# Oracle® Retail Store Inventory Management

Installation Guide

Release 13.0

April 2008

ORACLE

Oracle® Store Inventory Management Installation Guide, Release 13.0



Primary Author: Paul Kehler

## Value-Added Reseller (VAR) Language

(i) the software component known as **ACUMATE** developed and licensed by Lucent Technologies Inc. of Murray Hill, New Jersey, to Oracle and imbedded in the Oracle Retail Predictive Application Server – Enterprise Engine, Oracle Retail Category Management, Oracle Retail Item Planning, Oracle Retail Merchandise Financial Planning, Oracle Retail Advanced Inventory Planning and Oracle Retail Demand Forecasting applications.

(ii) 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.

(iii) the **SeeBeyond** component developed and licensed by Sun MicroSystems, Inc. (Sun) of Santa Clara, California, to Oracle and imbedded in the Oracle Retail Integration Bus application.

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

(v) the software component known as **Crystal Enterprise Professional and/or Crystal Reports Professional** licensed by Business Objects Software Limited ("Business Objects") and imbedded in Oracle Retail Store Inventory Management.

(vi) 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.

(vii) 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.

(viii) the software component known as **Style Report™** developed and licensed by InetSoft Technology Corp. of Piscataway, New Jersey, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

(ix) the software component known as **i-net Crystal-Clear™** developed and licensed by I-NET Software Inc. of Berlin, Germany, to Oracle and imbedded in the Oracle Retail Central Office and Oracle Retail Back Office applications.

(x) the software component known as **WebLogic™** developed and licensed by BEA Systems, Inc. of San Jose, California, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

(xi) the software component known as **DataBeacon™** developed and licensed by Cognos Incorporated of Ottawa, Ontario, Canada, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

# Contents

# 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 13.0 documentation set:

- Oracle Retail Store Inventory Management Data Model
- Oracle Retail Store Inventory Management Implementation Guide
- Oracle Retail Store Inventory Management Release Notes
- Oracle Retail Store Inventory Management Operations Guide
- Oracle Retail Store Inventory Management User Guide
- 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

For a base release (".0" release, such as 13.0), Oracle Retail strongly recommends that you read all patch documentation before you begin installation procedures. Patch documentation can contain critical information related to the base release, based on new information 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:

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

## 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
- Data sparcity
- 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.

# Check Database Server Requirements

General Requirements for a database server running SIM include:

| Supported on: | Versions Supported: |
|---|---|
| Database Server OS | UNIX based OS certified with Oracle RDBMS 10g Enterprise Edition (options are Oracle Enteprise Linux 4 Patch 5) |
| Database Server | Oracle RDBMS 10g Release 2 Enterprise Edition (minimum 10.2.0.3 patchset required) with the following patches and components:<br>Patches:<br>▪ 5397953 (ORA-07445: [KKPAPITGETALL()+2152] [SIGSEGV] [ADDRESS NOT MAPPED TO OBJECT] [0X34)<br>▪ 5648872 (SCHEDULER ORA-07445 [OPIDSA()+321] WHEN SETTING UP CHAIN TEST)<br>▪ 5921386 (WRONG RESULT WITH MERGE JOINT OUTER IN THE EXECUTION PLAN)<br>RAC Only<br>▪ 5721821 (ORA-7445[KGLOBCL] OCCURED AND INSTANCE WENT DOWN)<br>Components:<br>▪ Oracle Database 10g<br>▪ Oracle Partitioning<br>▪ Oracle Net Services<br>▪ Oracle Call Interface (OCI)<br>▪ Oracle Programmer<br>▪ Oracle XML Development Kit<br>x-Windows interface |

# Check Application Server Requirements

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

| Supported on: | Versions Supported: |
|---|---|
| Application Server OS | UNIX based OS (Oracle Enterprise Linux 4 Patch 5 for Linux x86-64) certified with Oracle Application Server 10g 10.1.3.3. |
| Application Server | Oracle Application Server 10g 10.1.3.3 with the following patches:<br>▪ 5632264 (NEED UPDATED TIMEZONE FILES (VERSION 4) FOR MORE DST RULE CHANGES)<br>▪ 5398506 (RUNTIME EXCEPTION DID NOT ROLLBACK MESSAGE ON EGATE (SEEBEYOND) TOPIC) |

**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 Single Sign-On Requirements

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

If Single Sign-On is to be used, verify the Oracle Infrastructure Server 10g version 10.1.2.2 server has been installed. Verify the OAS HTTP server used to launch SIM has been registered with the Oracle Single Sign-On server and the mod_osso module has been enabled within the HTTP Server's configuration.

For more details on this, see the *Oracle Single Sign-On Administration Guide*.

## 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 10.1.2.2*

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 Business Intelligence Publisher Enterprise 10.1.3.3
- 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 | minimum1GHz |
| Memory | minimum of 512MBytes |
| Networking | Intranet with at least 10Mbps data rate |
| Sun JRE | 5.0 Update 11 or newer (1.5.0_11) |
| Microsoft Internet Explorer | version 5.5 or higher<br>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) 13.0 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) 13.0 – Optional
- Retail Service Layer for RMS (RSLforRMS) 13.0 – Optional
- Retail Price Management (RPM) 13.0 – 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

# RAC and Clustering

Real Application Cluster RDBMS & Oracle Application Server Clustering for SIM has been validated to run only on Linux.

The Oracle Retail products have been validated against a 10.2.0.3 RAC database. It is important to note that the OCI JDBC driver is not supported by SIM. Instead, the THIN driver should be used. To properly load balance with multiple database servers, the JDBC connection string URL should look like this example, with an ADDRESS entry in the ADDRESS_LIST for each database server available:

```
jdbc:oracle:thin:@(DESCRIPTION =
   (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = dbserver1)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = dbserver2)(PORT = 1521))
      (LOAD_BALANCE = yes))
   (CONNECT_DATA =(SERVICE_NAME = simprod01)))
```

This configuration does not provide transparent RAC failover. If a DB connection for one address is lost, the application will get a system exception, and the next time it attempts to access the database it will access one of the other addresses.

