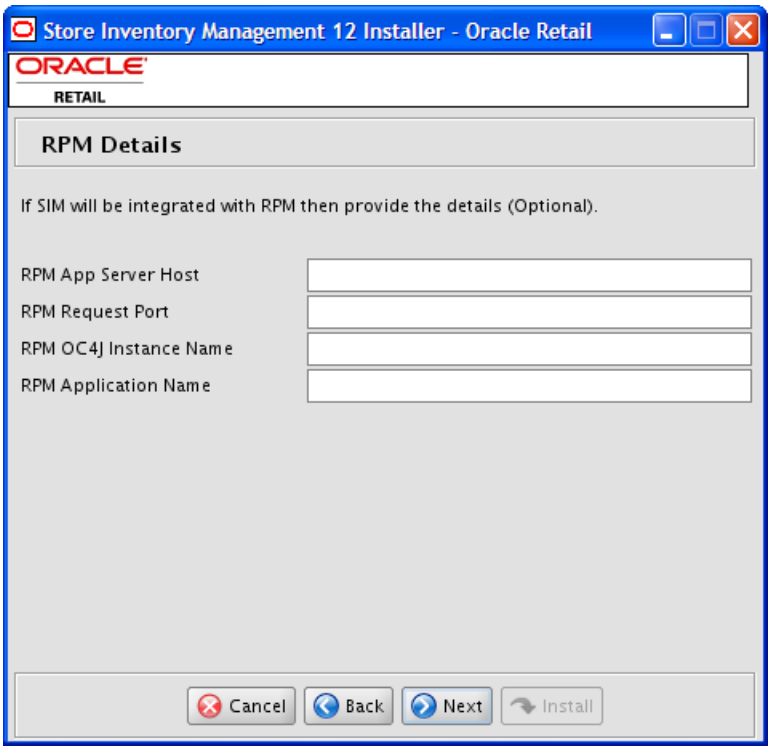
Fields on this screen:

– **Field Title**	RMS DB URL
– **Field Description**	The JDBC URL RMS database installed
– **Destination**	jndi_providers.xml
– **Example**	Jdbc:oracle:thin:@myhost.us.oracle.com:1521:mydb
– **Notes**	Used only if integrating SIM with RMS

– **Field Title**	RPM DB USERNAME
– **Field Description**	The schema username to the RMS database
– **Destination**	jndi_providers.xml
– **Example**	myschema
– **Notes**	Used only if integrating SIM with RMS

– **Field Title**	RPM DB Password
– **Field Description**	The schema password to the RMS database
– **Destination**	jndi_providers.xml
– **Example**	mypassword
– **Notes**	Used only if integrating SIM with RMS

Screen: RPM Details



Fields on this screen:

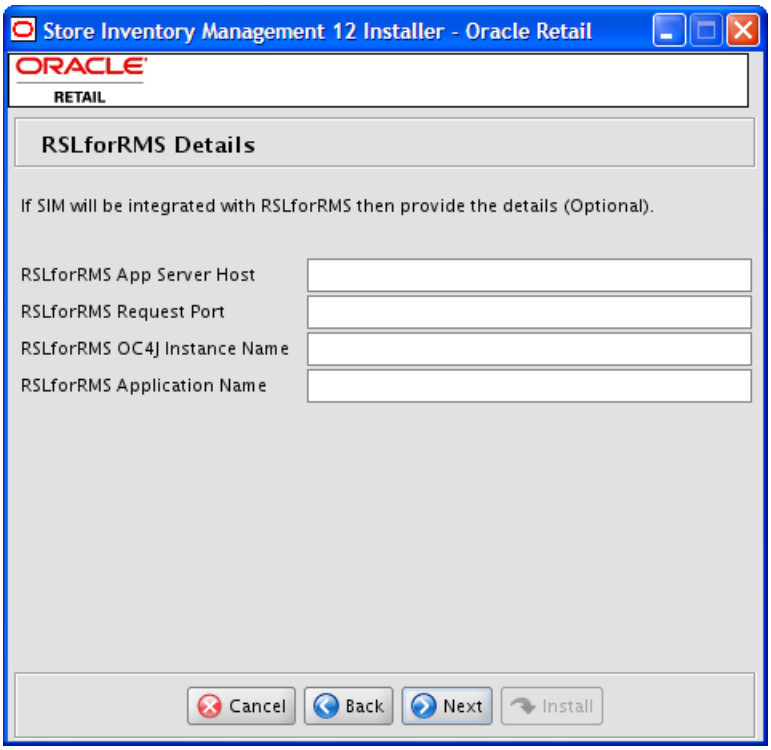
–	**Field Title**	RPM App Server Host
–	**Field Description**	The name of the application server host where the RPM application is installed
–	**Destination**	jndi_providers.xml
–	**Example**	myhost.us.oracle.com
–	**Notes**	Used only if integrating SIM with RPM

