Oracle® Retail Data Warehouse
Middle Tier Installation Guide
Release 13.0

April 2008

ORACLE®

Oracle® Retail Data Warehouse Middle Tier Installation Guide, Release 13.0

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 **WebLogic™** developed and licensed by BEA Systems, Inc. of San Jose, California, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

(x) 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 Data Warehouse 13.0 is deployed using Oracle Business Intelligence Enterprise Edition (Oracle BI) 10.1.3.3 as the front-end tool. This document describes installation steps for Oracle BI 10.1.3.3 products for a typical client configuration.

We highly recommend that you follow the steps in the order described within the chapters of this document.

RDW recommends the following installation locations:

- BI Server and Presentation Server should be installed on its own dedicated server machine.
- Oracle BI Client Tools should be installed on machines of all users who require the Administration functionality not available through Oracle BI Web interface.
- The rest of the products can be installed in varying combinations depending upon who uses them and on what machines.

Audience

This document is intended for Oracle BI administrators responsible for installing Oracle Retail Data Warehouse.

Related Documents

For more information, see the following documents in the Oracle Retail Data Warehouse Release 13.0 documentation set:

- Oracle Retail Data Warehouse Data Model
- Oracle Retail Data Warehouse Database Installation Guide
- Oracle Retail Data Warehouse Operations Guide
- Oracle Retail Data Warehouse Release Notes
- Oracle Retail Data Warehouse User Guide

Also see the documentation library for Oracle Business Intelligence Enterprise Edition at this URL:

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

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
System Requirements and Database and ODBC Configurations

Oracle Retail Data Warehouse (RDW) relies on Oracle Retail Business Intelligence Suite Enterprise Edition (Oracle BI). The system requirements for RDW are the same as those of Oracle BI. Refer to the "Platform Requirements for Oracle Business Intelligence" chapter of the *System Requirements and Supported Platforms for Oracle® Business Intelligence Suite Enterprise Edition* guide.

2

Oracle BI Infrastructure Installation and Configuration

This chapter provides general installation instructions and recommended configuration considerations for Oracle Business Intelligence Enterprise Edition (Oracle BI).

Install Oracle Business Intelligence Enterprise Edition

For step-by-step instructions on how to install Oracle BI, see the "Installing Oracle BI EE Infrastructure" chapter of the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

> **Note:** You must complete all the steps described in the RDW Database Installation Guide before performing the following steps.

Install RDW 13.0 Repository, Catalog and Set up Database Connection

Installing the Repository

1. Stop Oracle BI Server service.
2. Copy the Repository file (RDW13.rpd) to the <INSTALL_DIRECTORY>/OracleBI/server/Repository/ directory from the staging directory "<mmhome>/metadata".
3. Update the configuration file located at <INSTALL_DIRECTORY>/Oracle BI/server/Config/NQSConfig.INI. Add a new line under the [REPOSITORY] section. For example:

   ```
   [ REPOSITORY ]
   Star   =   RDW13.rpd, DEFAULT;
   ```
4. Other default repositories should be commented out in the NQSConfig.INI. For example:

   ```
        [ REPOSITORY ]
   Star   =   RDW13.rpd, DEFAULT;
   #Star   =   Paint.rpd, DEFAULT;
   ```
5. Save and close NQSConfig.INI.
6. Start Oracle BI Server service.

Set Up the Database Connection

Refer to the "Configuring Data Sources for Oracle Business Intelligence" chapter of the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for additional details about the database connection.

> **Note:** The screen shots in the steps below are samples only. The screens that appear depend on the set up of the client's system.