Clustering for Oracle Application Server 10.1.3 is managed as an Active-Active cluster accessed through a hardware Load Balancer. It is suggested that a VirtualHost be added to the OAS 10.1.3 reflecting the Virtual Server Name configured in the load balancer. It is also suggested that the OC4J select method be configured to prefer the use of local OC4J instances. The Oracle Retail products are currently not validated to be distributable at the application level in an OAS 10.1.3 cluster.

### References for Configuration:

- Oracle® Application Server High Availability Guide 10g Release 3 (10.1.3) Part Number B15977-02
- Oracle® Application Server High Availability Guide 10g Release 2 (10.1.2) Part Number B14003-05
- Oracle® Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide 10g Release 2 (10.2) Part Number B14197-03

# 3

# Database Installation Tasks

## Expand the SIM Database Schema Installer Distribution

1. Log into the UNIX server as a user which has sufficient access to run sqlplus from the Oracle RDBMS installation.
2. Create a new staging directory for the SIM database schema installer distribution (sim13dbschema.zip). There should be a minimum of 50 MB disk space available for the database schema installation files. This location is referred to as INSTALL_DIR for the remainder of this chapter.
3. Copy sim13dbschema.zip to <INSTALL_DIR> and extract its contents.

## Create Tablespaces

Before you run the SIM database schema installer, make sure that the following tablespaces have been created in the database: RETEK_INDEX, RETEK_DATA, USERS, and LOB_DATA. Below are sample tablespace creation statements for these tablespaces. Oracle Retail recommends the use of locally managed tablespaces with automatic extent and segment space management. These tablespaces are not sized for a production environment!

```
CREATE TABLESPACE RETEK_INDEX DATAFILE
    '<datafile_path>/retek_index01.dbf'  SIZE 500M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
CREATE TABLESPACE RETEK_DATA DATAFILE
    '<datafile_path>/retek_data01.dbf'  SIZE 500M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
CREATE TABLESPACE USERS DATAFILE
    '<datafile_path>/users01.dbf'  SIZE 100M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
CREATE TABLESPACE LOB_DATA DATAFILE
    '<datafile_path>/lob_data01.dbf'  SIZE 50M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
```

## Create the SIM Database User

The user in the database which will own the SIM tables must be created prior to running the SIM database schema installer. A create_user.sql script has been provided that can be used for this:

<INSTALL_DIR>/sim/dbschema/dbscripts/utility/create_user.sql

The script takes 3 arguments on the command line in sqlplus: username, password, and temporary tablespace.

> **Example:** SQL> @create_user.sql SIMUSER mypassword TEMP

Please review this script and run it as a user with adequate permissions, such as SYSTEM.

## Run the SIM Database Schema Installer

This installer installs the SIM database schema, compile SIM objects, insert SIM data, and produce the sim_dba.sql script.

1. Expand the sim13dbschema.zip distribution into <INSTALL_DIR>.
2. Change directories to sim/dbschema/dbscripts/db_objects, and use the insert_default_st_config_val.pls.template file to create a new file called insert_default_st_config_val.pls. (Copy insert_default_st_config_val.pls.template to insert_default_st_config_val.pls.)
3. If you are using a reporting tool with SIM, you need to configure the insert_default_st_config_val.pls file before running the SIM database schema installer. See the *Defaulting Store Configuration Parameters* section in *the Setup and Configuration* chapter of the *Implementation Guide* for more details. If you are not using a reporting tool with SIM, then skip this step.
4. Set the following environment variables:
   - Set the ORACLE_HOME to point to an installation that contains sqlplus. It is recommended that this be the ORACLE_HOME of the SIM database.
   - Set the PATH to: $ORACLE_HOME/bin:$PATH
   - Set the ORACLE_SID to the name of your database
   - Set the NLS_LANG for proper locale and character encoding

   > **Example:** NLS_LANG=AMERICAN_AMERICA.UTF8
5. 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.
6. Run the install.sh script. This launches the installer. After installation is completed, a detailed installation log file is created: <INSTALL_DIR>/ sim/dbschema/logs/sim-install-db.<timestamp>.log.

   > **Note:** Appendix B contains details on every screen and field in the database schema installer.
7. When the installer finishes it prints the values of the database SID and database schema user. Note these values as they are needed later when you run the SIM application installer.

8. The SIM database schema installer will produce a sim_dba.sql script which must be reviewed by a DBA and then run on the database server in order to complete the installation.
9. If you wish to run data seeding from your merchandising system (e.g. RMS), you should do so at this time. (See instructions below)

## Resolving Errors Encountered During Database Schema Installation

If the database schema installer encounters any errors, it halts execution immediately and print to the screen which SQL script it was running when the error occurred. It also writes the path to this script to the .dberrors file. When this happens, you must run that particular script using sqlplus. After you are able to complete execution of the script, delete the .dberrors file and run the installer again. You can run the installer in silent mode so that you do not 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.

Subsequent executions of the installer will skip the SQL scripts which have already been executed in previous installer runs. This is possible because the installer maintains a **.dbhistory** file with a listing of the SQL scripts that have been run. If you have dropped the SIM schema and want to start with a clean install, you can delete the .dbhistory file so that the installer runs through all of the scripts again. It is recommended that you allow the installer to skip the files that it has already run.

## Running Data Seeding

If your SIM server is going to integrate with RMS, you must run the DataSeeding utility at this time to load data from RMS into SIM.

Data seeding must be run from the server where the database is located. If you ran the database schema installer from a different server, then you will need to copy the entire data_seeding directory over to the database server. This data_seeding directory will be referred to as the DATA_SEEDING_DIR for the remainder of this chapter.

1. Set the following environment variables:
   - $RMS_USER/$RMS_PWD@$RMS_DB – These should be the values of the RMS that you are pulling data from.
   - Set the ORACLE_HOME to point to an installation that contains sqlldr. It is recommended that this be the ORACLE_HOME of the SIM database.
   - Set the PATH to: $ORACLE_HOME/bin:
   - Set the ORACLE_SID to the name of your database
2. Change to the <DATA_SEEDING_DIR> directory
3. Modify your <DATA_SEEDING_DIR>/database/oracle/data_seeding/sh/DataSeeding.sh
   - Set environment variables, userids, and passwords for SIM and RMS databases.
   - Set STORE_LIST=ALL (for running all stores)
   - Set SIM_COMMIT_BLOCK (Recommended: Commit Block 100000 for SIM_COMMIT_BLOCK)
   - DIRECT_SQL_LOAD=FALSE (default is set to FALSE)
4. Run CreateRMSIndexes.sh to create indexes from RMS. These are temporary indexes for the purpose of data seeding.
5. Do not run statistics on indexes above. Please review CreateRMSIndexes.sql.