– **Field Title**	RPM Request Port
– **Field Description**	The OPMN request port for the application server where RPM is intalled. The OPMN request port is found in $ORACLE_HOME/opmn/conf/opmn.xml <port local="6100" remote="6200" request="6003"/>
– **Destination**	jndi_providers.xml
– **Example**	6003
– **Notes**	Used only if integrating SIM with RPM

– **Field Title**	RPM OC4J Instance Name
– **Field Description**	The name of the OC4J instance where the RPM application is installed
– **Destination**	jndi_providers.xml
– **Example**	rpm-o4cj-instance
– **Notes**	Used only if integrating SIM with RPM

– **Field Title**	RPM Application Name
– **Field Description**	The name that will be used by the application server to identify the RPM application
– **Destination**	jndi_providers.xml
– **Example**	rpm12
– **Notes**	Used only if integrating SIM with RPM

Screen: RSLforRMS Details



Fields on this screen:

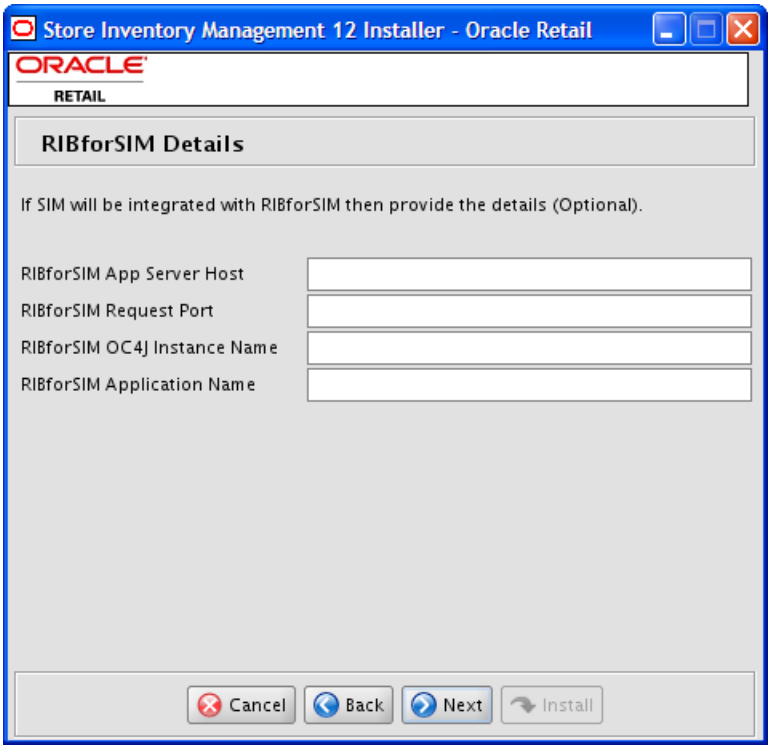
–	**Field Title**	RSLforRMS App Server Host
–	**Field Description**	The name of the application server host where the RSLforRMS application is installed
–	**Destination**	jndi_providers.xml
–	**Example**	myhost.us.oracle.com
–	**Notes**	Used only if integrating SIM with RSLforRMS