1. Open RDW13.rpd using the Oracle BI Server Administration tool.

2. In the Physical Layer, open the Connection Pool (RDW13) and specify the database name, username, and password and click OK.
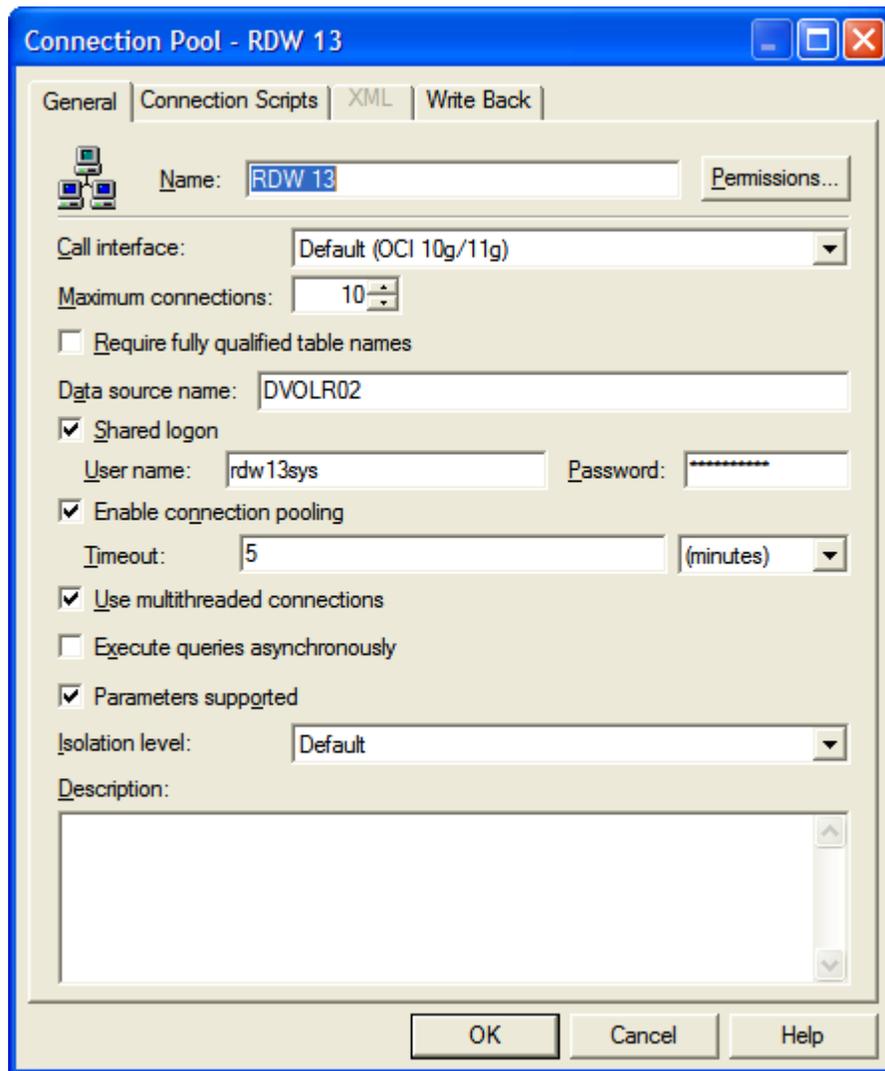
Connection Pool Editor

3. Save the RDW13.rpd file.

4. Test the Database connection by right clicking on any of the tables in the Physical Layer and select View Data.

Successful Database Connection Confirmation

Install Catalog

1. Stop Oracle BI Presentation Server service.

2. Copy the catalog folder (RDW13) to <INSTALL_DIRECTORY>/OracleBIData/web/catalog from "<mmhome>/metadata".

3. Update the instanceconfig.xml file to point to RDW13 catalog. Instanceconfig.xml is found at <INSTALL_DIRECTORY>/OracleBIData/web/config. Change the catalog path with in the file to point to RDW13. For example:

   <CatalogPath>C:/OracleBIData/web/catalog/RDW13</CatalogPath>

4. Save and close instanceconfig.xml.

5. Start Oracle BI Presentation Server service.

6. OC4J service should be running before proceeding to step 7.

7. Test RDW 13 Installation by opening the browser with the URL as http://<hostname>:9704/analytics.

   > **Note:** Refer to the "Making Oracle BI Presentation Services Configuration Changes" chapter of the *Oracle Business Intelligence Presentation Services Administration Guide* for additional information.