6. Run the command: nohup ./DataSeeding.sh &

> **Note:** The nohup command will redirect the output of the DataSeeding.sh script to a nohup.out file

7. Run DropRMSIndexes.sh to drop the temporary indexes from RMS.
8. Check the SIM schema for any disabled constraints.

> **Example:** SELECT * from DBA_CONSTRAINTS where OWNER='SIMUSER' and STATUS='DISABLED'

The above query should return no rows. If there are any disabled constraints, you should enable them at this time.

> **Example:** ALTER TABLE <TABLE> enable constraint <CONSTRAINT>

> **Post-Data-Seeding-Clean-up:** After data seeding is finished, and you are convinced that your data was correctly seeded, you can run DataSeedClear.sh to remove dat, dsc, log and bad files.

There is a known issue with data seeding for the SIM 13.0 release.

After data seeding, recheck the RK_ITM_LCN and RK_CODE_DETAIL tables:

- If data is not populated in the RK_ITM_LCN table, execute data_seeding/database/oracle/data_seeding/sql/sequencing_pop_rk_itm_lcn.sql and sequencing_pop_location_count.sql in the order specified in the sqlplus section.
- If data is not populated in the RK_CODE_DETAIL table, make sure all environment variables in DataSeeding.sh are set before executing the following command:

```
sqlldr userid=$SIM_USER/$SIM_PWD@$SIM_DB control=../ctl/RkCodeDetail.ctl
log=../logs/RkCodeDetail.log ROWS=$SIM_COMMIT_BLOCK direct=false
```

Metalink note **19499.1** contains more information. Access Metalink at the following URL:

http://metalink.oracle.com

If the problem noted in Metalink note 19499.1 occurs, or if you want to completely rerun data seeding, follow these steps:

1. Drop the schema.
2. Drop the sim directory in the install drive.
3. Unzip sim13dbschema.zip
4. Re-create the schema.
5. Rerun DataSeeding.sh.
6. Verify that the RK_ITM_LCN and RK_CODE_DETAIL tables are populated. If not, follow the two steps specified above.

# 4

# Application Installation UNIX/Linux (OEL)

Before proceeding you must install Oracle Application Server 10g 10.1.3.3 plus the patches listed in Chapter 1 of this document. The SIM application is deployed to an OC4J instance within the OracleAS 10g installation. It is assumed Oracle RDBMS has already been configured and loaded with the appropriate SIM schema for your installation.

## Create a New OC4J Instance and Group for SIM

You can skip this section if you are redeploying to an existing OC4J group in Oracle Application Server 10.1.3.3.

The SIM application must be deployed to its own dedicated OC4J group. For instructions on how to create a new OC4J group and 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 and group.

   **Example:** sim-oc4j-instance

   **Example:** sim_group

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

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

   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 group. 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 @cluster startproc ias-component=sim_group

6. Verify that the OC4J group 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 cannot 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

## Clustered Installations -- Pre-Install Steps

Skip this section if you are not clustering the application server.

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

1. On each remote node, create a new sim-home directory in the following location: $ORACLE_HOME/j2ee/<sim-oc4j-instance>/sim-home (The directory must be called "sim-home.")
2. Copy the sim-home.zip file into the new sim-home directory. $ORACLE_HOME/j2ee/<sim-oc4j-instance>/sim-home/sim-home.zip

   **Note:** The sim-home.zip file can be found inside the sim13application.zip under sim/application/sim13/sim-home.zip

3. Unzip the sim-home.zip file into the sim-home directory.

## 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 (sim13application.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 sim13application.zip to <INSTALL_DIR> and extract its contents.

## Run the SIM Application Installer

This installer configures and deploys the SIM application and Java WebStart client files.

1. If you are installing to a clustered Application Server, perform the pre-install tasks as described in the *Clustered Installations -- Pre-Install Steps* section above.
2. Expand the sim13application.zip distribution into <INSTALL_DIR>.
3. 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.
4. 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.
5. Verify that the OC4J instance(s) that you install SIM to are currently running.
6. 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.

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

7. If you are installing to a clustered Application Server, perform the post-install tasks as described in the *Clustered Installations -- Post-Install Steps* section below.
8. Sign the sim-config.jar file. (See instructions below.)
9. 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.)

## Clustered Installations -- Post-Installation Steps

Skip this section if you are not clustering the application server.

If you are installing the SIM 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 whose ORACLE_HOME you used for the SIM installer is referred to as the *master node*. All other nodes are referred to as the *remote nodes*.

1. Stop all of the SIM OC4J instances in the group

   **Example:** `$ORACLE_HOME/opmn/bin/opmnctl @cluster stopproc ias-component=sim_group`

2. On each remote node (but not on the master node), move or delete the sim-home directory

   **Example:**

   ```
   cd $ORACLE_HOME/j2ee/<sim-oc4j-instance>
   mv sim-home sim-home.old
   ```

3. For each remote node, copy the entire sim-home directory from the master node to the remote node under the same path as on the master node. ($ORACLE_HOME/j2ee/<sim-oc4j-instance>/sim-home)
4. On each remote node, the following files need to be modified so that the correct host name is used:
   - $ORACLE_HOME/j2ee/<sim-oc4j-instance>/applications/sim-client/sim-client/sim_config.jnlp
   - $ORACLE_HOME/j2ee/<sim-oc4j-instance>/sim-home/files/prod/config/JnlpLaunch.properties
   - $ORACLE_HOME/j2ee/<sim-oc4j-instance>/sim-home/files/prod/config/jndi.cfg
   - $ORACLE_HOME/j2ee/<sim-oc4j-instance>/sim-home/files/prod/config/client_master.cfg
5. On each remote node, the client_master.cfg file needs to be modified so that the correct host name is used. The client_master.cfg file is located inside the sim-config.jar file under the $ORACLE_HOME/j2ee/<sim-oc4j-instance>/applications/sim-client/sim-client/lib directory.

