Oracle® Application Server

Administrator's Guide

10*g* Release 2 (10.1.2)

B13995-08

September 2006

ORACLE®

Oracle Application Server Administrator's Guide, 10*g* Release 2 (10.1.2)

B13995-08

Contributing Authors: Helen Grembowicz, , Kevin Hwang, Peter LaQuerre, Mary Beth Roeser, Harry Schaefer, Deborah Steiner

Contributors: Megan Ginter, Pavana Jain, Kai Li, Thomas Van Raalte, Andrew Salt, Pavi Sandhu

Contents

2 Introduction to Administration Tools

3 Starting and Stopping

5 Managing Log Files

6 Managing an OracleAS Metadata Repository

Part III Advanced Administration

7 Reconfiguring Application Server Instances

8 Changing Network Configurations

9 Changing Infrastructure Services

10 Cloning Application Server Middle-Tier Instances

15 Managing Wallets and Certificates

16 Enabling SSL in the Infrastructure

20 Oracle Application Server Backup and Recovery Tool

21 Backup Strategy and Procedures

22 Recovery Strategies and Procedures

E Metadata Repository Schemas

F printlogs Tool Syntax and Usage

G Examples of Administrative Changes

H Supplementary Procedures for Configuring LDAP-Based Replicas

I Viewing Oracle Application Server Release Numbers

J Troubleshooting Oracle Application Server

List of Figures

List of Tables

Preface

This guide describes how to manage Oracle Application Server, including how to start and stop the Oracle Application Server, how to reconfigure components, and how to backup and recovery Oracle Application Server.

Audience

This guide is intended for administrators of Oracle Application Server.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

http://www.oracle.com/accessibility/

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documentation

For more information, see these Oracle resources:

- Oracle Application Server Documentation Library
- Oracle Application Server Platform-Specific Documentation

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN):

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

Conventions

The following text conventions are used in this document:

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

What's New in Oracle Application Server Administration?

This preface introduces the new administrative features of Oracle Application Server 10*g* Release 2 (10.1.2.0.2). This information is mostly useful to users who have managed previous releases of Oracle Application Server, including Oracle Application Server 10*g* Release 1 (9.0.4) and 10*g* Release 2 (10.1.2.0.0).

New Features for 10*g* Release 2 (10.1.2.0.2)

The new administrative features of Oracle Application Server 10*g* Release 2 (10.1.2.0.2) include:

New Application Server Control Feature for Backup and Recovery

You can use the Application Server Control Console to manage backup and recovery of an application server instance. You can perform all of the backup and recovery functions using the Application Server Control Console, instead of entering commands on the command line.

See:

- Section 21.2.4, "Performing an Instance Backup of Oracle Application Server Using Application Server Control Console"
- Section 22.2.1, "Using Application Server Control Console to Recover an Oracle Application Server Instance"

New Backup and Recovery Features

New backup and recovery features include:

- If your Oracle Application Server installation has a Metadata Repository database with a registered Portal middle-tier, you can run the Portal Schema Validation/Cleanup Utility (SVU) during a backup of the Metadata Repository database. Running SVU provides a way to ensure the integrity of your data whenever you perform a backup of your database.

  See: Section 20.4, "Running the Portal Validation/Cleanup Utility"

- The Backup and Recovery Tool provides an automatic recovery service to a new host when a loss of host occurs.

  See: Section 21.3, "Recovering a Loss of Host Automatically"

New Port Numbers for Some Components

To guard against conflicts with Windows ephemeral port number assignments, default port numbers and ranges have been changed for Application Server Control Console, OracleAS Web Cache, OracleAS Integration B2B, OC4J, Oracle Internet Directory, and Oracle Application Server Certificate Authority.

See: Appendix D, "Oracle Application Server Port Numbers"

New SSL Configuration Tool

The SSL Configuration Tool simplifies and automates SSL configuration for common Oracle Application Server configurations. It is designed to automate many of the manual steps currently required for configuring SSL.

See: Chapter 14, "Using the SSL Configuration Tool"

Support to Enable and Disable Oracle Internet Directory Anonymous Binds

In this release, you can disable anonymous binds in Oracle Internet Directory. In past releases, anonymous binds were enabled, but you could not disable them. Now, anonymous binds are enabled by default, but you can disable them for your runtime environment. Note that you must enable anonymous binds for most configuration changes.

See: Section 7.6, "Disabling and Enabling Anonymous Binds"

OracleAS Metadata Repository Uses Oracle Database 10*g*

In this release, the installer creates an Oracle Database 10*g* (10.1.0.4.2) for the OracleAS Metadata Repository.

Support for Copying a Production Environment to a Test Environment

Oracle Application Server provides support for changing from a test environment to a production environment. You can develop and test applications in a test environment, and then eventually roll out the test applications and, optionally, test data to your production environment. You can also use this approach for testing and rolling out upgrades.

See: Chapter 11, "Staging a Test Environment from a Production Environment"

New Features for 10*g* Release 2 (10.1.2.0.0)

The new administrative features of Oracle Application Server 10*g* Release 2 (10.1.2.0.0) include:

Quick Administration Guide

The Oracle Application Server documentation now includes a Quick Administration guide, which contains quick reference material for common administration tasks.

See: *Oracle Application Server Quick Administration Guide*.

New Library Path Variables

The value of the LD_LIBRARY_PATH environment variable has changed, and a new environment variable, LD_LIBRARY_PATH_64 has been introduced with this release.

See: Section 1.1, "Task 1: Set Up Environment Variables"

New Application Server Control Features

New features in Application Server Control include:

- Topology view of the Application Server environment

  A visual representation of the application server environment is essential for administrators to understand component relationships, such as where applications are deployed across the OracleAS Farm or OracleAS Cluster. The Application Server Control Console satisfies this requirement by providing Topology Viewer. Topology Viewer provides a graphical, real-time view of application server processes managed by Oracle Process Manager and Notification Server (OPMN). From Topology Viewer, you can perform various tasks such as:

  - Viewing the status of the farm, cluster and member components
  - Starting, stopping or restarting processes
  - Monitoring performance across the application server environment
  - Drilling down to component home pages for details

  See Section 2.4.1, "Reviewing the Application Server Component Topology".

- Support for viewing and configuring a file-based repository

  Using the Application Server Control, you can view and configure the file-based repository that is used by an Application Server instance. A wizard guides you through the steps of configuring the instance with either an existing or new file-based farm repository. See Section 7.5.5 and Section 7.5.6.

- Support for viewing and changing Infrastructure Services for Identity Management

  Using the Application Server Control, you can view, configure, and change the Infrastructure Services (Identity Management and Metadata Repository) used by Application Server components. A wizard guides you through the steps of changing the Metadata Repository configured for Identity Management. See Chapter 9.

- Views of all performance metrics and metric details

  For each component, you can view a list of performance metrics that Application Server Control is monitoring. For each performance metric being monitored, you can drill down to view a brief history of its performance. See Section 2.4.4.

- Complete integration of OracleAS Web Cache administration into Application Server Control

  Now, you can fully manage and administer OracleAS Web Cache instances through Application Server Control, instead of using a separate tool (OracleAS Web Cache Manager). The *Oracle Application Server Web Cache Administrator's Guide* describes how to use Application Server Control to manage OracleAS Web Cache.

- Support for starting and stopping Application Server Control from the Windows Start menu

  Now, you can start and stop the Application Server Control from the Windows Start menu. See Section A.1.

- Support for querying from a database log repository

  Now, the log repository feature of Application Server Control supports using a database repository, not only a file-based repository, as the log repository. However, you cannot use Application Server Control to create the database log repository; you must manually create it. See Section 5.1.2.

- Ability to change the port values of Application Server Control framework components

  Now, you can change port numbers of Enterprise Manager components, such as Application Server Control and Oracle Management Agent, by using the `emctl` command-line utility. See Section 4.3.1.

Automation of Configuration Steps to Modify Oracle HTTP Server Listen Port

You can modify port values for Application Server components. When some component port values are changed, other components can be affected by those changes. In this release, you can run a command-line program to modify the Oracle HTTP Server listen port value. This command-line program performs additional steps that are required because of the port change, such as configuring OracleAS Web Cache to use the new Oracle HTTP Server port.

> **See:** Section 4.3.3, "Changing the Oracle HTTP Server Listen Ports"

OracleAS Metadata Repository Uses Oracle Database 10*g*

In 10g (9.0.4), the installer created an Oracle9i Release 1 (9.0.1.5) database and loaded the OracleAS Metadata Repository into that database. In 10.1.2.0.0, the installer creates an Oracle database 10g (10.1.0.3) for the OracleAS Metadata Repository.

Oracle Enterprise Manager 10*g* Database Control

When you install the 10g (10.1.2) OracleAS Metadata Repository, Oracle Application Server now installs a Oracle Database 10*g*,which includes Oracle Enterprise Manager 10*g* Database Control, which you can use to manage your new Metadata Repository database.

> **See:** Section 2.5, "Managing the OracleAS Metadata Repository Database with Database Control"

Enhancements to Cloning an Installation of Oracle Application Server

Now you can clone a Portal and Wireless installation and a Business Intelligence installation, as well as a J2EE and Web Cache installation. The command-line interface is improved and the cloning scripts automatically perform many of the necessary tasks.

> **See:** Chapter 10, "Cloning Application Server Middle-Tier Instances"

Support for Changing from a Test to a Production Environment

Oracle Application Server provides support for changing from a test environment to a production environment. You can develop and test applications in a test environment, and then eventually roll out the test applications and, optionally, test data to your production environment. You can also use this approach for testing and rolling out upgrades.

> **See:** Chapter 12, "Changing from a Test to a Production Environment"

Consolidated SSL Instructions

This guide contains a new section about enabling SSL across Oracle Application Server.

> **See:** Part IV, "Secure Sockets Layer (SSL)"

Part I

Getting Started

This part contains information for getting started with managing Oracle Application Server.

It contains the following chapters:

1

Getting Started After Installing Oracle Application Server

This chapter contains tasks to help you get started managing Oracle Application Server after installation.

It contains the following topics:

1.1 Task 1: Set Up Environment Variables

When you installed Oracle Application Server, you were logged in to your operating system as a particular user. You should always log in as this user to manage your installation because this user has permission to view and modify the files in your installation's Oracle home.

To use Oracle Application Server, you must set environment variables as shown in the following tables:

Table 1–1 Oracle Application Server Environment Variables for UNIX

Environment Variable	Value
DISPLAY	*hostname*:*display_number*.*screen_number* Beginning with Oracle Application Server 10*g*, very few tools require the DISPLAY variable. Only a few tools, such as `oidadmin`, require it.

Table 1–1 (Cont.) Oracle Application Server Environment Variables for UNIX

Environment Variable	Value

LD_LIBRARY_PATH	On Solaris, make sure the value contains the following directory:
	`$ORACLE_HOME/lib32`
	On Linux, make sure the value contains the following directory:
	`$ORACLE_HOME/lib`
	On HP-UX, make sure the value contains the following directory:
	`$ORACLE_HOME/lib`
	On IBM AIX, make sure this environment variable is not set.

(IBM AIX only)
LIBPATH
	If the calling application is a 32-bit application, make sure the value contains the following directory:
	`$ORACLE_HOME/lib32`
	If the calling application is a 64-bit application, make sure the value contains the following directory:
	`$ORACLE_HOME/lib`

(Solaris only) LD_LIBRARY_PATH_64	Make sure the value contains the following directory: `$ORACLE_HOME/lib`
(HP-UX only) SHLIB_PATH	Make sure the value contains the following directory: `$ORACLE_HOME/lib32`
ORACLE_HOME	Set to the full path of the installation's Oracle home
ORACLE_SID (Infrastructure installations only)	Set to the OracleAS Metadata Repository SID you supplied during installation. The default is `orcl`.

PATH	Make sure the value contains the following directories, which contain basic commands used by all installations:
	`$ORACLE_HOME/bin`
	`$ORACLE_HOME/dcm/bin`
	`$ORACLE_HOME/opmn/bin`
	When you start to work with specific components, you may want to add additional directories to your path, as recommended by the component documentation.

Table 1–2 shows the environment variables for Windows.