– **Field Title**	RSLforRMS Request Port
– **Field Description**	The OPMN request port for the application server where RSLforRMS is intalled. The OPMN request port is found in $ORACLE_HOME/opmn/conf/opmn.xml <port local="6100" remote="6200" request="6003"/>
– **Destination**	jndi_providers.xml
– **Example**	6003
– **Notes**	Used only if integrating SIM with RSLforRMS

– **Field Title**	RSLforRMS OC4J Instance Name
– **Field Description**	The name of the OC4J instance where the RSLforRMS application is installed
– **Destination**	jndi_providers.xml
– **Example**	rsl-rms-o4cj-instance
– **Notes**	Used only if integrating SIM with RSLforRMS

– **Field Title**	RSLforRMS Application Name
– **Field Description**	The name that will be used by the application server to identify the RSLforRMS application
– **Destination**	jndi_providers.xml
– **Example**	rsl-rsm
– **Notes**	Used only if integrating SIM with RSLforRMS

Screen: RIBforSIM Details



Fields on this screen:

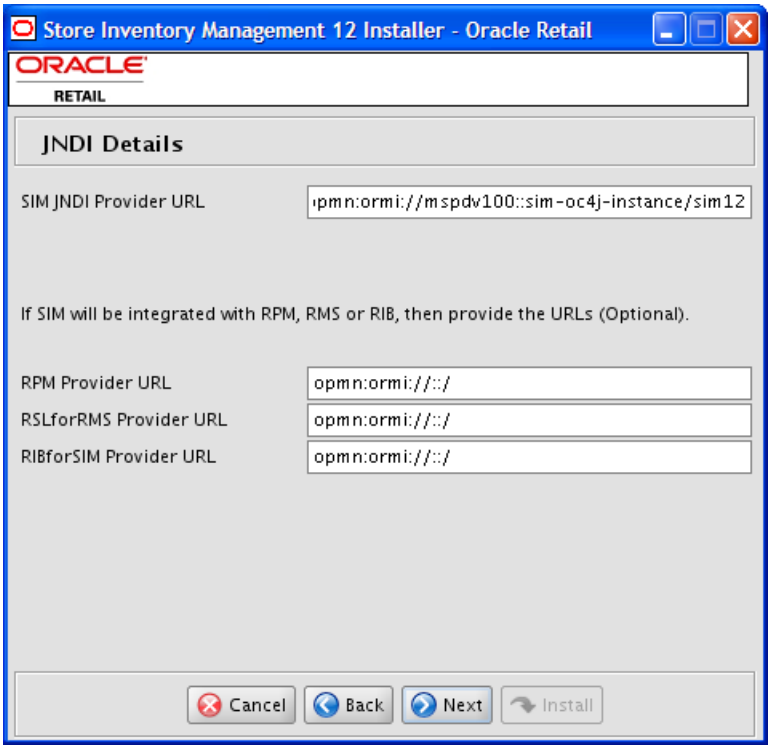
–	**Field Title**	RIBforSIM App Server Host
–	**Field Description**	The name of the application server host where the RIBforSIM application is installed
–	**Destination**	jndi_providers.xml
–	**Example**	myhost.us.oracle.com
–	**Notes**	Used only if integrating SIM with RIBforSIM

– **Field Title**	RIBforSIM Request Port
– **Field Description**	The OPMN request port for the application server where RIBforSIM is intalled. The OPMN request port is found in $ORACLE_HOME/opmn/conf/opmn.xml <port local="6100" remote="6200" request="6003"/>
– **Destination**	jndi_providers.xml
– **Example**	6003
– **Notes**	Used only if integrating SIM with RIBforSIM