   **Example:**

```
cd $ORACLE_HOME/j2ee/<sim-oc4j-
instance>/applications/sim-client/sim-
client/lib

cp sim-config.jar sim-config.jar.old

mkdir temp

cd temp

jar xf ../sim-config.jar
```

Modify the host name in the client_master.cfg file

```
jar uf ../sim-config.jar client_master.cfg
```

6. Start all of the SIM OC4J instances in the group

**Example:** `$ORACLE_HOME/opmn/bin/opmnctl @cluster startproc ias-component=sim_group`

7. On every node, sign the sim-config.jar file. (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.

## Review and/or Configure Oracle Single Sign-On

Skip this section if you are not using Single Sign-On for user identification and authentication.

Single Sign-On is applicable only to the JnlpLaunch Servlet. The JnlpLaunch Servlet is a dynamically protected application. The JnlpLaunch Servlet causes the SIM client application to execute under the SSO user name with a temporary password.

**Note:** The JnlpLaunch servlet may be configured for either an SSO or non-SSO environment.

**HTTP Server configuration requirements:** The HTTP Server must be registered with the Oracle Single Sign-On server and the mod_osso module enabled. The registration process typically involves running the ssoreg.sh script at the OSSO server installation and copying the output osso.conf file to the HTTP Server. This process is documented in the Oracle Single Sign-On administration documentation.

**JnlpLaunch requirements:** The JnlpLaunch Servlet uses the configuration file, "JnlpLaunch.properties", to control its behavior. Due to security considerations, this file must not be published or readable to the general public.

JnlpLaunch.properties has the following configuration entries that apply to Single Sign-On:

- secret.key – Used to create the temporary password, this property should contain a random string. If JnlpLaunch is deployed in a different JVM than the SIM Server EJBs, this string must be an exact match between the JnlpLaunch Servlet and the one available to the SIM EJBs. For security purposes, each separate instance of the SIM application (e.g. test versus development) should have a different secret key.
- user.validation.timeout – Number of seconds the SIM Server uses to determine if a temporary password is still valid.
- osso.used – Determines if the JnlpLaunch Servlet will throw a 499 error when an unathenticated user has been detected. This property must be set to 'true' if Oracle Single Sign-On is used and 'false' if not.

The JnlpLaunch.properties file is initialized by the SIM installer and should contain valid entries for SSO when the "Enable Single Sign-On in SIM?" prompt was answered by a 'Y' or 'Yes'. However, an administrator may want to alter the `user.validation.timeout` or other property after the initial installation.

## 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 do not 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 are overwritten by the successful installation.

## Oracle Configuration Manager

The first OCM collector distribution that will be aware of the Oracle Retail applications is in development. This version of OCM is scheduled to be posted for download but is not yet available. Oracle Retail recommends that retailers download OCM 10.3.0 from ARU and use the "emCCR update_components" command to upgrade installed OCM collectors. See the OCM Installation and Administration Guide for further instructions. The Retail OCM Installer released with Oracle Retail 13.0 applications will install OCM 10.2.7. If the collector remains at version 10.2.7 and is installed in connected mode, an automatic update to version 10.3.0 is expected to occur later this year, the time at which 10.3.0 becomes a mandatory upgrade.

For more information, see the following:

Metalink Note: 559539.1

The Oracle Configuration Manager Installer Guide describes the procedures and interface of the Oracle Retail Oracle Configuration Manager Installer that a retailer runs near the completion of its installation process.

## 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 makes the configured application files available under <INSTALL_DIR>/sim/application/sim13/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/sim13/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/sim13/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/sim13/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 backs up the sim-home directory if it finds a previous installation of SIM. The backups are made by adding a suffix to 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

## 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.

## Web Help Files

The application installer automatically copies the web 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 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 will 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:** The wireless functionality in SIM is dependent on Wavelink and includes a client and server component. Wavelink software ensures that the wireless user interface of SIM can work with various handheld devices.

For the handheld to interact correctly with SIM, it is required to install the appropriate Wavelink studio client. The Wavelink studio client and its installation instructions can be found at http://www.wavelink.com/download/downloads.aspx.

The Oracle Retail Wireless Foundation Server is bundled with the SIM server. It has a single session free license. For multiple sessions additional licenses need to be obtained.

Please contact your Oracle sales representative or client partner for Wavelink Studio Client and Oracle Retail Wireless Foundation Server license information.

**Note:** For configurations of physical handheld devices or wireless network setup, check your hardware manufacturer's manual or Wavelink's studio client information. This information is not covered in the *SIM Installation Guide*.

**Note:** For adiditonal information about LDAP configuration please refer to the Implementaion Guide.

# 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, SSO toggle 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>

- To enable/disable SSO in SIM, there are 2 properties:

  osso_used=true

  token.sso_enabled=true

## 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 or RMS, 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>"

## RIB JNDI Providers in jndi_providers_ribclient.xml

If SIM is integrated with RIB, then the jndi_providers_ribclient.xml is configured similar to the following.

**Example:** name="java.naming.provider.url" value="opmn:ormi://<host>:<opmn-req-port>:<rib-sim-oc4j-instance-name>/<rib-sim-app-name>"