Table 1–2 Oracle Application Server Environment Variables for Windows

Environment Variable	Value
ORACLE_HOME	Set to the full path of the installation's Oracle home. The value is automatically set by Oracle Universal Installer.
ORACLE_SID (Infrastructure installations only)	Set to the OracleAS Metadata Repository SID you supplied during installation. The default is `orcl`. The value is automatically set by Oracle Universal Installer.
TEMP	Set to your temp directory, for example, `C:\temp`.
TMP	Set to your temp directory, for example, `C:\temp`.

Best Practices for Multiple Installations on a UNIX Host

If you have multiple installations of Oracle Application Server on a UNIX host, it is very important to completely set your environment when managing a particular installation.

Some Oracle Application Server commands use the ORACLE_HOME environment variable to determine which installation to operate on, and some use the directory location of the command. It is, therefore, not sufficient to simply reset your environment variables or `cd` to a different Oracle home as you move between installations. You must fully change to the new installation as follows:

1. Log in as the user who installed the installation you want to work on.

   On UNIX hosts, you may also use the `su` command to switch to the user, but be sure to use the dash (`-`) option so your environment is set the same as it would have been had you actually logged in as that user.

   ```
   su - user
   ```

2. Set the correct environment variables for the installation, as described in Table 1–1.
3. Execute commands in the Oracle home of the correct installation.

Multiple Installations by the Same User If you installed multiple installations as the same user, make sure that you are in the correct Oracle home and have the correct environment variables set when working on a particular installation. You may want to set up some scripts to make it easy to change from one installation to another.

1.2 Task 2: Use the Oracle Application Server Welcome Page

The Oracle Application Server Welcome Page is a great starting point for managing your application server. It includes the following:

- A link to details about New Features in Oracle Application Server 10*g* Release 2 (10.1.2)
- A link to a Quick Tour that provides a graphical introduction to Oracle Application Server 10*g* Release 2 (10.1.2)
- A link to the Oracle Application Server 10*g* Release 2 (10.1.2) documentation library
- Release Notes for your platform
- A link to the Oracle Enterprise Manager 10*g* Application Server Control Console—a Web-based tool for managing Oracle Application Server
- Demonstrations and code samples for Oracle Application Server components and features

Figure 1–1 shows the Oracle Application Server Welcome Page.

Figure 1–1 Oracle Application Server Welcome Page

Accessing the Welcome Page

You can locate the URL for accessing the Welcome Page on the End of Installation Screen text, which is in the following file:

```
(UNIX) ORACLE_HOME/install/setupinfo.txt
(Windows) ORACLE_HOME\install\setupinfo.txt
```

To view the Welcome Page, connect to it using the HTTP listener port on your installation. For example:

```
http://hostname.domain:port
```

The default port is 7777 on UNIX, 80 on Windows.

Tip If you cannot access the Welcome Page, try the following:

1. Check `setupinfo.txt` and make sure you are using the correct URL (hostname and port number).
2. Try restarting Oracle HTTP Server:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server

   (Windows) ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=HTTP_Server
   (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=HTTP_Server
   ```

3. If you have OracleAS Web Cache configured, try restarting it:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=WebCache
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=WebCache
   ```

```
(Windows) ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=WebCache
(Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=WebCache
```

1.3 Task 3: Check Your Port Numbers

During installation, Oracle Application Server assigned port numbers to various components and services. It is important to check these port numbers for two reasons:

- You need to know these port numbers in order to start managing your application server.
- Oracle Application Server takes several measures to ensure that port number assignments are unique. However, it is possible that a port assignment could conflict with a non-Oracle Application Server process on your host that was not running during the installation. If you determine there is a conflict, stop the non-Oracle Application Server process and continue with the tasks in this chapter. Once you have completed the tasks in this chapter and have verified that your installation is running properly, you can consider changing Oracle Application Server port numbers.

See Also: Chapter 4 for information on changing port numbers

You can find the complete list of port numbers in:

```
(UNIX) ORACLE_HOME/install/portlist.ini
(Windows) ORACLE_HOME\install\portlist.ini
```

Example 1–1 shows a sample of this file from an installation on UNIX.

Example 1–1 A Sample portlist.ini File

```
;OracleAS Components reserve the following ports at install time.
;As a postinstallation step, you can reconfigure a component to use a different
port.
;Those changes will not be visible in this file.

[System]
Host Name = host1.mycompany.com

[Ports]
Oracle HTTP Server port = 7777
Oracle HTTP Server Listen port = 7778
Oracle HTTP Server SSL port = 4443
Oracle HTTP Server Listen (SSL) port = 8250
Oracle HTTP Server Diagnostic port = 7200
Application Server Control RMI port = 1850
Oracle Notification Server Request port = 6003
Oracle Notification Server Local port = 6100
Oracle Notification Server Remote port = 6200
Log Loader port = 44000
Java Object Cache port = 7000
DCM Discovery port = 7101
Application Server Control port = 1156
Enterprise Manager Agent port = 1830
Web Cache HTTP Listen port = 7777
Web Cache HTTP Listen (SSL) port = 8250
Web Cache Administration port = 9400
Web Cache Invalidation port = 9401
Web Cache Statistics port = 9402
```

Note the following about `portlist.ini`:

- You should leave the port numbers as they are until you have completed the tasks in this chapter and confirmed that all of your components are running properly. Then, you can consider changing port numbers. Note that some port numbers cannot be changed, and some require additional steps for updating other components.

  See Also: Chapter 4 for information about port assignments and changing port numbers

- The `portlist.ini` file contains port numbers for components you did not select during installation because Oracle Application Server reserves ports for all components during installation, even those that were not configured. These port numbers will be used if you configure components after installation. See Section 7.2, "Configuring Additional Components After Installation" for information.
- The `portlist.ini` file contains the port numbers that were assigned during installation and is very useful for getting started. However, it is not updated if you modify port numbers after installation. Once you start managing the components, you should use the Application Server Control Console Ports property page for viewing port numbers, because it displays the current port numbers.
- The `portlist.ini` file is not valid after you upgrade Oracle Application Server.

1.4 Task 4: Get Started with Managing Components

This task provides an introduction to managing components and includes instructions for accessing component administration tools, postinstallation notes about components, and pointers to more information. It contains the following topics:

- Getting Started with Oracle Process Manager and Notification Server (OPMN)
- Getting Started with Distributed Configuration Management (DCM)
- Getting Started with Oracle HTTP Server
- Getting Started with Oracle Application Server Containers for J2EE (OC4J)
- Getting Started with OracleAS Web Cache
- Getting Started with OracleAS Portal
- Getting Started with OracleAS Wireless
- Getting Started with OracleBI Discoverer
- Getting Started with OracleAS Forms Services
- Getting Started with OracleAS Reports Services
- Getting Started with OracleAS Personalization
- Getting Started with Oracle Application Server Integration Products
- Getting Started with Identity Management Components

  See Also: Appendix C for a quick reference on how to access components

Many of the following sections refer to specific ports. Review the `portlist.ini` file, at the following location, to find the port number for the specific port:

```
(UNIX) ORACLE_HOME/install/portlist.ini
(Windows) ORACLE_HOME\install\portlist.ini
```

1.4.1 Getting Started with Oracle Process Manager and Notification Server (OPMN)

Oracle Process Manager and Notification Server (OPMN) manages and monitors most Oracle Application Server components. It is installed and configured in every middle-tier and OracleAS Infrastructure installation and is essential for running Oracle Application Server.

OPMN provides the `opmnctl` command. The executable file is located in the following directory:

```
(UNIX) ORACLE_HOME/opmn/bin
(Windows) ORACLE_HOME\opmn\bin
```

To get started with OPMN, use the `opmnctl` command to query the status of the components in your installation:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl status
(Windows) ORACLE_HOME\opmn\bin\opmnctl status
```

Example 1–2 shows sample output from the command. It displays the component name, process type, operating system process ID (PID), and status of each process.

Example 1–2 Sample Output from opmnctl status Command

```
Processes in Instance: mid.myhost.myco.com
------------------+--------------------+--------+---------
ias-component     | process-type       |   pid  | status
------------------+--------------------+--------+---------
LogLoader         | logloaderd         |   N/A  | Down
dcm-daemon        | dcm-daemon         |   2787 | Alive
DSA               | DSA                |   N/A  | Down
OC4J              | home               |   2964 | Alive
OC4J              | OC4J_BI_Forms      |   2965 | Alive
OC4J              | OC4J_Portal        |   2967 | Alive
WebCache          | WebCache           |   2958 | Alive
WebCache          | WebCacheAdmin      |   2959 | Alive
HTTP_Server       | HTTP_Server        |   2961 | Alive
Discoverer        | ServicesStatus     |   2960 | Alive
Discoverer        | PreferenceServer   |   2963 | Alive
wireless          | performance_server |   2985 | Alive
wireless          | messaging_server   |   2986 | Alive
wireless          | OC4J_Wireless      |   2987 | Alive
```

You can use OPMN to start and stop your application server, monitor components, configure event scripts, and perform many other tasks related to process management. For example, you can use the following commands to start and stop OPMN and all OPMN-managed processes such as DCM, Oracle HTTP Server, OC4J instances, and Oracle Internet Directory, on UNIX:

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/opmn/bin/opmnctl stopall
```

On Windows, you can invoke these commands from the Windows Start menu. For example to start all processes, on Windows 2000, select **Start** > **Programs** > **OracleAS 10*g* - *Oracle_home_name*** > **Start** ***Oracle_home.hostname***.

See Also: *Oracle Process Manager and Notification Server Administrator's Guide*

Note that the following two processes are not started after you finish installing Oracle Application Server:

- Log Loader: This is a feature that compiles log messages from various log files into a single repository. If you would like to use Log Loader, you can start it after installation.

  See Also: Chapter 5, especially Section 5.5.1, "Starting and Stopping Log Loader"

- DSA: This is the OracleAS Guard server. If you are using OracleAS Guard, you can start this after installation.

  See Also: *Oracle Application Server High Availability Guide*

1.4.2 Getting Started with Distributed Configuration Management (DCM)

With Distributed Configuration Management (DCM), you can manage configuration information for application server instances, OracleAS Clusters, Oracle HTTP Server, Oracle Application Server Containers for J2EE (OC4J), Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider (JAZN), and OPMN.

DCM is installed and configured with every middle-tier and OracleAS Infrastructure installation. All DCM installations use a DCM repository. There are two types of DCM repositories:

- Database: This repository is stored in the OracleAS Metadata Repository in the DCM schema. This repository type is used by Portal and Wireless, and Business Intelligence and Forms installations. It is the repository for J2EE and Web Cache installations if you chose to join an OracleAS Database Farm during installation.
- File Based: This repository is stored in the file system in your Oracle home. This repository type is used by J2EE and Web Cache installations if you chose to join a File-Based farm during installation.