– **Field Title**	RIBforSIM OC4J Instance Name
– **Field Description**	The name of the OC4J instance where the RIBforSIM application is installed
– **Destination**	jndi_providers.xml
– **Example**	rib-sim-o4cj-instance
– **Notes**	Used only if integrating SIM with RIBforSIM

– **Field Title**	RIBforSIM Application Name
– **Field Description**	The name that will be used by the application server to identify the RIBforSIM application
– **Destination**	jndi_providers.xml
– **Example**	rib-sim
– **Notes**	Used only if integrating SIM with RIBforSIM

Screen: JNDI Details



Fields on this screen:

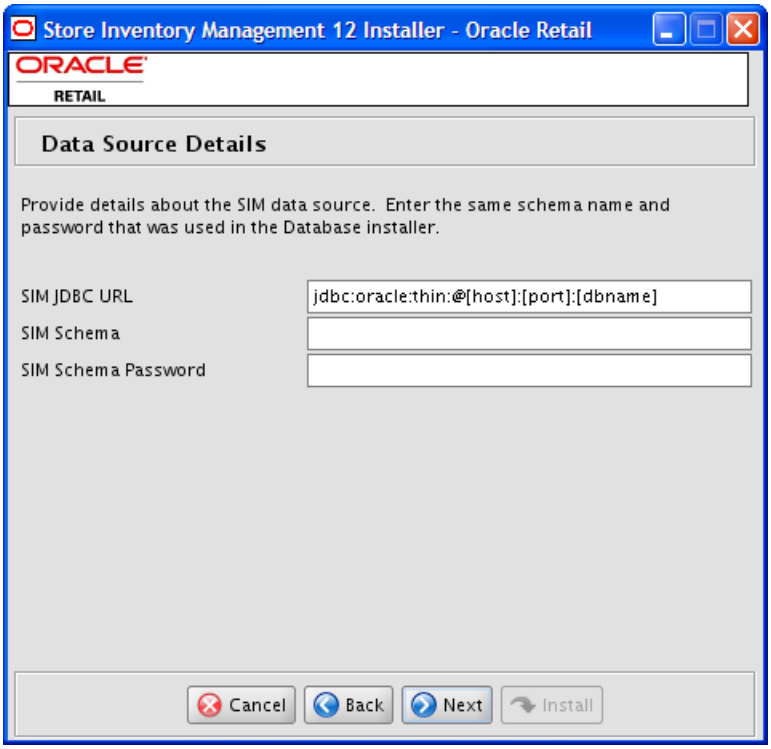
–	**Field Title**	SIM JNDI Provider URL
–	**Field Description**	JNDI provider URL for the SIM application
–	**Destination**	jndi.cfg, JnlpLaunch.properties
–	**Example**	opmn:ormi://myhost.us.oracle.com:6003:sim-oc4j-instance/sim12
–	**Notes**	Confirm the JNDI provider URL, which is constructed based on previous inputs for Hostname, OPMN Request Port, OC4J Instance Name, and Application Deployment Name

– **Field Title**	RPM Provider URL
– **Field Description**	JNDI provider URL for the RPM application
– **Destination**	jndi_providers.xml
– **Example**	opmn:ormi://myhost.us.oracle.com:6003:rpm-oc4j-instance/rpm12
– **Notes**	Confirm the JNDI provider URL, which is constructed based on previous inputs for Hostname, OPMN Request Port, OC4J Instance Name, and Application Deployment Name

– **Field Title**	RSLforRMS Provider URL
– **Field Description**	JNDI provider URL for the RSLforRMS application
– **Destination**	jndi_providers.xml
– **Example**	opmn:ormi://myhost.us.oracle.com:6003:rsl-rms-oc4j-instance/rsl
– **Notes**	Confirm the JNDI provider URL, which is constructed based on previous inputs for Hostname, OPMN Request Port, OC4J Instance Name, and Application Deployment Name