**Example:** name="java.naming.security.principal" value="oc4jadmin"

**Example:** name="java.naming.security.credentials" value="<oc4jadmin-password>"

## 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"/>
```

## 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

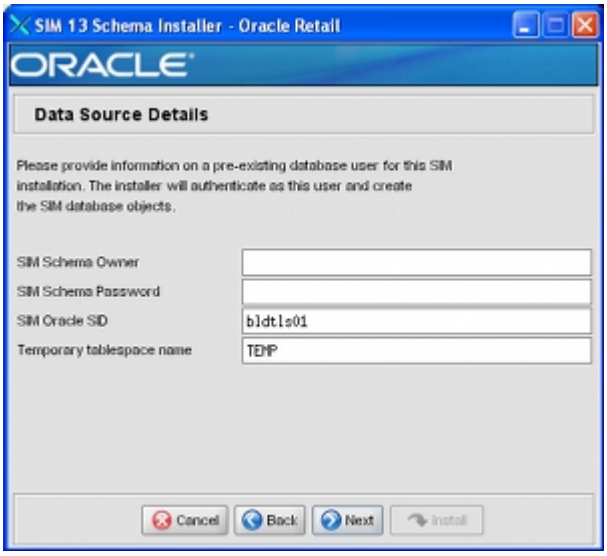
DEXNEX_ERRORS_DIR=<path-to-alternate-sim-home>/files/prod/dexnex/dexnex-error

# B

# Appendix: SIM Database Schema Installer Screens

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

Screen: Data Source Details




Fields on this screen:

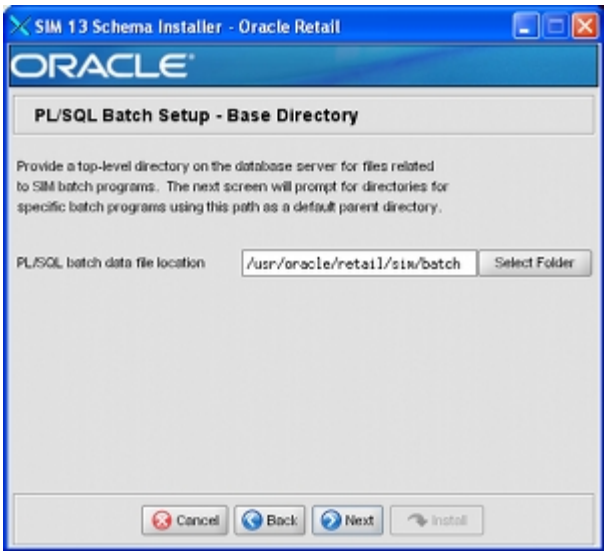
| | |
|---|---|
| **Field Title** | SIM Schema Owner |
| **Field Description** | The pre-existing database user for this installation |
| **Destination** | sim_dba.sql, dataseeding.cfg |
| **Example** | myschema |
| **Notes** | |

| | |
|---|---|
| **Field Title** | Sim Schema Password |
| **Field Description** | The SIM Schema Owner's password |
| **Destination** | |
| **Example** | |
| **Notes** | |

| | |
|---|---|
| **Field Title** | SIM Oracle SID |
| **Field Description** | The name of the database where the SIM schema will be installed |
| **Destination** | |
| **Example** | mydb |
| **Notes** | |

| | |
|---|---|
| **Field Title** | Temporary tablespace name |
| **Field Description** | Temporary tablespace provided to the create_user.sql script at the time that the SIM database user was created. |
| **Destination** | |
| **Example** | TEMP |
| **Notes** | |

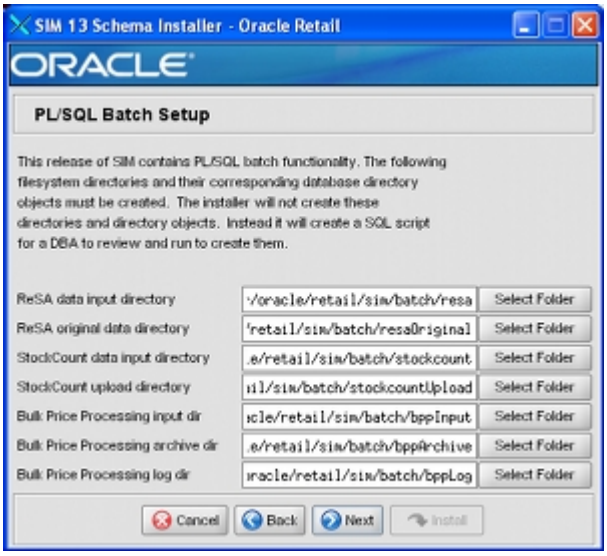
## Screen: PL/SQL Batch Setup – Base Directory




Fields on this screen:

| | |
|---|---|
| **Field Title** | PL/SQL batch data file location |
| **Field Description** | A directory which will be the parent directory for all other PL/SQL batch processing directories |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch |
| **Notes** | |

## Screen: PL/SQL Batch Setup




Fields on this screen:

| | |
|---|---|
| **Field Title** | ReSA data input directory |
| **Field Description** | A filesystem directory and database directory object used for processing ReSA data |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/resa |
| **Notes** | The installer will not create these directories or directory objects. It will produce the sim_dba.sql script, which can be used to create them. |

| | |
|---|---|
| **Field Title** | ReSA original data directory |
| **Field Description** | A filesystem directory and database directory object used for processing ReSA data |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/resaOriginal |
| **Notes** | The installer will not create these directories or directory objects. It will produce the sim_dba.sql script, which can be used to create them. |

| | |
|---|---|
| **Field Title** | StockCount data input directory |
| **Field Description** | A filesystem directory and database directory object used for processing StockCount data |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/stockcount |
| **Notes** | The installer will not create these directories or directory objects. It will produce the sim_dba.sql script, which can be used to create them. |

| | |
|---|---|
| **Field Title** | StockCount upload directory |
| **Field Description** | A filesystem directory and database directory object used for processing StockCount data |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/stockcountUpload |
| **Notes** | The installer will not create these directories or directory objects. It will produce the sim_dba.sql script, which can be used to create them. |

| | |