Internationalization setup

In order to access all the languages supported by RDW 13.0, copy the following folders from "<mmhome>/metadata" to <INSTALL_DIRECTORY>/OracleBIData/web/res directory:

- L_de
- L_es
- L_fr
- L_it
- L_ja
- L_ko
- L_pt-br
- L_ru
- L_zh
- L_zh-tw

Each of these folders has language-specific translated strings in the form of xml files. For example, the L_de folder has two xml files; DashboardPromptscaptions_de.xml and RDWPackagedreportscaptions_de.xml, which have all the translated catalog strings in German.

After copying, restart the presentation server service. This ensures that the entire RDW specific catalog strings for example report names, dashboard names and dashboard prompts names can be viewed in any of the RDW 13.0 supported languages. Translations of the repository objects are handled through LOCALIZATION_DM table which is part of the RDW 13.0 data model.

> **Note:** The following is a listing of the RDW 13 provided translated languages:
>
> de: German
> es: Spanish
> fr: French
> it: Italian
> ja: Japanese
> ko: Korean
> pt-br: Brazilian Portuguese
> ru: Russian
> zh: Chinese Simplified
> zh-tw: Chinese Traditional

Manage Users and Security

For information regarding users, user groups, security roles, permissions and privileges refer to the "Security in Oracle BI" chapter of the *Oracle Business Intelligence Server Administration Guide*.

By default RDW 13.0 is packaged with only one user called administrator (by default there is no password).

Other Important Notes

- RDW 13.0 supports retail 4-5-4, combined 4-5-4/Gregorian, or 13 period calendar. You can choose which calendar to use during database installation. The default calendar is retail 4-5-4. To implement either combined 4-5-4/Gregorian or 13 period, see the Appendix: Time of this document.

- In RDW 13.0, Differentiator Types are hard-coded within the attribute names. See the Appendix: Item Differentiators for detailed information regarding adding or modifying of these attributes.

- While making any changes to the repository, RDW recommends opening the .rpd file offline and performing a Global Consistency Check before saving the changes. The Global Consistency Check should not show any errors other then the two mentioned below.

- The RDW 13.0 repository gives two errors in the Business Layer when a Global Consistency Check is done. These errors do not affect RDW functionality in any way. These errors occur due to a limitation in Oracle BI 10.1.3.3. When a dimension is created as a time dimension then the tables within the dimension cannot be joined to any other tables directly. The two errors appear below.

  BUSINESS MODEL RDW:

  [38091] Physical table 'DIM_TIME_WK' joins to non-fact table 'DIM_ORG_LOC_WEEK_MTX' that is outside of its time dimension table source 'DIM_TIME_WK'.

  [38091] Physical table 'DIM_TIME_WK' joins to non-fact table 'DIM_ORG_LOC_WEEK_MTX' that is outside of its time dimension table source 'DIM_TIME_WK'.

3

Oracle BI Client Tools Installation and Configuration

This chapter provides Oracle BI Client Tools general installation instructions and recommended configuration for users that are not administrators, but require the Client Tools functionality not available in Oracle BI Presentation Services. Oracle BI Client tools are required for the power user like the developers for customizing any repository objects.

Install Oracle BI Client Tools

For step-by-step instructions on how to install Oracle BI, see the "Installing Oracle BI EE Infrastructure" chapter of the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

When asked what products to install, select the Oracle Business Intelligence Client Tools option.

Opening RDW 13.0 Repository

1. Open the Oracle BI Administration Tool.
2. Click on File→Open→Offline.
3. In the dialog box, select the RDW 13.0 rpd file which was installed in the "Install Repository" section of this guide.

It is highly recommended that you create a MUD (Multi User Development environment) when multiple developers are working with the same repository in Administration Tool. Refer to "Setting up and Using the Oracle BI Multiuser Development Environment" in the "Completing Setup and Managing Oracle BI Repository Files" chapter of the *Oracle Business Intelligence Server Administration Guide*.

A

Appendix: Time

Time Calendar (4-5-4)

RDW provides support for the retail 4-5-4 calendar. The fiscal 4-5-4 calendar is the calendar supported by RMS and other Oracle Retail applications and is populated in RDW via an extract from RMS. The 4-5-4 calendar is the default calendar used when viewing the time dimension via the Oracle BI middle tier layer of RDW.

> **Note:** Even if 4-5-4 is the default calendar for RDW, you can still see the Gregorian Time attributes and transformations from within Oracle BI. However, you can only utilize these objects if you have opted for Gregorian calendar during database installation.