– **Field Title**	RIBforSIM Provider URL
– **Field Description**	JNDI provider URL for the RIBforSIM application
– **Destination**	jndi_providers.xml
– **Example**	opmn:ormi://myhost.us.oracle.com:6003:rib-sim-oc4j-instance/rib-sim
– **Notes**	Confirm the JNDI provider URL, which is constructed based on previous inputs for Hostname, OPMN Request Port, OC4J Instance Name, and Application Deployment Name

Screen: Data Source Details



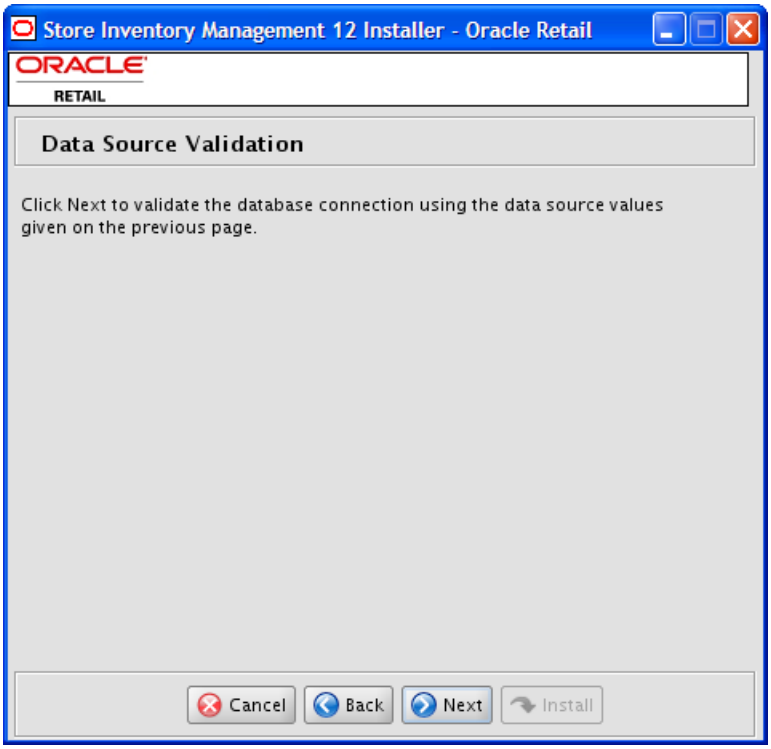
Fields on this screen:

– **Field Title**	SIM JDBC URL
– **Field Description**	URL used by the SIM application to access the SIM database schema.
– **Destination**	batch_db.cfg, data-sources.xml
– **Example**	jdbc:oracle:thin:@myhost:1525:mydatabase

– **Field Title**	SIM Schema
– **Field Description**	The schema name
– **Destination**	batch_db.cfg, data-sources.xml
– **Notes**	The schema name should match the name you provided when you ran the database schema installer.

– **Field Title**	SIM Schema Password
– **Field Description**	The password for the SIM Schema
– **Destination**	batch_db.cfg, data-sources.xml