|---|---|
| **Field Title** | Bulk Price Processing input dir |
| **Field Description** | A filesystem directory and database directory object used for placing the input price change (Promotion, Clearance and Regular Price Change) files. The SQL procedure, reads the file from this input file from this directory. |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/bppInput |
| **Notes** | . |

| | |
|---|---|
| **Field Title** | Bulk Price Processing archive dir |
| **Field Description** | A filesystem directory and database directory object used to place the price change (Promotion, Clearance and Regular Price Change) files which are sucessfully processed by bulk price processing procedure |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/bppArchive |
| **Notes** | . |

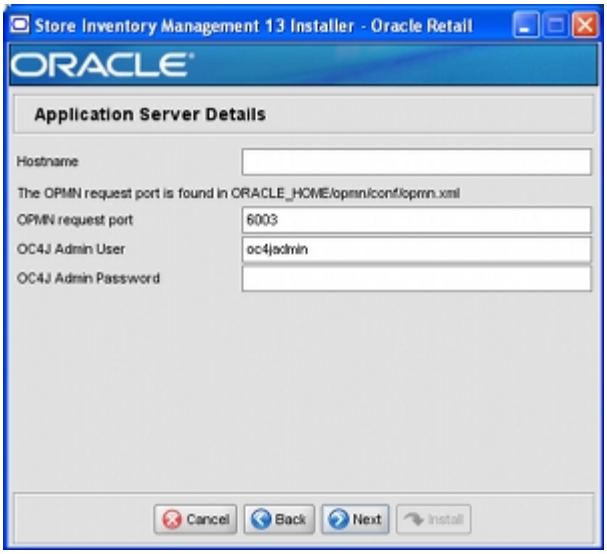
| | |
|---|---|
| **Field Title** | Bulk Price Processing log dir |
| **Field Description** | A filesystem directory and database directory object used to place logs of bulk price processing execution details |
| **Destination** | sim_dba.sql |
| **Example** | /usr/oracle/retail/sim/batch/bppLog |
| **Notes** | . |

# C

# Appendix: SIM Application Installer Screens

You 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<br><port local="6100" remote="6200" request="6003"/> |
| **Destination** | |
| **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 |
| **Field Description** | |
| **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 |
| **Example** | |
| **Notes** | Used by installer scripts to deploy EAR and WAR files |

## Screen: Application Deployment Details


Fields on this screen:

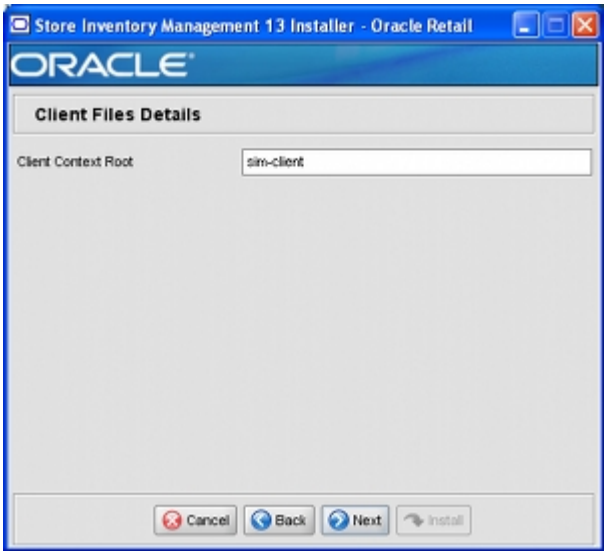
| | |
|---|---|
| **Field Title** | OC4J Group Name |
| **Field Description** | Name of the OC4J group that was created for this SIM application. The OC4J instance given in the OC4J Instance Name field should be a member of this group.<br>The installer will deploy the SIM application to all OC4J instances 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. |
| **Destination** | |
| **Example** | sim_group |
| **Notes** | |

| | |
|---|---|
| **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 |
| **Notes** | |

| | |
|---|---|
| **Field Title** | Application Deployment Name |
| **Field Description** | The name that will be used by the application server to identify the SIM application |
| **Destination** | |
| **Example** | sim13 |
| **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 deploy the sim-client.ear file. |
| **Destination** | |
| **Example** | sim-client |
| **Notes** | |

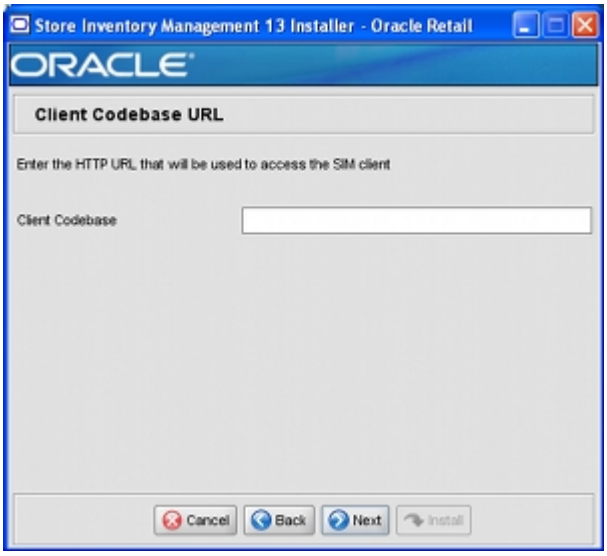
Screen: Client Files Details




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 |

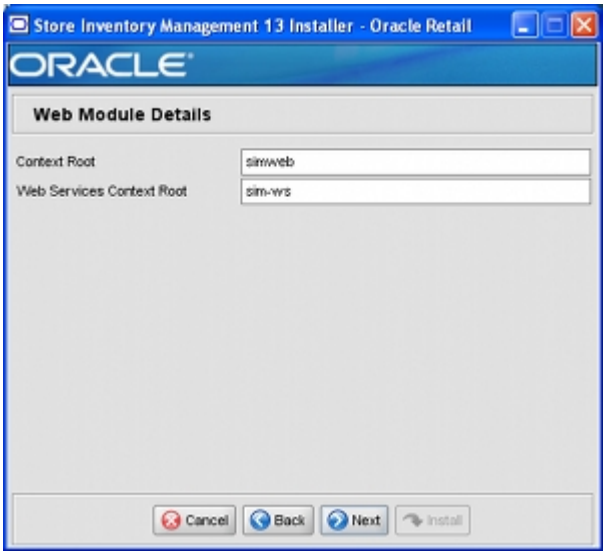
### Screen: Client Codebase URL




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

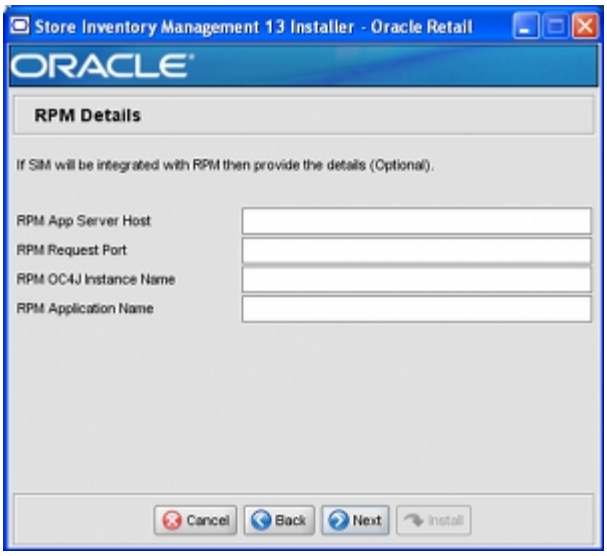



Fields on this screen:

| | |
|---|---|
| **Field Title** | Context Root |
| **Field Description** | The context root for sim.war |
| **Destination** | application.xml |
| **Example** | simweb |
| **Notes** | |

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

## Screen: RPM Details




Fields on this screen:

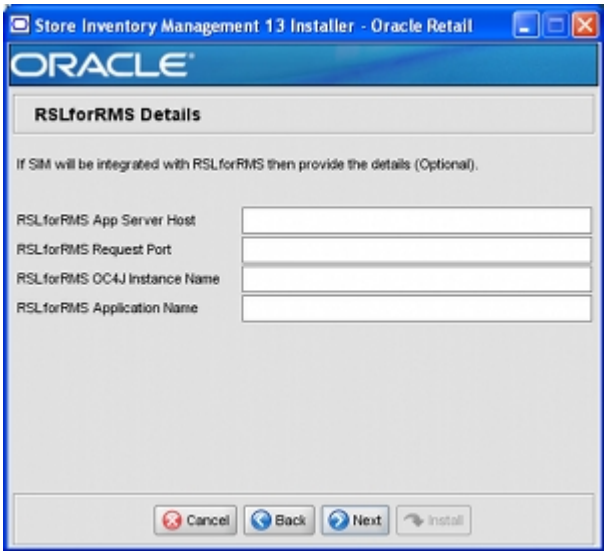
| | |
|---|---|
| **Field Title** | RPM App Server Host |
| **Field Description** | The name of the application server host where the RPM application is installed |
| **Destination** | jndi_providers.xml |
| **Example** | myhost.us.oracle.com |
| **Notes** | Used only if integrating SIM with RPM |

| | |
|---|---|