Time Calendar (4-5-4/Gregorian)

RDW provides support for the combined 4-5-4 calendar/Gregorian calendar. If a client chooses to use the combined 4-5-4/Gregorian calendar, the client must execute the batch program that generates the text files needed to populate the Gregorian time dimension.

> **Note:** In order to update RDW to report in combined 4-5-4/Gregorian calendar, please see the section, "Create Time in RDW" in the *RDW Database Installation Guide* for specific instructions. Nothing needs to be modified within Oracle BI.

Time Calendar (13 period)

The 13-Period calendar can also be used, but RMS does not support it. If a client chooses to use the 13-Period calendar, the client can either provide a flat file with its 13-period time, or utilize a sample 13-period time flat file, and then ETL scripts populate the time dimension with this file during RDW installation. Within the middle-tier application, all references to Month must be manually updated to Period, to comply with the 13 period nomenclature.

> **Note:** In order to update RDW to report in 13 period time, follow these steps:

1. Update the time dimension tables. (For complete instructions, reference the RDW Database Installation Guide.)

2. Within the repository, delete the attribute Half Year, and re-name the objects referencing month.

   - Highlighting the object in the Business Layer and pressing the delete key deletes the attribute Half Year. All relationships and hierarchies are automatically updated. The table TIME_HALF_DM and its Aliases can also be deleted the same way in the Physical layer. Global Consistency check is recommended after the change.

   - Each repository object referencing Month can be re-named by selecting the object and pressing F2. After all the changes, Global Consistency check is recommended before saving the repository.

Appendix: Item Differentiators

Diff Attributes

As part of the RDW install, the first five diff types are hard-coded in both the DIFF_TYPE_DM look-up table and in the names of the attributes that point to them. This matches the initial set of pre-defined diff types that are provided within the Oracle Retail Merchandising System (RMS). This leaves the remaining diff types, and their corresponding attributes, to be defined at the client's discretion at a later date.

The following is a list of differentiators:

DIFF_TYPE_KEY	Attribute Name
1	Size
2	Color
3	Flavor
4	Scent
5	Pattern

Creating New Diff Types

With each additional diff type that is added to the merchandising system, a new attribute, within RDW, should be created to suitably match the name of the new diff type. Perform the following procedure to create new Diff Type attributes:

1. Open the RDW13.rpd.

2. Create a new logical column in the Dim 0 Product Differentiator Logical table of the RDW.rpd business layer. The first time, this will be Diff 6 desc.

3. Once completed, save the repository.

Modifying Diff Types

Functionality exists within the merchandising system to modify existing diff types. This can be done by either changing the description of the diff type (that is, change the diff type "Size" to "Fashion Size") or by deleting the diff type altogether.

When the description is changed within the merchandising system, the batch process updates the description column within the PROD_DIFF_TYPE_DM table to match this new description. The attribute name within the repository must be updated manually by changing the attribute name.

When deleting a diff type within the merchandising system, the batch process updates the description column within the PROD_DIFF_TYPE_DM table to 'NULL'. The attribute name within the repository must be updated manually to the generic name corresponding to the DIFF_TYPE_KEY, by renaming. That is, if Color was deleted, the system updates the name of the Color attribute to DIFF 2.

Appendix: Configuring RDW and Oracle BI for Single Sign-on

Overview

What is Single Sign-On?

Single Sign-On (SSO) is a term for the ability to sign onto multiple web applications via a single user ID/Password. There are many implementations of SSO – Oracle currently provides three different implementations: Oracle Single Sign-On (SSO), Java SSO (with the 10.1.3.1 release of OC4J) and Oracle Access Manager (provides more comprehensive user access capabilities).

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

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

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

What Do I Need for Oracle Single Sign-On?

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

- An Oracle Internet Directory (OID) LDAP server, used to store user, role, security, and other information. OID uses an Oracle database as the back-end storage of this information.

- An Oracle Single Sign-On servlet, used to authenticate the user and create the SSO session cookie. This servlet is deployed within the infrastructure Oracle Application Server (OAS).