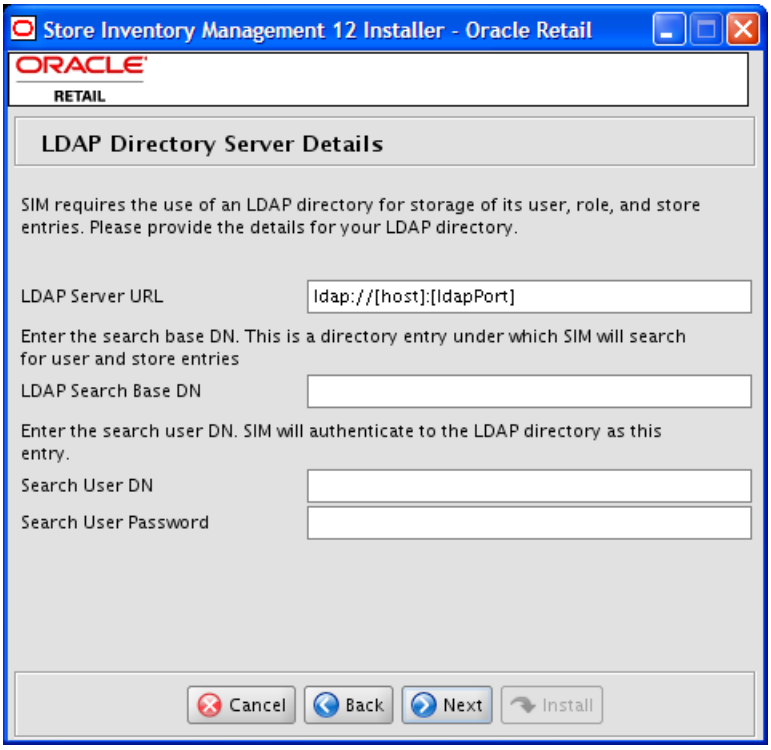
Screen: Data Source Validation



Fields on this screen:

This screen contains no input fields. It is used to verify the data source information that was provided on the previous screen. Just click Next to continue.

Screen: LDAP Directory Server Details



Fields on this screen:

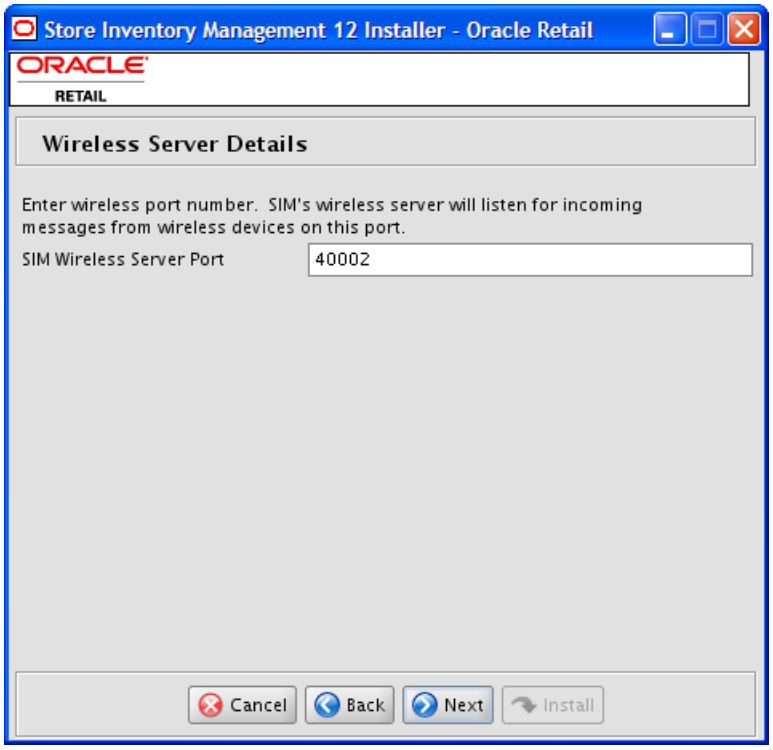
– **Field Title**	LDAP Server URL
– **Field Description**	URL for your LDAP directory server. See Appendix E: URL Reference for expected syntax.
– **Destination**	ldap.cfg
– **Example**	ldap://myhost:389

– **Field Title**	LDAP Search Base DN
– **Field Description**	Distinguished name of the LDAP directory entry under which SIM should search for users.
– **Destination**	ldap.cfg
– **Example**	cn=Users,dc=mycompany,dc=com

– **Field Title**	Search User DN
– **Field Description**	Distinguished name of the user that SIM will use to authenticate to the LDAP directory.
– **Destination**	ldap.cfg
– **Example**	cn=admin,dc=mycompany,dc=com