| **Field Title** | RPM Request Port |
| **Field Description** | The OPMN request port for the application server where RPM is intalled.<br>The OPMN request port is found in $ORACLE_HOME/opmn/conf/opmn.xml<br><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** | rpm13 |
| **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_ribclient.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_ribclient.xml |
| **Example** | 6003 |
| **Notes** | Used only if integrating SIM with RIBforSIM |

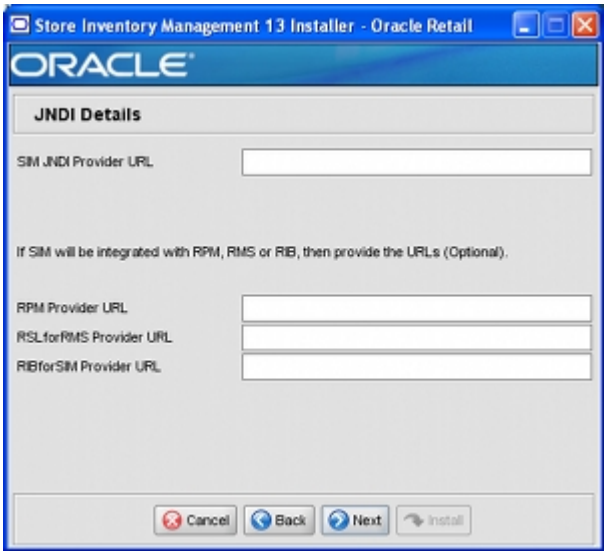
| | |
|---|---|
| **Field Title** | RIBforSIM OC4J Instance Name |
| **Field Description** | The name of the OC4J instance where the RIBforSIM application is installed |
| **Destination** | jndi_providers_ribclient.xml |
| **Example** | rib-sim-o4cj-instance |
| **Notes** | Used only if integrating SIM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

| | |
|---|---|
| **Field Title** | rib-sim OC4J User |
| **Field Description** | The OC4J Admin User for the OC4J instance where rib-sim is installed. |
| **Destination** | jndi_providers_ribclient.xml |
| **Example** | oc4jadmin |
| **Notes** | Used only if integrating SIM with RIBforSIM |

| | |
|---|---|
| **Field Title** | rib-sim OC4J Password |
| **Field Description** | The password of the OC4J Admin User for the OC4J instance where rib-sim is installed. |
| **Destination** | jndi_providers_ribclient.xml |
| **Example** | |
| **Notes** | Used only if integrating SIM with RIBforSIM |

## Screen: JNDI Details




Fields on this screen:

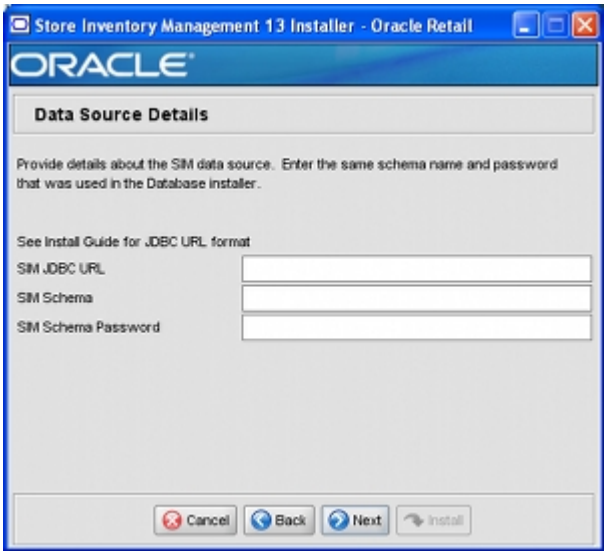
| | |
|---|---|
| **Field Title** | SIM JNDI Provider URL |
| **Field Description** | JNDI provider URL for the SIM application |
| **Destination** | jndi.cfg, JnlpLaunch.properties |
| **Example** | opmn:ormi://myhost.us.oracle.com:6003:sim-oc4j-instance/sim13 |
| **Notes** | Confirm the JNDI provider 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/rpm13 |
| **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



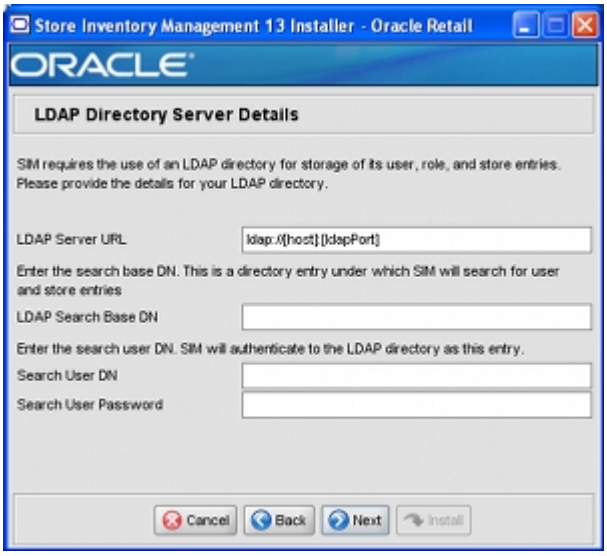

Fields on this screen:

| | |
|---|---|
| **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:1521:mydatabase<br><br>jdbc:oracle:thin:@(DESCRIPTION =(ADDRESS_LIST =(ADDRESS = (PROTOCOL = TCP)(HOST = myhost1)(PORT = 1521))(ADDRESS = (PROTOCOL = TCP)(HOST = myhost2)(PORT = 1521))(LOAD_BALANCE = yes))(CONNECT_DATA =(SERVICE_NAME = mydatabase))) |
| **Notes** | |

| | |
|---|---|
| **Field Title** | SIM Schema |
| **Field Description** | The schema name |
| **Destination** | batch_db.cfg, data-sources.xml |
| **Example** | |
| **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 |
| **Example** | |
| **Notes** | |

## 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 |
| **Notes** | |

| | |
|---|---|
| **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 |
| **Notes** | |

| | |
|---|---|
| **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 |
| **Notes** | |

| | |
|---|---|
| **Field Title** | Search User Password |
| **Field Description** | Password for the search user DN. |
| **Destination** | ldap.cfg |
| **Example** | |
| **Notes** | |

## Screen: Wireless Server Details


Fields on this screen:

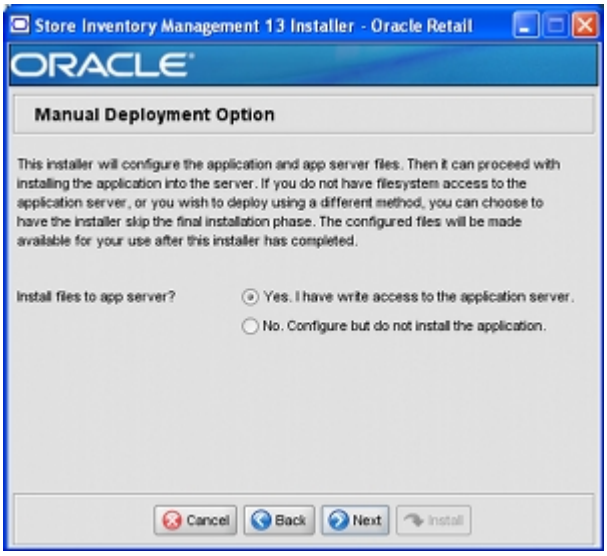
| | |
|---|---|
| **Field Title** | SIM Wireless Server Port |
| **Field Description** | Choose an available port that the Wavelink server will use to listen for incoming messages from wireless devices |
| **Destination** | wireless_services.cfg, wavelink-startup.sh |
| **Example** | 40002 |
| **Notes**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