You can determine the repository type as follows:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl whichFarm
(Windows) ORACLE_HOME\dcm\bin\dcmctl whichFarm
```

During installation, DCM created a copy of your initial configuration. If, after you start configuring your application server, you would like to return to the initial configuration, you can use the `dcmctl restoreInstance` command.

You can use DCM to save and restore configuration information, deploy applications, manage OracleAS Clusters, and much more.

See Also: *Distributed Configuration Management Administrator's Guide*

1.4.3 Getting Started with Oracle HTTP Server

Oracle HTTP Server is installed and configured with every middle-tier and OracleAS Infrastructure installation.

You can access Oracle HTTP Server as follows:

```
http://hostname.domain:port
```

In the example, *port* is the Oracle HTTP Server Listen port number, which is listed in the `portlist.ini` file.

For example:

`http://hostname.domain:7778`

When you access Oracle HTTP Server, you see the Oracle Application Server Welcome Page. Click the link for **Oracle Application Server Logins** to log in to Application Server Control Console. Then, on the Home page, click **HTTP Server** to manage Oracle HTTP Server.

See Also:

- *Oracle HTTP Server Administrator's Guide*
- Section 2.3, "Getting Started with the Application Server Control Console"

1.4.4 Getting Started with Oracle Application Server Containers for J2EE (OC4J)

Oracle Application Server Containers for J2EE (OC4J) is a complete Java 2 Enterprise Edition (J2EE) environment.

When you install an instance, you get the following OC4J instances, depending on your configuration:

- `home`: The default OC4J instance that comes with every middle-tier installation
- `OC4J_BI_Forms`: Contains servlets that support OracleAS Reports Services, OracleBI Discoverer, and OracleAS Forms Services
- `OC4J_Portal`: Contains a servlet that supports OracleAS Portal
- `OC4J_Security`: Supports Identity Management Services
- `OC4J_Wireless`: Contains a servlet that supports OracleAS Wireless
- `oca`: Supports OracleAS Certificate Authority

You can use Application Server Control Console to manage OC4J instances.

See Also: *Oracle Application Server Containers for J2EE User's Guide*

1.4.5 Getting Started with OracleAS Web Cache

If you configured OracleAS Web Cache during installation, you can access it as follows:

`http://hostname.domain:port`

In the example, *port* is the Web Cache HTTP Listen port number as listed in the `portlist.ini` file.

For example:

`http://hostname.domain:7777`

When you access OracleAS Web Cache, you will see the Oracle Application Server Welcome Page, which is cached by OracleAS Web Cache. Click the link for **Oracle Application Server Logins** to log in to Application Server Control Console. Then, on the Home page, click **Web Cache** to manage OracleAS Web Cache.

Accessing OracleAS Web Cache Manager

In addition to using Application Server Control Console to manage OracleAS Web Cache, as described in Section 2.3, you can use OracleAS Web Cache Manager. OracleAS Web Cache Manager is a graphical user interface tool for configuring and monitoring OracleAS Web Cache.

You can access OracleAS Web Cache Manager by navigating to the following URL:

`http://`*`hostname.domain`*`:`*`port`*`/webcacheadmin`

In the example, *`port`* is the Web Cache HTTP Administration port number listed in the `portlist.ini` file.

For example:

`http://`*`hostname.domain`*`:9400/webcacheadmin`

You can log in to OracleAS Web Cache Manager as `ias_admin` or `administrator`. The password for both accounts is the `ias_admin` password you supplied during installation.

> **See Also:** *Oracle Application Server Web Cache Administrator's Guide* for a list of postinstallation configuration tasks

1.4.6 Getting Started with OracleAS Portal

If you configured OracleAS Portal during installation, you can access it as follows:

`http://`*`hostname.domain`*`:`*`port`*`/pls/portal`

In the example, *`port`* is the Web Cache HTTP Listen port number listed in the `portlist.ini` file.

For example:

`http://`*`hostname.domain`*`:7777/pls/portal`

You can log in as the user `portal`. If this is the first OracleAS Portal instance to use the OracleAS Metadata Repository, the password is the original `ias_admin` password you supplied for this middle tier during installation. The original `ias_admin` password is required, even if you changed the `ias_admin` password after installation.

> **See Also:** *Oracle Application Server Portal Configuration Guide* for information on getting started and managing OracleAS Portal

1.4.7 Getting Started with OracleAS Wireless

If you configured OracleAS Wireless during installation, you can access it as follows:

`http://`*`hostname.domain`*`:`*`port`*`/webtool/login.uix`

In the example, *`port`* is the Web Cache HTTP Listen port number listed in the `portlist.ini` file.

You can log in as `orcladmin` using the `orcladmin` password.

> **See Also:** *Oracle Application Server Wireless Administrator's Guide*

1.4.8 Getting Started with OracleBI Discoverer

If you configured OracleBI Discoverer during installation, you can access it as follows:

- Discoverer Viewer:

  `http://hostname.domain:port/discoverer/viewer`

- Discoverer Plus:

  `http://hostname.domain:port/discoverer/plus`

- Discoverer Portlet Provider:

  `http://hostname.domain:port/discoverer/portletprovider`

In the example, *port* is the Web Cache HTTP Listen port number in the `portlist.ini` file.

> **See Also:** *Oracle Business Intelligence Discoverer Configuration Guide* for additional steps for configuring Discoverer, including installing Discoverer workbooks and End User Layer (EUL) into each database that contains data to be analyzed

Before OracleBI Discoverer end users can start analyzing data, the steps in either or both the following sections must be completed:

- Prepare for Multidimensional Analysis
- Prepare for Relational Analysis

1.4.8.1 Prepare for Multidimensional Analysis

Before users can use OracleBI Discoverer to query data from a multidimensional data source, perform the following steps:

1. Install and prepare the database.

   See Also:

   - Section 1.4.8.1.1, "Prepare an Oracle9i Release 2 Database for Use with Discoverer"
   - Section 1.4.8.1.2, "Prepare Oracle Database 10g Enterprise Edition for Use with Discoverer"

2. Set up the data warehouse.

   If you are using Oracle Business Intelligence Warehouse Builder, see the *OracleBI Warehouse Builder User's Guide*. Otherwise, see the *Oracle9i OLAP User's Guide* (for Oracle9*i* Database) or the *Oracle OLAP Application Developer's Guide* (for Oracle Database 10*g*).

   The tasks to perform include:

   - Installing the schema
   - Creating appropriate metadata
   - (Optional) Creating an Analytic Workspace and adding content to it
   - Granting appropriate privileges to users (users must have `SELECT` privileges on the OLAP dimension tables, measures, and views)

   Note that all users of Discoverer Plus OLAP have the `D4OPUB` role, which in turn has the `OLAP_USER` role. The `OLAP_USER` role provides access to OLAP metadata in the database.

3. Use Application Server Control Console to install the Discoverer Catalog on the Oracle OLAP database instance to which the users will connect.

   See Also:

   - *Oracle Business Intelligence Discoverer Configuration Guide*
   - Application Server Control Console help system

4. Use Application Server Control Console to authorize database users.

   See Also: *Oracle Business Intelligence Discoverer Configuration Guide*

5. Perform optional configuration tasks as appropriate.
6. Provide end users with the information required to launch OracleBI Discoverer and to connect to the multidimensional data source.

   See Also: *Oracle Business Intelligence Discoverer Configuration Guide*

1.4.8.1.1 Prepare an Oracle9*i* Release 2 Database for Use with Discoverer Complete the following tasks to run against Oracle9*i* Release 2 Database:

1. If you have not already done so, install the Enterprise Edition of the Oracle9*i* Database, Release 2.

   - For instructions, download the Oracle9*i* installation guide for the appropriate platform from Oracle Technology Network:

     `http://www.oracle.com/technology`

   - See the *Oracle Application Server Installation Guide* for information on which database versions are supported.

   Note: When you install the database client, be sure to install it into a separate Oracle home directory.

2. Configure the database, following the configuration settings shown in *Best Practices for Tabular Cube Aggregation and Query Operations*.

   To access this document, download patch set 2529822. You must follow these configuration settings *exactly* to ensure that OracleBI Discoverer works correctly and performs well. Because this document is updated as needed, check for a new version whenever you download a new patch set.

   See Also: You can download this patch from Oracle*Metalink*:

   `http://metalink.oracle.com`

3. Define the appropriate OLAP metadata.

   As an alternative, you can create metadata using Oracle Warehouse Builder. If you do not define appropriate metadata, then you will not be able to create OLAP queries.

   See Also: *Oracle9i OLAP User's Guide*, Release 2 from the Oracle Database documentation library

If you do not define appropriate metadata, then you will not be able to create OLAP queries. Use one of the following tools to define the metadata:

- The OLAP management tool of Oracle Enterprise Manager.

  See Also: Enterprise Manager Online Help for information

- OracleBI Warehouse Builder

  See Also: *OracleBI Warehouse Builder User's Guide*

1.4.8.1.2 Prepare Oracle Database 10*g* Enterprise Edition for Use with Discoverer To run against Oracle Database 10*g* Enterprise Edition, complete the following tasks:

1. If you have not already done so, install Oracle Database 10*g* Enterprise Edition.

   - For instructions, download the Oracle Database 10*g* Enterprise Edition installation guide for the appropriate platform from Oracle Technology Network:

     `http://www.oracle.com/technology`

   - See the *Oracle Application Server Installation Guide* for information on which database versions are supported.

   - If you are migrating an existing Discoverer Catalog to Oracle Database 10*g* Enterprise Edition Release 2 (10.2.0.1 and later), then see Section 1.4.8.1.3.

   Note: When you install the database client, be sure to install it into a separate Oracle home directory.

2. Configure the database, following the configuration settings shown in *Best Practices for Tabular Cube Aggregation and Query Operations*.

   To access this document, download patch set 3760779. You must follow these configuration settings *exactly* to ensure that OracleBI Discoverer works correctly and performs well. Because this document is updated as needed, check for a new version whenever you download a new patch set.

   See Also: You can download this patch from Oracle*Metalink*:

   `http://metalink.oracle.com`

3. Define the appropriate OLAP metadata.

   See Also: *Oracle OLAP Application Developer's Guide* from the Oracle Database documentation library

   If you do not define appropriate metadata, then you will not be able to create OLAP queries. Use one of the following tools to define the metadata:

   - The OLAP management tool of Oracle Enterprise Manager.

     See Also: Enterprise Manager Online Help for information

   - OracleBI Warehouse Builder

     See Also: *OracleBI Warehouse Builder User's Guide*

   - Analytic Workspace Manager

See Also: *Oracle OLAP Application Developer's Guide* from the Oracle Database documentation library

1.4.8.1.3 How to Migrate an Existing Discoverer Catalog to Oracle Database 10*g* Enterprise Edition Release 2

To migrate an existing Discoverer Catalog to Oracle Database 10*g* Enterprise Edition Release 2 (10.2.0.1 and later), you must update a PL/SQL package on the database server by performing the following procedure:

1. Install the patch as follows:
   a. Locate the `d4o.jar` in the following directory.

      BI_Home`\sysman\webapps\emd\WEB-INF\lib`

   b. Extract the `bibcoreb.pls` file from `d4o.jar` to a local directory. The jar has packing scope and will be extracted into the `oracle\dss\persistence\storagemanager\bi\scripts` directory.
2. Apply the patch as follows:
   a. At the command prompt, enter:

      ```
      cd oracle\dss\persistence\storagemanager\bi\scripts
      ```

   b. Connect to the `D4OSYS` schema as the `D4OSYS` user with SQL*Plus.
   c. At the SQL*Plus prompt, enter:

      ```
      SQL> @bibcoreb.pls
      ```

      This following output displays:

      ```
      Package body created.
      Commit complete.
      ```

3. Ensure that the package is valid as follows:
   a. Open a SQL*Plus session.
   b. Enter the following SQL commands:

      ```
      SQL> column OBJECT_NAME format a30;
      SQL> column STATUS format a10;
      SQL> select object_name, status from user_objects where object_name='BISM_
      CORE';
      ```

      The following output indicates that the patch has been applied successfully:

      ```
      OBJECT_NAME                    STATUS
      ------------------------------------------------
      BISM_CORE                      VALID
      BISM_CORE                      VALID
      ```

1.4.8.2 Prepare for Relational Analysis

Before users can use Discoverer to query data from a relational data source, they must have access to an EUL that has been created (or upgraded) using either of the following:

- Discoverer EUL Command Line for Java

  See Also: *Oracle Business Intelligence Discoverer EUL Command Line for Java User's Guide* from the Business Intelligence Tools product CD

- Version of Discoverer Administrator shipped on the Oracle Business Intelligence Tools 10*g* 10.1.2 CD-ROM/DVD

  See Also: *Oracle Business Intelligence Discoverer Administration Guide* from the Business Intelligence Tools product CD

Note that if you upgrade an EUL from Discoverer version 9.0.2 or 9.0.4 (with EUL version number 5.0.*x*):

- The Discoverer version 10.1.2 EUL (with EUL version number 5.1.*x*) will overwrite (that is, destructively upgrade) the earlier EUL.
- Existing users cannot use Discoverer Version 9.0.*x* to access the EUL either during the upgrade process or after the upgrade process is completed.

Therefore, create a backup of the earlier EUL before upgrading.

See Also: *Oracle Business Intelligence Discoverer Administration Guide* from the Business Intelligence Tools product CD for more information about creating and upgrading an EUL, enabling users and roles, and setting up security

1.4.9 Getting Started with OracleAS Forms Services

If you configured OracleAS Forms Services during installation, you can access it as follows:

`http://`*`hostname.domain`*`:`*`port`*`/forms/frmservlet`

In the example, *`port`* is the Web Cache HTTP Listen port number listed in the `portlist.ini` file.

See Also: OracleAS Forms Services online help for more information on configuring and using OracleAS Forms Services

1.4.10 Getting Started with OracleAS Reports Services

If you configured OracleAS Reports Services during installation, you can access it as follows:

`http://`*`hostname.domain`*`:`*`port`*`/reports/rwservlet/getserverinfo`

In the example, *`port`* is the Web Cache HTTP Listen port number listed in the `portlist.ini` file.

You can log in as `orcladmin` with the `orcladmin` password.

See Also: *Oracle Application Server Reports Services Publishing Reports to the Web* for more information on configuring and using Reports

1.4.11 Getting Started with OracleAS Personalization

Before you use OracleAS Personalization, you must run the OracleAS Personalization Schema Creation Wizard, which creates the required schemas in the Oracle database. Then, you can start managing OracleAS Personalization.

See Also: *Oracle Application Server Personalization Administrator's Guide*

1.4.12 Getting Started with Oracle Application Server Integration Products

For information about getting started with Oracle Application Server Integration components, such as OracleAS Integration B2B, Oracle BPEL Process Manager, and Oracle BPEL Process Analytics, see the Oracle Application Server Integration documentation.

1.4.13 Getting Started with Identity Management Components

Table 1–3 describes how to access Identity Management components.

Table 1–3 Accessing Identity Management Components

Component	URL
Oracle Internet Directory Manager	On UNIX, use the following command: `$ORACLE_HOME/bin/oidadmin` On Windows, select **Start**, **Programs**, **Oracle - *Oracle_Home***, **Integrated Management Tools**, **Oracle Directory Manager**.
OracleAS Single Sign-On Administration pages	`http://host:7777/pls/orasso` Log in as `orcladmin` using the ias_admin password supplied during installation.
OracleAS Certificate Authority Administration Interface	`http://host:4400/oca/admin` Log in as OracleAS Certificate Authority Administrator using the password supplied during installation.

1.5 Task 5: Enable SSL (Optional)

During installation, SSL is not configured for some components. If you would like to enable SSL, refer to Part IV, "Secure Sockets Layer (SSL)".

2

Introduction to Administration Tools

This chapter introduces the Oracle Application Server administration tools.

It contains the following topics:

2.1 Overview of Oracle Application Server Administration Tools