– **Field Title**	Search User Password
– **Field Description**	Password for the search user DN.
– **Destination**	ldap.cfg

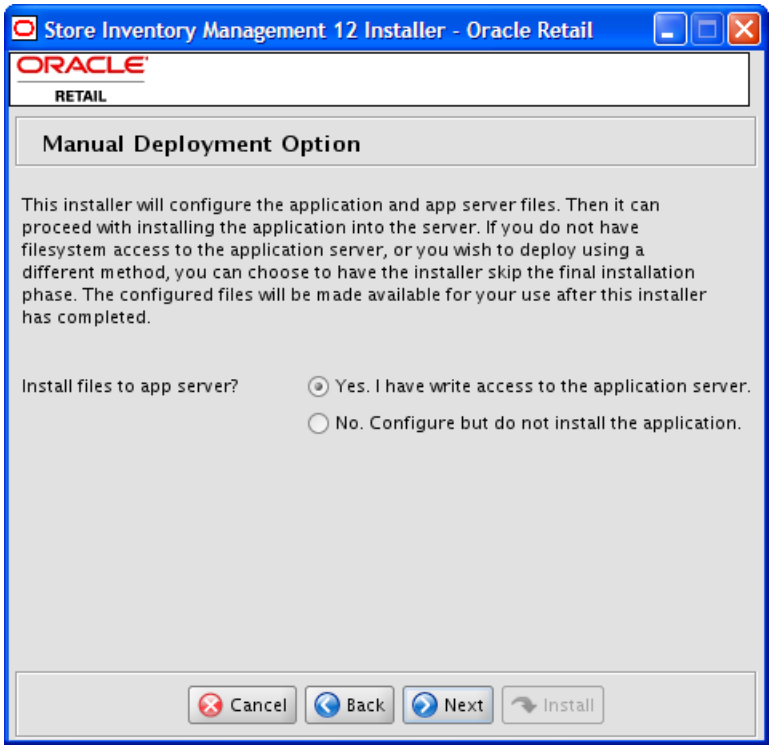
Screen: Wireless Server Details



Fields on this screen:

–	**Field Title**	SIM Wireless Server Port
–	**Field Description**	Choose an available port that the Wavelink server will use to listen for incoming messages from wireless devices
–	**Destination**	wireless_services.cfg, wavelink-startup.sh
–	**Example**	40002

Screen: Manual Deployment Options



Fields on this screen:

– **Field Title**	Install files to app server?
– **Field Description**	If you are running the installer as a user who doesn't have permissions to write to the filesystem under the ORACLE_HOME, then choose "No" to have the installer perform all the configuration within the staging directory but not install any files into the ORACLE_HOME.
– **Notes**	If you choose "No", you will need to perform manual steps to complete the installation.

C

Appendix: Installer Silent Mode

Repeating an Installation Attempt

In addition to the GUI and text interfaces of the 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. Run the installer again with the silent argument.

Example: install.sh silent

D

Appendix: URL Reference

Both the database schema and application installers ask for several different URLs. These include the following.

JDBC URL for a Database

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

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

HTTP URL for a WebStart Client

Used within a web browser to access the application client.

Syntax: http://<host>:<port>/<client-context-root>/launch?template=sim_jnlp_template.vm

- <host>: hostname of the OracleAS environment
- <port>: HTTP port for the Oracle Http Server (OHS). This can be found in the Listen parameter in the ORACLE_HOME/Apache/Apache/conf/httpd.conf file, or in the output of opmnctl status –l.
- <client-context-root>: The context root for sim-client.war

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.

Syntax: opmn:ormi://<host>:<port>:<instance>/<app>

- <host>: hostname of the OracleAS environment
- <port>: OPMN request port of the OracleAS environment. This can be found in the <ORACLE_HOME>/opmn/conf/opmn.xml file.
- <instance>: Name of the OC4J instance running the application
- <app>: Deployment name for the application.