Screen: Enable SSO in SIM


Fields on this screen:

| | |
|---|---|
| **Field Title** | Enable Single Sign-On in SIM? |
| **Field Description** | Configures SIM to enable/disable SSO |
| **Destination** | JnlpLaunch.properties |
| **Example** | |
| **Notes** | |

## 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. |
| **Destination** | |
| **Example** | |
| **Notes** | If you choose "No", you will need to perform manual steps to complete the installation. |

# D

# 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

# E

# 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

Format for RAC database:

> **Example:** jdbc:oracle:thin:@(DESCRIPTION =(ADDRESS_LIST =(ADDRESS = (PROTOCOL = TCP)(HOST = myhost1)(PORT = 1521))(ADDRESS = (PROTOCOL = TCP)(HOST = myhost2)(PORT = 1521))(LOAD_BALANCE = yes))(CONNECT_DATA =(SERVICE_NAME = mydatabase)))

## 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/sim13

**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

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

# F

# 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 ear file

**Symptom**:

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

07/12/19 10:53:17 Notification ==>Error while unpacking sim13.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.

## A Second Login Screen Appears After Single Sign-On Login

If you are using Oracle Single Sign-On, you should not need to enter a SIM username and password once SIM is launched. If the SIM login screen pops up, it means something went wrong with the SSO login. This could be caused by any of the following problems:

- There is no SIM user in LDAP for the SSO username you're using
- Permissions are not set up correctly for the SSO user in SIM
- SSO is configured wrong on the server
- SSO timed out (This can happen especially the first time you launch SIM. Try launching SIM again.)

**Symptom**:

A second login screen appears after you have already logged in to Single Sign-On.

**Solution:**

See the SIM *Implementation Guide* for more information on setting up SIM users and using LDAP and SSO with SIM.

# G

# 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 some, but not all, of the applications the order is still valid less the applications not being installed.

1. Oracle Retail Merchandising System (RMS), Oracle Retail Trade Management (RTM), Oracle Retail Sales Audit (ReSA)
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 Allocation
7. Oracle Retail Invoice Matching (ReIM)
8. Oracle Retail Price Management (RPM)

   **Note:** During installation of RPM, you are asked for the RIBforRPM provider URL. Since RIB is installed after RPM, make a note of the URL you enter. If you need to change the RIBforRPM provider URL after you install RIB, you can do so by editing the jndi_provider.xml file.

9. Oracle Retail Central Office (ORCO)
10. Oracle Retail Back Office (ORBO)
11. Oracle Retail Store Inventory Management (SIM)

    **Note:** During installation of SIM, you are asked for the AIP provider URL. Since AIP is installed after SIM, make a note of the URL you enter. If you need to change the AIP provider URL after you install AIP, you can do so by editing the jndi_providers_ribclient.xml file.

12. Oracle Retail Integration Bus (RIB)
13. Oracle Retail Point-of-Service (ORPOS)
14. Oracle Retail Analytics Applications
15. Oracle Retail Advanced Inventory Planning (AIP)
16. Oracle Retail Predictive Application Server (RPAS)
17. Oracle Retail Data Warehouse (RDW)
18. Oracle Retail Workspace (ORW)