Oracle realizes that the procedures you use to monitor and administer your application server components can vary, depending upon the size of your organization, the number of administrators you employ, and the types of components you manage. As a result, Oracle offers options for managing your Oracle Application Server installations.

These management options can be divided into the following categories:

2.1.1 Managing Oracle Application Server with Oracle Enterprise Manager 10*g*

The primary tool for managing Oracle Application Server—as well as your entire Oracle environment—is Oracle Enterprise Manager 10*g*.

With Enterprise Manager, you can use your Web browser to:

- Manage individual Oracle Application Server instances with Oracle Enterprise Manager 10*g* Application Server Control.
- Centrally manage all the components of your network and your enterprise with Oracle Enterprise Manager 10*g* Grid Control.
- Manage your Oracle Application Server Metadata Repository with the Oracle Enterprise Manager 10*g* Database Control.

When used together, Application Server Control, Grid Control, and Database Control provide a complete set of efficient tools to reduce the cost and complexity of managing your enterprise.

2.1.1.1 Using Application Server Control to Manage Oracle Application Server

Application Server Control is installed with every instance of Oracle Application Server. As a result, you can immediately begin managing your application server and its components from your Web browser.

Note: If you select the OracleAS Metadata Repository-only installation type, the Application Server Control is installed, but it is not configured or started automatically by the installation procedure. In fact, there is no need to start or use the Application Server Control Console for the Metadata Repository-only installation type.

See Section 2.5, "Managing the OracleAS Metadata Repository Database with Database Control" for information.

From the Application Server Control Console, you can monitor and administer a single Oracle Application Server instance, an Oracle Application Server Farm of application server instances, or an Oracle Application Server Cluster.

The Application Server Control Console organizes a wide variety of performance data and administrative functions into distinct, Web-based home pages for each application server component. The Enterprise Manager home pages make it easy to locate the most important monitoring data and the most commonly used administrative functions—all from your Web browser.

> **See Also:** Section 2.2, "About Oracle Enterprise Manager 10g Application Server Control"

2.1.1.2 Using Grid Control to Manage Your Enterprise

Oracle Enterprise Manager 10*g* Grid Control is installed from a separate installation CD-ROM, that is part of the Oracle Application Server CD–ROM pack. The Grid Control Console provides a wider view of your enterprise so you can manage multiple Oracle Application Server instances. In addition, the Grid Control Console provides a feature set designed to help you manage all aspects of your enterprise, including your Oracle databases, hosts, listeners, and other components.

> **See Also:** Section 2.6, "About Oracle Enterprise Manager 10g Grid Control"

2.1.1.3 Using Database Control to Manage an OracleAS Metadata Repository Database

Oracle Enterprise Manager 10*g* Database Control is installed and configured when you install the OracleAS Metadata Repository. As a result, you can use Database Control Console to manage the OracleAS Metadata Repository database.

The Database Control Console is similar to the Application Server Control Console, but it is designed to help you manage your Oracle database. It provides a Web-based user interface for performing database management tasks. For example, you can monitor the performance of the database, schedule backups, and manage the tablespaces of the database.

> **See Also:** Section 2.5, "Managing the OracleAS Metadata Repository Database with Database Control"

If you use OracleAS Metadata Repository Creation Assistant to install the OracleAS Metadata Repository in an existing database, Database Control will also be available if the existing database is an Oracle Database 10*g* database and if the Database Control was configured when the database was created.

Note that if you use Grid Control to manage your OracleAS Metadata Repository database, there is no need to use Database Control. All the features of Database Control are available from the database management pages within the Grid Control Console.

> **See Also:** Section 2.6, "About Oracle Enterprise Manager 10g Grid Control"

2.1.2 Managing Oracle Application Server from the Command Line

Oracle Application Server also provides command-line interfaces to several key management technologies. After you become familiar with the architecture and components of your application server, command-line tools can help you automate your management procedures with scripts and custom utilities.

The most important administration command-line tools are:

- `opmnctl`, which provides a command-line interface to Oracle Process Manager and Notification Server (OPMN). You can use `opmnctl` to:
  - Start and stop components, instances, and OracleAS Clusters
  - Monitor processes

  > **See Also:** Section 2.2.2, "About the Underlying Technologies" and *Oracle Process Manager and Notification Server Administrator's Guide*

- `dcmctl`, which provides a command-line interface to Distributed Configuration Management (DCM). You can use `dcmctl` to:
  - Create and remove OC4J instances and OracleAS Clusters
  - Deploy and undeploy OC4J applications
  - Archive and restore configuration information
  - Obtain configuration information

  > **See Also:** Section 2.2.2, "About the Underlying Technologies" and *Distributed Configuration Management Administrator's Guide*

In addition to `opmnctl` and `dcmctl`, Oracle Application Server provides many other command-line tools for performing specific tasks.

> **See Also:** Appendix B, "Oracle Application Server Command-Line Tools"

2.1.3 Using Other Tools to Monitor the Built-In Performance Metrics

After you install and start Oracle Application Server, the application server automatically begins gathering a set of built-in performance metrics. These built-in

performance metrics are measured continuously using performance instrumentation inserted into the implementations of Oracle Application Server components.

The Application Server Control Console presents a subset of these performance metrics in an organized fashion on the application server component home pages. For example, the Oracle HTTP Server metrics are presented as a series of charts on the Status Metrics page, which is available from the Oracle HTTP Server home page. You can also display all the metrics for a particular component by using the All Metrics page.

> **See Also:** Section 2.4.5, "Displaying the All Metrics Page for the Application Server or an Application Server Component"

Alternatively, you may want to view the complete set of built-in performance metrics, or you may need to monitor a specific set of application server component metrics. Oracle Application Server provides a set of command-line and servlet-based tools to view the Oracle Application Server built-in performance metrics directly, outside of the Application Server Control Console.

> **See Also:** *Oracle Application Server Performance Guide*

2.2 About Oracle Enterprise Manager 10*g* Application Server Control

Oracle Enterprise Manager 10*g* Application Server Control provides Web-based management capabilities designed specifically for Oracle Application Server. Using the Application Server Control Console, you can monitor, diagnose, and configure the components of your application server. You can deploy applications, manage security, and create and manage OracleAS Clusters.

Application Server Control consists of:

- The Application Server Control Console and its Enterprise Manager home pages that you use to manage Oracle Application Server

  These Web pages provide you with a high-level view of your Oracle Application Server environment. You can then drill down for more detailed performance and diagnostic information.

- The underlying software technologies that keep track of your application server instances and components

  These technologies automatically perform many of the management tasks as you select options and functions within the Application Server Control Console. For example, they discover the components of each application server instance, gather and process performance data, and provide access to application configuration information.

The following sections provide more information about Application Server Control:

- Introducing the Enterprise Manager Home Pages
- About the Underlying Technologies
- Managing Previous Versions of Oracle Application Server
- Using the Application Server Control Console Online Help

2.2.1 Introducing the Enterprise Manager Home Pages

Oracle Application Server provides a wide variety of software solutions designed to help you run all aspects of your business. As a result, you will want to manage Oracle Application Server from different levels of detail.

At times, you may want to manage a single application server instance; or, you may find it efficient to combine multiple instances into an OracleAS Cluster. At other times, you will want to manage a specific application server component.

To support these multiple levels of management, Oracle introduces the Oracle Enterprise Manager home pages. Each home page provides the information you need to monitor the performance and availability of Oracle Application Server from a particular level of management detail. Selected home pages also provide tools for configuring your Oracle Application Server components.

From each home page, you can obtain high-level information or you can drill down to get more specific information about an instance, component, or application.

Consider the following pages that are available when you use the Application Server Control Console:

- Use the OracleAS Farm page to view a set of related application server instances on your network and to create OracleAS Clusters that speed up the configuration and deployment of your Web applications. See Section 2.3.4, "Using the Oracle Application Server Farm Page" for more information.
- Use the Application Server Home page to manage all aspects of an individual application server instance. See Section 2.3.3, "Using the Application Server Home Page" for more information.
- Drill down to a component home page to monitor or configure an individual component of the application server. For example, use the Oracle HTTP Server Home page to monitor the performance of your Web server, or use the Oracle Application Server Containers for J2EE (OC4J) home page to deploy a custom Web-based application. See Section 2.3.5, "Using an Oracle Application Server Component Home Page" for more information.

2.2.2 About the Underlying Technologies

The Application Server Control Console relies on various technologies to discover, monitor, and administer the Oracle Application Server environment. Table 2–1 provides a summary of the underlying technologies leveraged by the Application Server Control Console.

Table 2–1 Summary of the Application Server Control Underlying Technologies

Technology	Description
Dynamic Monitoring Service (DMS)	The Application Server Control Console uses DMS to gather performance data about your Oracle Application Server components. For more information, see *Oracle Application Server Performance Guide*.
Oracle Process Manager and Notification Server (OPMN)	OPMN provides process control and monitoring for application server instances and their components. It gathers component status information, and distributes the status information to components that are interested in it. Application Server Control uses OPMN for such tasks as starting and stopping the components of your application server instance. For more information, see *Oracle Process Manager and Notification Server Administrator's Guide*.
Distributed Configuration Management (DCM)	DCM manages configurations among application server instances that are associated with a common Metadata Repository. It enables Oracle Application Server cluster-wide deployment so you can deploy an application to one instance and have it automatically propagated to the entire cluster. You can also make a single host or instance configuration change to one instance and have it propagated across all instances in the cluster. Application Server Control uses DCM to make configuration changes and to propagate configuration changes and deployed applications across the cluster. For more information, see *Distributed Configuration Management Administrator's Guide*.
Oracle Management Agent	A local version of the Oracle Management Agent designed specifically to monitor your application server components.
Oracle Management Watchdog Process	The Management Watchdog Process monitors the Management Agent and the Application Server Control Console to make sure both processes are running and available at all times. For more information, see *Oracle Enterprise Manager Advanced Configuration*.

2.2.3 Managing Previous Versions of Oracle Application Server

Previous versions of Oracle Application Server (specifically, Oracle9*i* Application Server 9.0.2 and 9.0.3) included the Oracle Enterprise Manager Web site, a Web-based tool that offers management capabilities similar to those provided by the Application Server Control Console.

In fact, you can still use the Enterprise Manager Web site to manage Oracle9*i* Application Server 9.0.2 and 9.0.3 after you begin deploying Oracle Application Server 10*g* (10.1.2 or 9.0.4) and its Application Server Control.

However, if you are familiar with the Enterprise Manager Web site and you plan to continue managing previous versions of Oracle Application Server, you should be aware of several differences between the Enterprise Manager Web site and the Application Server Control. In particular, you should note the following:

- Oracle9*i* Application Server (9.0.2) and Oracle9*i* Application Server (9.0.3) used one Enterprise Manager Web site to manage all the application server instances on a host.

  You could navigate to individual Enterprise Manager home pages for each application server, but only one instance of the Enterprise Manager Web site was running on the host and you managed all the application server instances from one Enterprise Manager Web site URL. This approach to application server management was convenient, but it required all application server instances to be installed and managed by the same operating system user.

- The current version of Oracle Application Server provides one Application Server Control for each application server instance on a host.

For example, if you install two application server instances on a single host, and you want to manage both instances, two separate instances of the Application Server Control—one for each application server instance—must be started on the host.

As a result, each application server instance provides a unique URL (specifically, a unique HTTP Server listening port number) for accessing the Application Server Control Console.

- If you have Oracle9*i* Application Server (9.0.2 or 9.0.3) and Oracle Application Server 10*g* (10.1.2 or 9.0.4) instances on the same host, and you have to deinstall a 9.0.2 or 9.0.3 instance, you must apply a patch to ensure Oracle Enterprise Manager continues to work after the change. Refer to the section on deinstallation of 9.0.2 or 9.0.3 Oracle homes after you upgrade to Oracle Application Server 10*g* (10.1.2) in the *Oracle Application Server Upgrade and Compatibility Guide*.

2.2.4 Using the Application Server Control Console Online Help

At any time while using the Application Server Control Console, you can click **Help** at the top of the page to get more information. In most cases, the Help window displays a help topic about the current page. Click **Contents** in the Help window to browse the list of help topics, or click **Search** to search for a particular word or phrase.

2.3 Getting Started with the Application Server Control Console

Use the following sections to get started with the Application Server Control Console and become familiar with the Enterprise Manager home pages within the Application Server Control Console:

- Displaying the Application Server Control Console
- Understanding the Initial Application Server Control Console Home Page
- Using the Application Server Home Page
- Using the Oracle Application Server Farm Page
- Using an Oracle Application Server Component Home Page

2.3.1 Displaying the Application Server Control Console

The following sections describe how to display the Application Server Control Console and introduce you to the initial home pages you should see when you display the Application Server Control Console for the first time:

- Using the Application Server Control Console URL
- Displaying the Application Server Control Console from the Welcome Page

2.3.1.1 Using the Application Server Control Console URL

The URL for the Application Server Control Console, including the port number, is included in the text file that displays at the end of the Oracle Application Server installation procedure. This text file is saved in the following location after you install the application server:

```
(UNIX) ORACLE_HOME/install/setupinfo.txt
(Windows) ORACLE_HOME\install\setupinfo.txt
```

The Application Server Control Console URL typically includes the name of the host computer and the port number assigned to the Application Server Control Console during the installation. For example:

```
http://mgmthost1.acme.com:1156
```

2.3.1.2 Displaying the Application Server Control Console from the Welcome Page

To view the Application Server Control Console from the Oracle Application Server Welcome Page:

1. Display the Oracle Application Server Welcome Page by entering the following URL in your Web browser:

   ```
   http://hostname.domain:port
   ```

   For example:

   ```
   http://sys42.acme.com:7777
   ```

   > **Note:** The default port for Oracle HTTP Server (and, as a result, the Welcome page) is provided at the end of the Oracle Application Server installation, as well as in the following text file in the `install` directory of the application server Oracle home:
   >
   > `setupinfo.txt`

2. Click **Log on to the Oracle Enterprise Manager 10*g* Application Server Control Console**.

   Enterprise Manager displays the administrator logon dialog box.

3. Enter the Oracle Application Server administrator user name and password and click **OK**.

   The user name for the administrator user is `ias_admin`. The password is the one you supplied during the installation of Oracle Application Server.

2.3.2 Understanding the Initial Application Server Control Console Home Page

When you first display the Application Server Control Console, the initial home page you see varies depending upon whether or not the instance belongs to an OracleAS Farm.

> **See Also:** "What is a Farm?" in the *Oracle Application Server Installation Guide* for your platform

Table 2–2 describes the Enterprise Manager home pages that might be used as a starting point when you first browse to the Application Server Control Console.

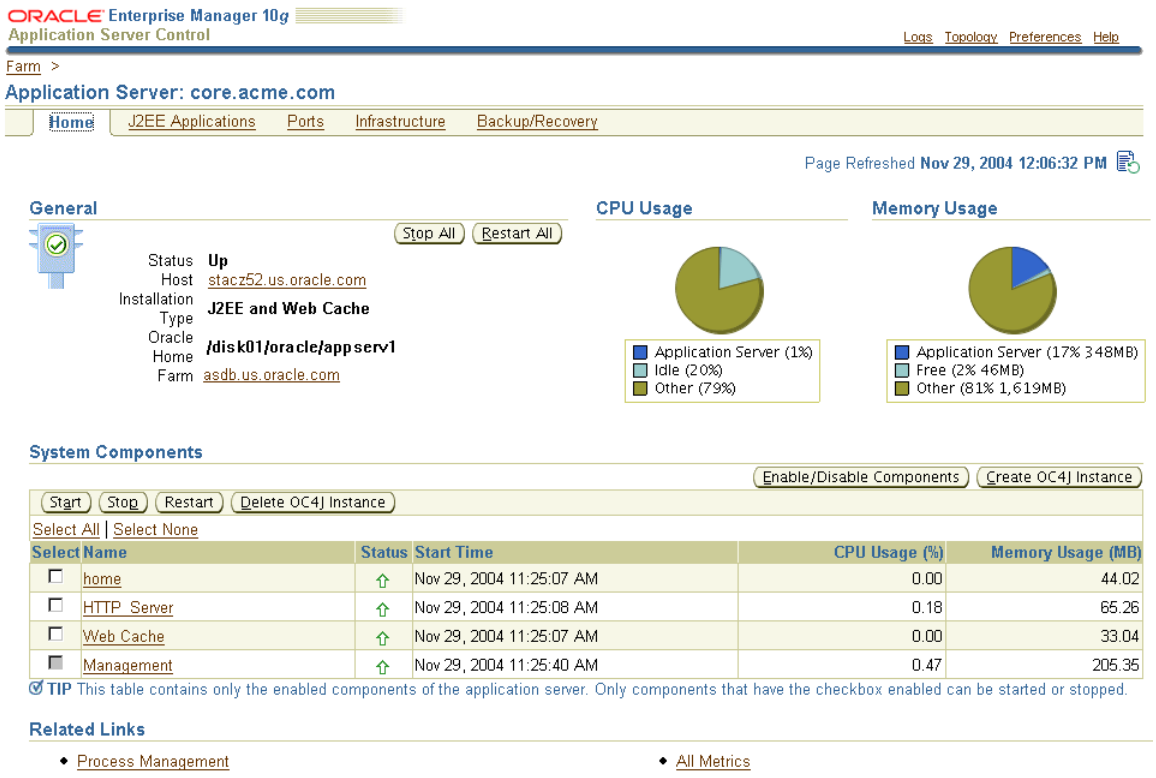
Table 2–2 Enterprise Manager Home Pages for Managing Oracle Application Server

Enterprise Manager Home Page	Description
Application Server Home page	Use this home page to monitor and configure a single application server instance. See Section 2.3.3, "Using the Application Server Home Page" for more information. The Application Server home page is the first page you see if you have installed a single application server instance that is not using an OracleAS Metadata Repository.
OracleAS Farm page	Use this page to view a list of all the application server instances that use a common OracleAS Farm. See Section 2.3.4, "Using the Oracle Application Server Farm Page" for more information. The OracleAS Farm page is the first page you see if you have installed one or more application server instances that use a common set of Infrastructure Services—or more specifically, a common file-based or database-based OracleAS Metadata Repository.

2.3.3 Using the Application Server Home Page

From the Application Server Home page (Figure 2–1), you can start and stop the application server instance, monitor the overall performance of the server, and review the components of the server. You can also drill down and examine the performance of a particular component and configure the component.

Figure 2–1 Application Server Home Page



The Application Server Home page provides a table that lists the components of the application server. From this table, you can also get a snapshot of how each individual component is performing.

From the **System Components** table, you can display a home page for each component of the application server.

You can perform the following management functions from the Application Server Home page:

- Click **Logs** at the top of the page to locate and search the various Oracle Application Server log files, as well as the Oracle Application Server Log Repository.
- Click **Topology** to view the Topology Viewer, which provides a graphical view of the application server processes managed by OPMN.

  See Also: Section 2.4.1, "Reviewing the Application Server Component Topology"

- Click **J2EE Applications** to display a list of the applications deployed across all the OC4J instances within this Oracle Application Server.
- Click **Ports** to view a list of all the ports currently in use by the various Oracle Application Server components. You can also modify many of the port assignments when necessary.
- Click **Infrastructure** to configure Identity Management, Grid Control Management, or OracleAS Farm Repository Management.
- Click **Backup/Recovery** to perform backup and recovery operations for the selected Oracle Application Server instance.

  See Also: Part V, "Backup and Recovery" and the Enterprise Manager online help for more information about backup and recovery procedures.

- Click **Enable/Disable Components** to control whether or not the selected components appears in the list of system components and whether or not the component is affected by server-wide actions, such as **Start All** or **Restart All**. When a component is disabled, it does not consume any system resources and you can always enable it later.

  See Also: Section 3.4, "Enabling and Disabling Components"

For more information, click **Help** after selecting an option on the Application Server Home page.

See Also: Section 2.2.4, "Using the Application Server Control Console Online Help"

2.3.4 Using the Oracle Application Server Farm Page

If your application server instance is part of an OracleAS Farm, your start page for the Application Server Control Console is the OracleAS Farm page (Figure 2–2).

See Also: "What is a Farm?" in the *Oracle Application Server Installation Guide* for your platform

The OracleAS Farm page displays a list of the standalone application server instances and OracleAS Clusters associated with your Infrastructure Services. Standalone instances are application server instances that are not part of an OracleAS Cluster.

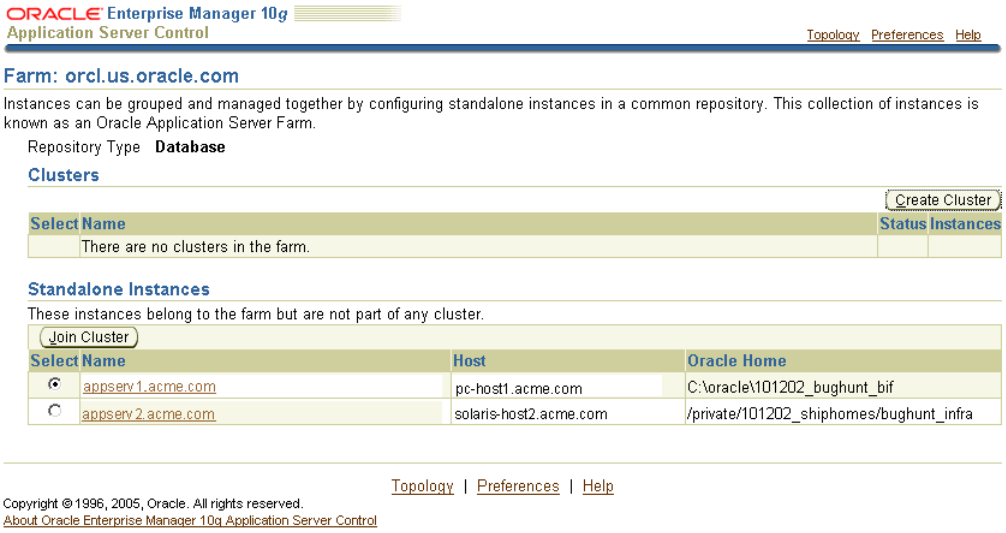
You can configure your application server instance to use Infrastructure Services by clicking **Infrastructure** on the Application Server Home page. For more information, see the Enterprise Manager online help.

Using the Farm page, you can perform the following tasks:

- View multiple application server instances on multiple hosts
- Drill down to the Application Server Home page for each instance
- Create and manage OracleAS Clusters

> **See Also:** *Oracle Application Server High Availability Guide* for more information about using OracleAS Clusters

Figure 2–2 OracleAS Farm Home Page



2.3.5 Using an Oracle Application Server Component Home Page

Component home pages vary from one component to another because each component has different monitoring and configuration requirements. However, most of the component home pages have the following common elements:

- A general information section that includes an icon to indicate the current state of the component and buttons for starting and stopping the component (if applicable)
- Status information, including CPU and memory usage information, so you can get a snapshot of how the component is performing
- Component-specific information, such as a virtual hosts tab on the HTTP Server home page or a list of deployed applications on the OC4J home page
- Links to administrative functions where appropriate, so you can modify the configuration of selected components. In many cases, this means you can use a graphical user interface to modify configuration files.

2.4 Monitoring and Diagnosing with the Application Server Control Console

The Application Server Control Console is designed to encourage a top-down approach to monitoring and diagnostic activities. For example, you can start by reviewing the basic characteristics of your application server and then drill down to examine the performance of individual components of the server.

The following sections provide an outline of this monitoring methodology:

- Reviewing the Application Server Component Topology
- Reviewing General Information and Resource Usage
- Reviewing the Resources of the Application Server Host
- Monitoring Application Server Components
- Monitoring J2EE Applications
- Obtaining More Information About Monitoring Oracle Application Server

2.4.1 Reviewing the Application Server Component Topology

Click **Topology** at the top of any page in Application Server Control Console to display the Topology Viewer (Figure 2–3). The Topology Viewer provides a graphical, real-time representation of application server processes managed by Oracle Process Manager and Notification Server (OPMN).

> **See Also:** Section 2.2.2, "About the Underlying Technologies" for more information about OPMN and the other technologies used by Application Server Control