Example: opmn:ormi://myhost:6003:sim-oc4j-instance/sim12

Note: The JNDI provider URL can have a different format depending on your cluster topology. Consult the Oracle Application Server documentation for further details.

Deployer URI

The deployer URI is used by the Oracle ANT tasks to deploy an application to an OC4J group. The application installer does not ask the user for this value; it is constructed based on other inputs and written to the ant.install.properties file for input to the installation script. For repeat installations using silent mode, you may need to correct mistakes in the deployer URI.

Note: There are several different formats for the deployer URI depending on your cluster topology. Consult the *Deploying with the OC4J Ant Tasks* chapter of the *OC4J Deployment Guide* for further details.

Syntax (managed OC4J): deployer:cluster:opmn://<host>:<port>/<group>

- <host>: hostname of the OracleAS environment
- <port>: OPMN request port of the OracleAS environment. This can be found in the <ORACLE_HOME>/opmn/conf/opmn.xml file.
- <group>: Name of the OC4J group where the application will be deployed.

Example: deployer:cluster:opmn://myhost:6003/sim_group

Syntax (standalone OC4J):

deployer:oc4j:<host>:<port>

- <host>: hostname of the OracleAS environment
- <port>: RMI port of the OC4J server. This can be found in the ORACLE_HOME/j2ee/home/config/rmi.xml file.

Example: deployer:oc4j:myhost:23791

E

Appendix: Common Installation Errors

This section provides some common errors encountered during installation.

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 JDK. Set JAVA_HOME to a Java development kit of version 1.4.2 or later and run the installer again.

"Unable to get a deployment manager" Message

Symptom:

The application installer quits with the following error message:

```
[oracle:deploy] Unable to get a deployment manager.
[oracle:deploy]
[oracle:deploy] This is typically the result of an invalid deployer URI
format being supplied, the target server not being in a started state or
incorrect authentication details being supplied.
[oracle:deploy]
[oracle:deploy] More information is available by enabling logging --
please see the Oracle Containers for J2EE Configuration and
Administration Guide for details.
```

Solution:

This error can be caused by any of the following conditions:

- - OC4J instance provided is not running.
- - Incorrect OC4J instance name provided
- - Incorrect OC4J administrative username and/or password
- - Incorrect OPMN request port provided.

Make sure that the OC4J instance is running, and then check the **ant.install.properties** file for entry mistakes. Pay close attention to the input.deployer.uri (see Appendix E: *URL Reference*), input.oc4j.instance, input.admin.user, and input.admin.password properties. If you need to make a correction, you can run the installer again with this file as input by running silent mode (see Appendix D of this document).

"Could not create system preferences directory" Warning

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.

Keystore Errors When Signing sim-config.jar

Symptom:

keytool error: java.io.IOException: Keystore was tampered with, or password was incorrect

Solution:

This message may be encountered when you use the keytool utility to create an alias for signing the sim-config.jar file. This usually happens when the alias for which you are generating a key already exists in the keystore file.

Delete or rename the ~/.keystore file and run the keytool command again. This creates a fresh keystore file.

"Couldn't find X Input Context" Warnings

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.

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.

Error while Unpacking the sim12.ear

Symptom:

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

07/12/19 10:53:17 Notification ==>Error while unpacking sim12.ear
java.util.zip.ZipException: error in opening zip file
```

Solution:

This is a known bug (BugID 6330834) related to Solaris and NFS in Oracle Application Server 10.1.3.3. Follow the workaround documented for this bug: in the opmn.xml file in $ORACLE_HOME/opmn/conf add the following parameter to the java-options for the instance you are installing.

-Doc4j.autoUnpackLockCount=-1

After making this change you should reload OPMN, restart the affected OC4J instance(s), and retry the retail application installation.