- The Delegated Administration Services (DAS) application, used to administer users and group information. This information may also be loaded or modified via standard LDAP Data Interchange Format (LDIF) scripts.

- Additional administrative scripts for configuring the SSO system and registering HTTP servers.

Additional OAS servers will be needed to deploy the business applications leveraging the SSO technology.

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

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

Oracle Single Sign-on Terms and Definitions

Authentication

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

Dynamically Protected URLs

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

Identity Management Infrastructure

The Identity Management Infrastructure is the collection of product and services which provide Oracle Single Sign-on functionality. This includes the Oracle Internet Directory, an Oracle HTTP server, and the Oracle Single Sign-On services. The Oracle Application Server deployed with these components is typically referred as the "Infrastructure" instance.

MOD_OSSO

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

Oracle Internet Directory

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

Partner Application

A partner application is an application that delegates authentication to the Oracle Identity Management Infrastructure. One such partner application is the Oracle HTTP Server (OHS) supplied with the Oracle Application Server. OHS uses the MOD_OSSO module to configure this functionality.

All partner applications must be registered with the Oracle Single Sign-On server. An output product of this registration is a configuration file the partner application uses to verify a user has been previously authenticated.

Realm

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

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

Statically Protected URLs

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

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

What Single Sign-On is not

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

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

How Oracle Single Sign-On Works

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

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

Statically Protected URLs

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

1. The Oracle HTTP server recognizes the user has not been authenticated and redirects the browser to the Oracle Single Sign-On servlet.
2. The SSO servlet determines the user must authenticate, and displays the SSO login page.
3. The user must sign in via a valid user ID and password. If the SSO servlet has been configured to support multiple Realms, a valid realm must also be entered. The user ID, password, and realm information is validated against the Oracle Internet Directory LDAP server.
4. The SSO servlet creates and sends the user's browser an SSO session cookie. This cookie is never persisted to disk and is specific only to the current browser session. This cookie contains the user's authenticated identity. It does NOT contain the user's password.
5. The SSO servlet redirects the user back to the Oracle HTTP Server, along with SSO specific information.
6. The Oracle HTTP Server decodes the SSO information, stores it with the user's session, and allows the user access to the original URL.

Dynamically Protected URLs

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

1. The Oracle HTTP server recognizes the user has not been authenticated, but allows the user to access the URL.

2. The application determines the user must be authenticated and sends the Oracle HTTP server a specific status to begin the authentication process.

3. The Oracle HTTP Server redirects the user's browser session to the SSO Servlet.

4. The SSO servlet determines the user must authenticate, and displays the SSO login page.

5. The user must sign in via a valid user ID and password. If the SSO servlet has been configured to support multiple Realms, a valid realm must also be entered. The user ID, password, and realm information is validated against the Oracle Internet Directory LDAP server.

6. The SSO servlet creates and sends the user's browser an SSO session cookie. This cookie is never persisted to disk and is specific only to the current browser session. This cookie contains the user's authenticated identity. It does NOT contain the user's password.

7. The SSO servlet redirects the user back to the Oracle HTTP Server, along with SSO specific information.

8. The Oracle HTTP Server decodes the SSO information, stores it with the user's session, and allows the user access to the original URL.

Single Sign-on Topology
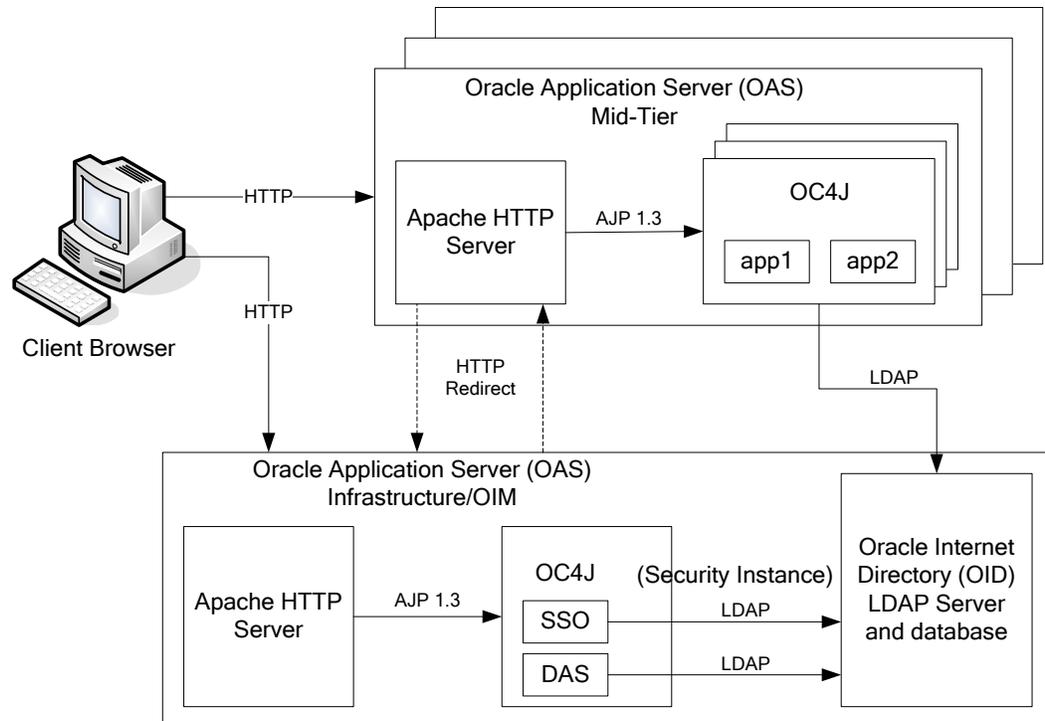
Installation Overview

Installing Oracle Single Sign-On consists of installing the following components:

1. Installing the Oracle Internet Directory (OID) LDAP server and the Infrastructure Oracle Application Server (OAS). These are typically performed using a single session of the Oracle Universal Installer and are performed at the same time. OID requires an Oracle relational database and if one is not available, the installer will also install this as well.

   The Infrastructure OAS includes the Delegated Administration Services (DAS) application as well as the SSO servlet. The DAS application can be used for user and realm management within OID.

2. Installing additional OAS 10.1.2 midtier instances for the Oracle Retail applications, such as RMS, that are based on Oracle Forms technologies. These instances must be registered with the Infrastructure OAS installed in step 1).

3. Installing additional application servers to deploy other Oracle Retail applications and performing application specific initialization and deployment activities.

Infrastructure Installation and Configuration

The Infrastructure installation for SSO is dependent on the environment and requirements for its use. Deploying an Infrastructure OAS to be used in a test environment does not have the same availability requirements as for a production environment. Similarly, the Oracle Internet Directory (OID) LDAP server can be deployed in a variety of different configurations. See the *Oracle Application Server Installation Guide* and the *Oracle Internet Directory Installation Guide* for more details.

OID User Data

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

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

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

OID with Multiple Realms

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

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

User Management

User Management consists of displaying, creating, updating or removing user information. There are two basic methods of performing user management: LDIF scripts and the Delegate Administration Services (DAS) application.

OID DAS

The DAS application is a web based application designed for both administrators and users. A user may update their password, change their telephone number of record, or modify other user information. Users may search for other users based on partial strings of the user's name or ID. An administrator may create new users, unlock passwords, or delete users.

The DAS application is fully customizable. Administrators may define what user attributes are required, optional or even prompted for when a new user is created.

Furthermore, the DAS application is secure. Administrators may also what user attributes are displayed to other users. Administration is based on permission grants, so different users may have different capabilities for user management based on their roles within their organization.

LDIF Scripts

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

User Data Synchronization

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

Enabling Single Sign-on for Oracle Business Intelligence Enterprise Edition

Single sign-on for RDW is enabled in Oracle Business Intelligence Enterprise Edition (Oracle BI). See the *Oracle Business Intelligence Deployment Guide* for information.

> **Note:** The Oracle Business Intelligence Deployment Guide has you use the "mail" attribute to uniquely identify users. When this is done, a users' SSO username must match their e-mail address or a "Not Logged In" page appears when using SSO with Oracle BI. Oracle Retail strongly suggests Oracle BI is configured to use the "uid" attribute to uniquely identify users.