The Topology Viewer identifies each component of the application server with an icon. The position of each icon on the page and the connections between the icons represent the relationships between each component. Visual clues in the Topology Viewer help you quickly identify components that are down or performing poorly.

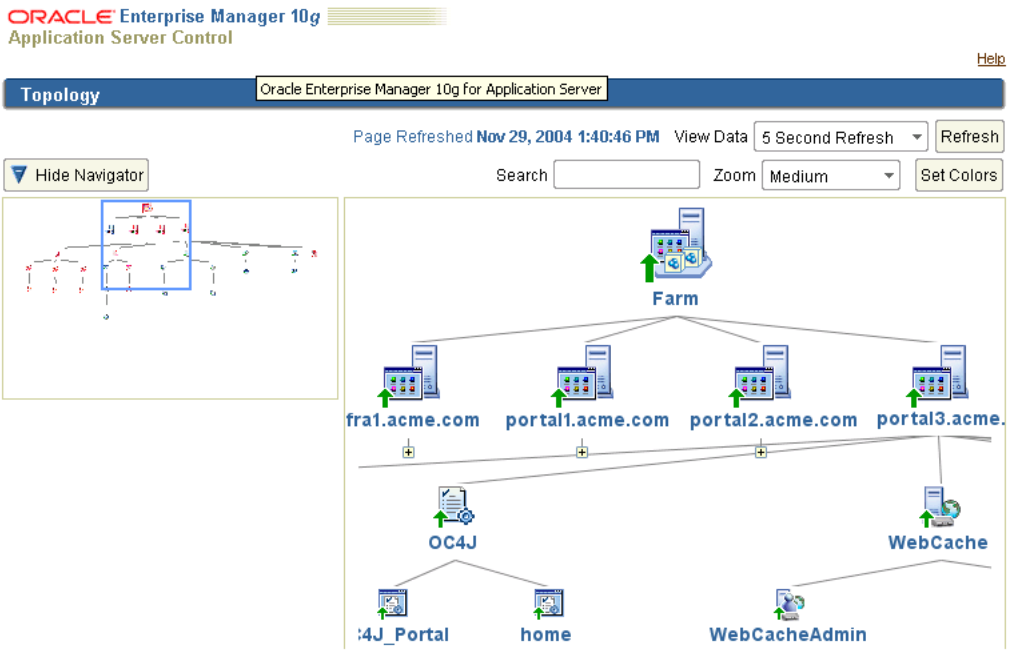
If you are managing multiple application servers as part of an OracleAS Farm, the viewer also shows the relationships between the application server instances, including any OracleAS Clusters you have created.

From the Topology Viewer, you can perform the following tasks:

- View the status of components
- Start, stop, or restart processes
- Monitor performance across the application server environment
- Drill down to component home pages for details

> **See Also:** 'About Topology Viewer" in the Enterprise Manager online help

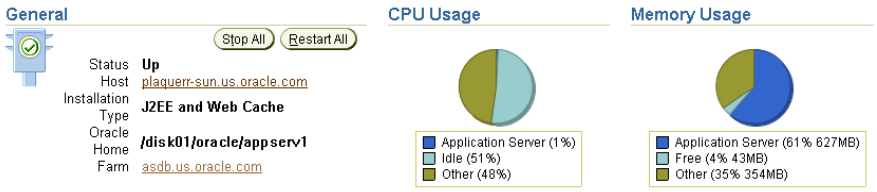
Figure 2–3 Topology Viewer



2.4.2 Reviewing General Information and Resource Usage

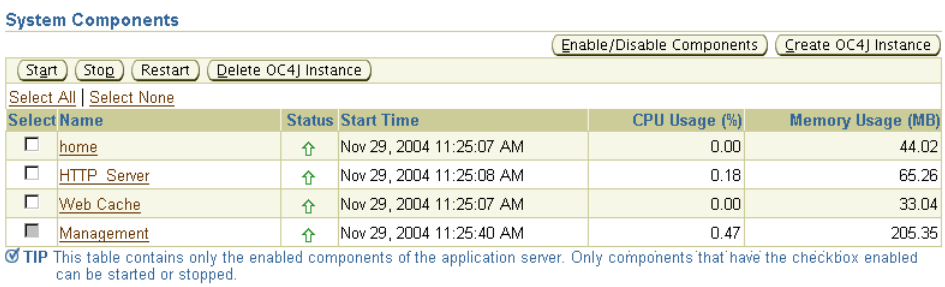
The Application Server Home page provides general information about the status of your server, including the name, location, and application server availability. The home page also provides high-level information about CPU and memory usage. When reviewing the home page, review the CPU Usage and Memory Usage charts for excessive CPU or memory usage by the application server (Figure 2–4).

Figure 2–4 General Section of the Application Server Home Page



If you suspect that the application server is using too many resources, review the list of components to review the resource usage of each component (Figure 2–5).

Figure 2–5 System Components Table on the Application Server Home Page



Consider disabling any components that you are not currently using as part of this application server instance. Disabled components are not started when you start the application server and as a result do not consume system resources. You can always enable a disabled application server component at a later time.

> **See Also:** 'Disabling and Enabling Components" in the Enterprise Manager online help

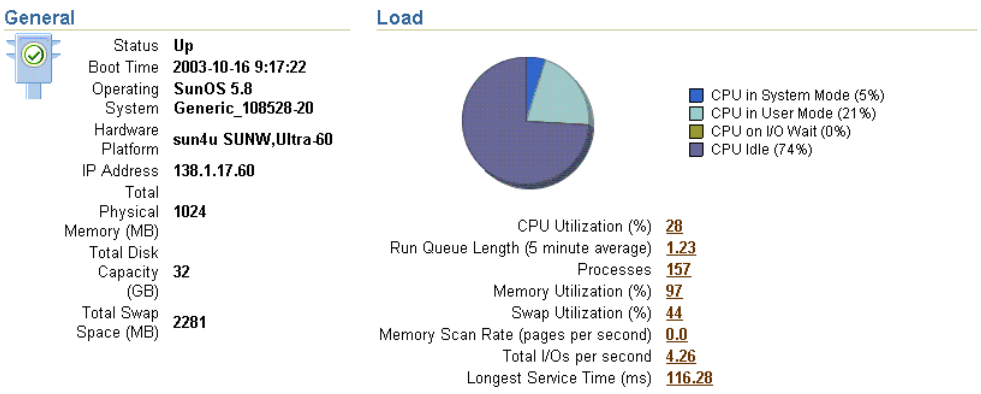
2.4.3 Reviewing the Resources of the Application Server Host

Many performance or configuration issues are directly related to a lack of available resources on the host. Before you drill down to analyze the performance and resource usage of the individual application server components, review the resources and characteristics of the application server host.

Click the host name in the General section of the Application Server home page to display the Host home page. The Host home page provides a summary of the operating system, memory, and disk capacity. The Load section of the page provides a CPU chart that breaks down the CPU usage into categories of usage; the load metrics beneath the chart provide details about system memory usage (Figure 2–6).

> **See Also:** "About Memory Usage" in the Enterprise Manager online help for information about how Enterprise Manager calculates the memory usage for your application server.

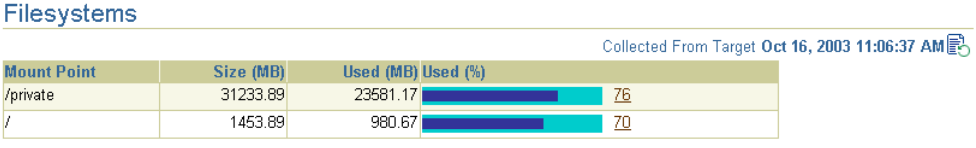
Figure 2–6 General Information and Load Statistics on the Host Home Page



Scroll to the bottom of the page to view a set of links to real-time performance metrics. If you are concerned about the CPU and memory usage on the system, click **Top Processes** to display tables listing the processes that are using the most resources on the host.

Click **Filesystems** to display a bar chart that reveals the amount of disk space available on the application server host (Figure 2–7).

Figure 2–7 Disk Space Usage Chart Available from the Host Home Page



2.4.4 Monitoring Application Server Components

After you review the high-level performance metrics and the resources available on the application server host computer, you can then begin to look for potential issues within the individual application server components.

To diagnose problems with individual application server components, click the component name in the **System Components** table on the Application Server home page. This technique of "drilling down" to obtain more detail can help you isolate problems in a particular component or area of the application server.

2.4.5 Displaying the All Metrics Page for the Application Server or an Application Server Component

The Application Server Control Console provides you with selected performance metrics that you can use to determine the overall performance of your application server. In some cases, the metrics are shown as performance charts; in other cases, you can monitor the real-time value of the metrics in numeric format.

For a comprehensive list of the metrics that are monitored by Enterprise Manager, you can view the All Metrics page. The All Metrics page is available from the Application Server Home page and from each of the component Home pages.

For example, to view All Metrics page for an application server instance:

1. Navigate to the Application Server Home page.
2. Click **All Metrics** in the Related Links section of the page.

   Enterprise Manager displays the All Metrics page for the application server.
3. Click **Expand All** to see all the application server metrics in each of the metric categories.

   Figure 2–8 shows the Application Server All Metrics page after you have expanded all the metric categories.
4. Click the name of metric to display the Metric detail page.
5. Click **Help** to display information about the metric.

Figure 2–8 Application Server All Metrics Page

To view the All Metrics page for a component, such as Oracle HTTP Server:

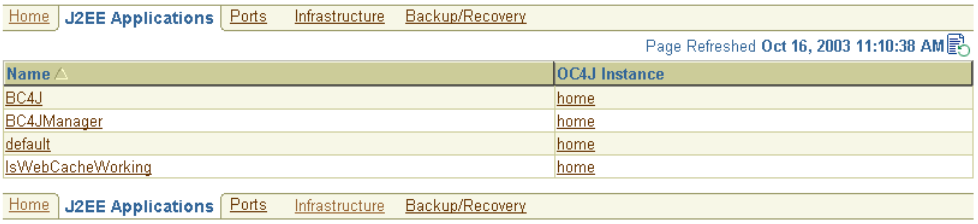
1. Navigate to the component Home page.
2. Click **All Metrics** in the Related Links section of the page.

   Enterprise Manager displays the All Metrics page for the selected component.

2.4.6 Monitoring J2EE Applications

The J2EE applications you deploy and maintain with Oracle Application Server represent the most important aspects of your application server deployments. As a result, Enterprise Manager also provides a shortcut you can use to review the performance of your J2EE applications. Simply click **J2EE Applications** on the Application Server home page to display a list of the applications deployed from this application server instance (Figure 2–9).

Figure 2–9 List of Applications on the J2EE Applications Page



From this list of J2EE applications, you can navigate quickly to the OC4J instance or application page for information on the performance and availability of each application you have deployed.

2.4.7 Obtaining More Information About Monitoring Oracle Application Server

For more complete information about monitoring Oracle Application Server, refer to the Application Server Control Console online help and the *Oracle Application Server Performance Guide*.

2.5 Managing the OracleAS Metadata Repository Database with Database Control

Many features of Oracle Application Server depend upon the OracleAS Metadata Repository, which uses an Oracle database to contain the OracleAS Metadata Repository. When you install the OracleAS Metadata Repository, you can choose to install a preconfigured instance of Oracle Database 10*g* for the OracleAS Metadata Repository.

> **See Also:** *Oracle Application Server Installation Guide* for your platform

If you have installed and deployed Oracle Enterprise Manager 10*g* Grid Control, you can also use the Grid Control Console to manage the OracleAS Metadata Repository.

> **See Also:** Section 2.6, "About Oracle Enterprise Manager 10g Grid Control"

However, if you are not centrally managing your environment with Grid Control, the database that is installed to host the OracleAS Metadata Repository comes with its own management tools.

Specifically, the OracleAS Metadata Repository database comes with Oracle Enterprise Manager 10*g* Database Control, which is provided with Oracle Database 10*g*.

To display the Database Control, which you can use to manage the OracleAS Metadata Repository database:

1. Use a Web browser to access the Database Control URL:

   ```
   http://hostname.domain:port/em
   ```

   In this example:

   - *hostname* is the name of the computer on which you installed Oracle Database.
   - *domain* is the domain of your computer.
   - *port* is the port number reserved for the Database Control during installation.

   If you do not know the correct port number to use, look for the following line in the `portlist.ini` file, which is stored in the `install` directory of your OracleAS Metadata Repository Oracle home:

   ```
   Enterprise Manager Console HTTP Port (db_name) = 5500
   ```

The installation reserves the first available port from the range 5500 to 5519. For example, if you installed Oracle Database on host `mgmt42`, and the Database Control uses port 5500, enter the following URL:

```
http://mgmt42.acme.com:5500/em
```

Oracle Enterprise Manager displays the Database Control login page.

2. Log in to the database using the user name SYS and connect as SYSDBA.

   Use the password that you specified for the SYS account during the installation.

3. Enterprise Manager displays the Database Home page (Figure 2–10).

   From the Database Home page, you can review the current state of your database and access a wide range of monitoring and administration features.

> **See Also:** *Oracle 2 Day DBA* in the Oracle Database 10*g* documentation library for an introduction to database management with the Database Control Console

Figure 2–10 Database Home Page in the Database Control Console

2.6 About Oracle Enterprise Manager 10*g* Grid Control

Application Server Control provides all the tools you need to manage your application server instances, farms, clusters, and system components. However, if you have an environment that includes other Oracle products and applications in addition to Oracle Application Server, consider using Oracle Enterprise Manager 10*g* Grid Control.

Grid Control, when used with Application Server Control, provides a wider view of your Oracle environment beyond the application server. From a central location, you can use the Grid Control Console to manage databases, application servers, and Oracle applications across your entire network.

The Grid Control Console offers advanced management features, such as a notification system to notify administrators of changes in your environment and a Job system to automate standard and repetitive tasks, such as executing a SQL script or executing an operating system command.

The following sections provide more information about Grid Control:

- About the Components of Grid Control
- Installing the Grid Control Components
- Logging In to the Grid Control Console
- Viewing a List of Application Servers in the Grid Control Console
- Overview of Grid Control Monitoring Tasks
- Obtaining More Information About Grid Control

2.6.1 About the Components of Grid Control

When you centrally manage your enterprise, including your Oracle Application Server instances, you take advantage of the Enterprise Manager three-tier architecture:

- The Grid Control Console provides a Web-based graphical interface you can use to manage all aspects of your enterprise.
- The Oracle Management Service and Management Repository provide a scalable middle tier for storing crucial management data and processing system management tasks.

  Note that the Management Repository is a separate repository from the OracleAS Metadata Repository. The Management Repository is designed specifically for Enterprise Manager.
- The Oracle Management Agent, which you install on each host computer on which there are services to be monitored, monitors the host services and executes tasks from the Management Service.

> **See Also:** *Oracle Enterprise Manager Concepts* for more information about the Oracle Enterprise Manager 10*g* components and architecture

2.6.2 Installing the Grid Control Components

You install Oracle Enterprise Manager 10*g* Grid Control from a separate CD-ROM.

To centrally manage your enterprise, you typically perform the following steps:

1. Install the Management Service and the Management Repository on a host computer.

2. Install the Oracle Management Agent on each of the computers that you want to manage from the Grid Control Console.

   Note: You install the Oracle Management Agent into its own Oracle home directory on each managed hosts.

   On each host, the Management Agent gathers information about the various targets on the host. A target is a software component (such as Oracle Application Server), a host computer, and or other service that you manage with Oracle Enterprise Manager 10*g*.

   Specifically, information about the targets on a host are discovered by the Management Agent during the Management Agent installation. When a target is discovered, information about the target is added to the Management Repository and the target displayed in the list of managed targets in the Grid Control Console.

   If you later install additional application servers on a managed host, you can add them to the Grid Control Console later. To add additional application server targets, click **Add** on the Application Servers page in the Grid Control Console, or use the Grid Control Management section of the Infrastructure page in the Application Server Control Console.

   See Also: *Oracle Enterprise Manager Grid Control Installation and Basic Configuration* for complete instructions about installing Grid Control and starting and stopping the Grid Control components

   Oracle Enterprise Manager Advanced Configuration for information about common configurations when installing the Grid Control components

   "Configuring Your Application Server for Grid Control Management" in the Application Server Control Console online help

2.6.3 Logging In to the Grid Control Console

After you have configured and started Oracle Management Service, you can log in to the Grid Control Console by entering the following URL in your Web browser:

`http://`*`grid_control_hostname`*`.`*`domain`*`:`*`port`*`/em`

For example:

```
http://mgmthost2.acme.com:7777/em
```

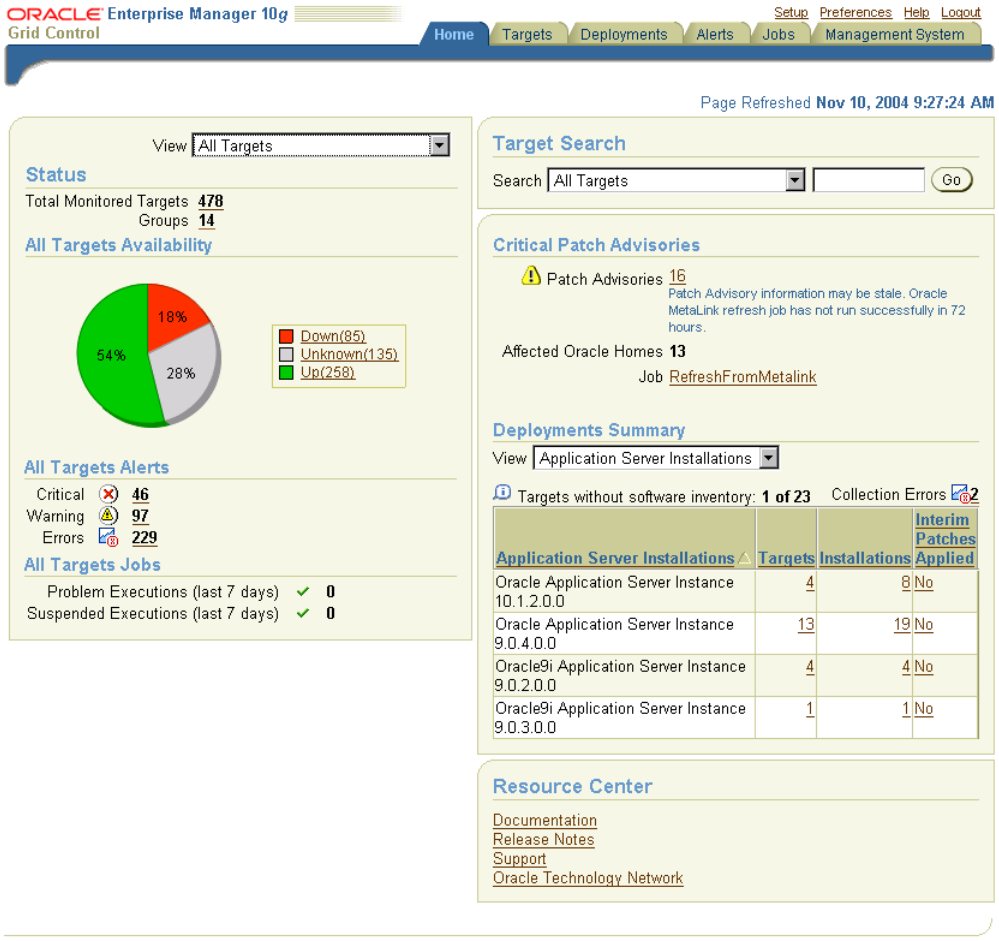
If you are uncertain about the port number, you can refer to one of the following files in the `install` directory of the Management Service Oracle home after you install the Management Service:

- The `setupinfo.txt`, which includes information displayed by the Oracle Universal Installer at the end of the Grid Control install
- The `portlist.ini`, which describes the ports assigned during the Management Service installation

When the Grid Control login page appears, enter the username and password for the Super administrator SYSMAN account, which you defined during the Grid Control installation.

After you log in, Enterprise Manager displays the Grid Control Console Home page (Figure 2–11).

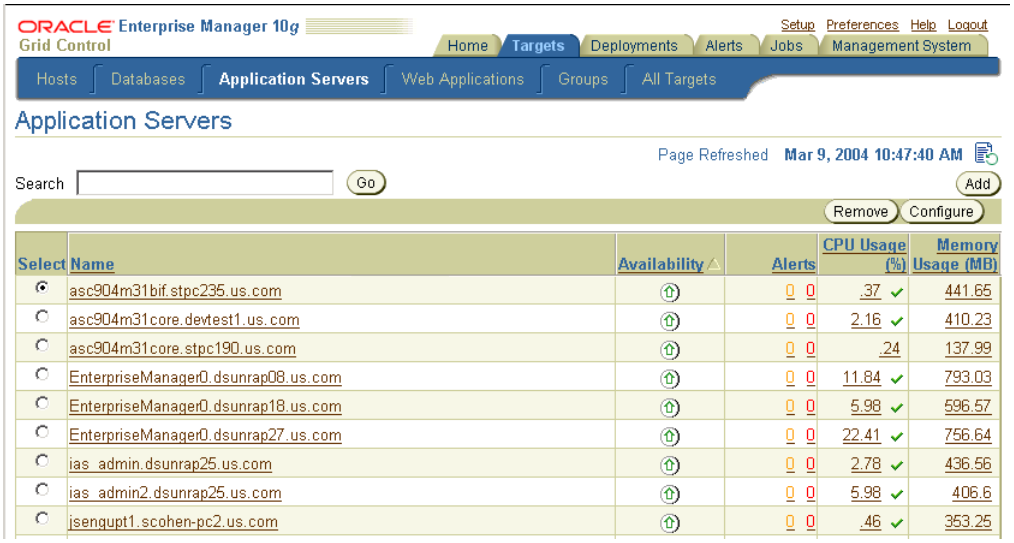
Figure 2–11 Grid Control Console Home Page



2.6.4 Viewing a List of Application Servers in the Grid Control Console

From the Grid Control Console home page, click the **Targets** tab and then click **Application Servers** in the horizontal navigation bar. Enterprise Manager displays the Application Servers page (Figure 2–12), which lists all the application servers currently being monitored by Oracle Management Agents in your enterprise.

Figure 2–12 List of Application Servers in the Grid Control Console



This list provides you with a snapshot of the availability, number of alerts, and the CPU and memory usage of each application server target.

2.6.5 Overview of Grid Control Monitoring Tasks

After you have installed the Management Agent on the Oracle Application Server hosts and have identified your application server targets in the Grid Control Console, you can perform a variety of monitoring tasks. For example, you can:

- Set and adjust a set of default metric thresholds for the application servers that you monitor. You can then configure Enterprise Manager so you are notified automatically when a particular application server metric reaches its threshold.
- Organize your application server targets into groups so you can monitor them as a single unit; groups also allow you to compare the performance of the application servers you monitor and to perform administration tasks, such as blackouts, on the group.
- Review historical data and analyze trends in the performance of your application server components and J2EE applications. For example, you can:
  - Emulate and monitor the client experience from remote locations.
  - Measure real end-user performance against a Web application.
  - Trace Web site transactions through the application stack, Oracle HTTP Server, OC4J, and the back-end Oracle database.
  - Correlate application performance across components to rapidly isolate problems.
- Use Application Service Level Management to measure the performance and availability of your J2EE Web applications.
- Perform configuration management tasks, such as software and hardware inventory tracking, cloning, and patching.

2.6.6 Obtaining More Information About Grid Control

For information about starting, configuring, and using Grid Control, see the following documentation:

- *Oracle Enterprise Manager Grid Control Installation and Basic Configuration*
- *Oracle Enterprise Manager Concepts*
- *Oracle Enterprise Manager Advanced Configuration*

The Grid Control Console also provides extensive online help. To display the Grid Control Console online help, click **Help** at the top of any of the Grid Control Console pages.

3

Starting and Stopping

This chapter describes various procedures for starting and stopping Oracle Application Server.

It contains the following topics:

3.1 Overview of Starting and Stopping Procedures

Oracle Application Server is a flexible product that you can start and stop in different ways, depending on your requirements. See the following sections:

- Section 3.2, "Starting and Stopping Application Server Instances"

  Use the procedures in this section when starting an instance from scratch, for example, after restarting a host, or when you want to stop your entire instance, for example, in preparation for shutting down your system.

- Section 3.3, "Starting and Stopping Components"

  Use the procedures in this section after you have started your instance and want to start or stop individual components.

- Section 3.4, "Enabling and Disabling Components"

  This section describes how to disable components (prevent them from starting when you start an instance) and enable components (allow them to start when you start an instance).

- Section 3.5, "Starting and Stopping an Oracle Application Server Environment"

  This section describes how to perform an orderly shutdown of your entire environment.

3.2 Starting and Stopping Application Server Instances

This section describes how to start and stop application server instances. It contains the following topics:

3.2.1 Starting OracleAS Infrastructure

This section describes how to start all processes in an OracleAS Infrastructure. Follow this procedure after you have restarted your host, or any other time you want to start up your entire OracleAS Infrastructure.

This procedure applies to all OracleAS Infrastructure types:

- Oracle Identity Management and OracleAS Metadata Repository

  Follow both steps to start Oracle Identity Management and OracleAS Metadata Repository.

- OracleAS Metadata Repository only

  Follow only Step 1 to start OracleAS Metadata Repository. You do not need to perform the second step of starting Oracle Identity Management because you do not need OPMN or the Application Server Control Console in a OracleAS Metadata Repository-only installation.

- Oracle Identity Management only

  Follow only Step 2 to start Oracle Identity Management. Make sure the OracleAS Metadata Repository that supports Oracle Identity Management (residing in another Oracle home) is already started.

To start OracleAS Infrastructure:

1. If your OracleAS Infrastructure contains OracleAS Metadata Repository, start it as follows:

   a. Set the ORACLE_HOME environment variable to the OracleAS Infrastructure Oracle home.

   b. Set the ORACLE_SID environment variable to the OracleAS Metadata Repository SID (default is `orcl`).

   c. Start the Net Listener:

   ```
   ORACLE_HOME/bin/lsnrctl start
   ```

   d. Start the OracleAS Metadata Repository instance:

   ```
   ORACLE_HOME/bin/sqlplus /nolog
   SQL> connect SYS as SYSDBA
   SQL> startup
   SQL> quit
   ```

   e. Start the Oracle Enterprise Manager 10*g* Database Control:

   ```
   emctl start dbconsole
   ```

2. If your OracleAS Infrastructure contains Oracle Identity Management, start it as follows:

   a. Start components:

   ```
   opmnctl startall
   ```

This command starts OPMN and all OPMN-managed processes such as DCM, Oracle HTTP Server, OC4J instances, and Oracle Internet Directory.

b. Start the Application Server Control Console:

```
emctl start iasconsole
```

Alternatively, on Windows, you can start the Application Server Control Console using the Windows Services control panel. The name of the service for the Application Server Control uses the following format:

```
Oracle<OracleHomeName>ASControl
```

See Section A.1.2 for more information.

Alternatively, on Windows, you can start the Infrastructure from the Programs menu: **Start** > **Programs** > **Oracle Application Server Infrastructure** - ***Infra_name*** > **Start** ***instanceName***.

3.2.2 Stopping OracleAS Infrastructure

This section describes how to stop all processes in OracleAS Infrastructure. Follow this procedure when you are preparing to shut down your host, or any other time you want to stop your entire OracleAS Infrastructure.

This procedure applies to all OracleAS Infrastructure types:

- Oracle Identity Management and OracleAS Metadata Repository

  Follow both steps to stop Oracle Identity Management and OracleAS Metadata Repository.

- OracleAS Metadata Repository only

  Follow step 2 only to stop OracleAS Metadata Repository.

- Oracle Identity Management only

  Follow step 1 only to stop Oracle Identity Management.

To stop OracleAS Infrastructure:

1. If your OracleAS Infrastructure contains Oracle Identity Management, stop it as follows:

   a. Stop the Application Server Control Console:

   ```
   emctl stop iasconsole
   ```

   Alternatively, on Windows, you can stop the Application Server Control Console using the Services control panel. See Section A.1.2 for more information.

   b. Stop components:

   ```
   opmnctl stopall
   ```

   This command stops OPMN and all OPMN-managed processes such as DCM, Oracle HTTP Server, OC4J instances, and Oracle Internet Directory.

2. If your OracleAS Infrastructure contains OracleAS Metadata Repository, stop it as follows:

a. Set the ORACLE_HOME environment variable to the OracleAS Infrastructure Oracle home.

b. Set the ORACLE_SID environment variable is set to the OracleAS Metadata Repository SID (default is `orcl`).

c. Stop the OracleAS Metadata Repository instance:

```
ORACLE_HOME/bin/sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> shutdown
SQL> quit
```

d. Stop the Net Listener:

```
ORACLE_HOME/bin/lsnrctl stop
```

e. Stop the Oracle Enterprise Manager 10*g* Database Control:

```
emctl stop dbconsole
```

Alternatively, on Windows, you can stop the Infrastructure from the Programs menu: **Start** > **Programs** > **Oracle Application Server Infrastructure** - *Infra_name* > **Stop** ***instanceName***.

3.2.3 Starting a Middle-Tier Instance

This section describes how to start all processes in a middle-tier instance. You can follow this procedure after you have restarted your host, or any other time you want to start up the entire instance.

This procedure applies to all middle-tier instance types:

- J2EE and Web Cache
- Portal and Wireless
- Business Intelligence and Forms

To start a middle-tier instance:

1. If the middle-tier instance uses OracleAS Infrastructure services, such as Oracle Identity Management or OracleAS Metadata Repository, make sure they are started.
2. Start components:

   ```
   opmnctl startall
   ```

   This command starts OPMN and all OPMN-managed processes such as DCM, Oracle HTTP Server, OC4J instances, and OracleAS Web Cache, OracleAS Forms Services, and OracleAS Reports Services.
3. Start the Application Server Control Console:

   ```
   emctl start iasconsole
   ```

   Alternatively, on Windows, you can start the Application Server Control Console using the Services control panel. See Section A.1.2, "Starting and Stopping the Application Server Control Console on Windows" for more information.

Alternatively, on Windows, you can start the middle tier from the Programs menu: **Start** > **Programs** > **Oracle Application Server** - *Oracle_Home* > **Start** > ***instanceName***.

3.2.4 Stopping a Middle-Tier Instance

This section describes how to stop all processes in a middle-tier instance. Follow this procedure when you are preparing to shut down your host, or any other time you want to stop the entire instance.

This procedure applies to all middle-tier instance types:

- J2EE and Web Cache
- Portal and Wireless
- Business Intelligence and Forms

To stop a middle-tier instance:

1. Stop the Application Server Control Console:

   ```
   emctl stop iasconsole
   ```

   Alternatively, on Windows, you can stop the Application Server Control Console using the Services control panel. See Section A.1.2, "Starting and Stopping the Application Server Control Console on Windows" for more information.

2. Stop components:

   ```
   opmnctl stopall
   ```

   This command stops OPMN and all OPMN-managed processes such as DCM, Oracle HTTP Server, OC4J instances, and OracleAS Web Cache, OracleAS Forms Services, and OracleAS Reports Services.

Alternatively, on Windows, you can stop the middle tier from the Programs menu: **Start** > **Programs** > **Oracle Application Server** - ***Oracle_Home*** > **Stop** > ***instanceName***.

3.3 Starting and Stopping Components

You can use the following tools to start, stop, restart, and view the status of components:

- `opmnctl`: A command-line tool. See Section 3.3.1.
- Application Server Control Console: A Web-based tool. See Section 3.3.2.

These tools are completely compatible—they both use OPMN as their underlying technology for managing processes—and can be used interchangeably. For example, you can start a component using `opmnctl` and stop it using the Application Server Control Console.

Although the two tools can be used interchangeably, they offer different features. With the `opmnctl` command, you can start and stop sub-processes within components, as well as the entire component. For example, you can start and stop OracleAS Web Cache, or you can start and stop only the OracleAS Web Cache admin sub-process. With the Application Server Control Console, you can view components that cannot be started or stopped, but whose status depends on other components. For example, the Application Server Control Console displays the status of the Single Sign-On component, whose status depends on the HTTP_Server.

3.3.1 Starting and Stopping Components Using opmnctl

You can use the `opmnctl` command line tool to start and stop components. It is located in the following directory:

```
(UNIX) ORACLE_HOME/opmn/bin
(Windows) ORACLE_HOME\opmn\bin
```

To start, stop, or restart a component using opmnctl:

```
opmnctl stopproc ias-component=component
opmnctl startproc ias-component=component
opmnctl restartproc ias-component=component
```

To start, stop, or restart the sub-process of a component:

```
opmnctl stopproc process-type=process
opmnctl startproc process-type=process
opmnctl restartproc process-type=process
```

To view the status of components and processes:

```
opmnctl status
```

To learn more about using opmnctl, refer to *Oracle Process Manager and Notification Server Administrator's Guide*.

3.3.2 Starting and Stopping Components Using Application Server Control Console

You can start, stop, restart, and view status of components on the Application Server home page:

1. Navigate to the Application Server home page on the Application Server Control Console. Scroll to the System Components section.
2. In the **Select** column, select the components you want to start, stop, or restart.
3. Click the **Start, Stop,** or **Restart** button on the top right of the System Components section.

You can also start and stop individual components on each component home page.

3.4 Enabling and Disabling Components

When you disable a component, you prevent it from starting when you start the application server instance, and you remove it from the list of System Components displayed on the Application Server home page.

When you enable a component, you allow it to start when you start the application server instance, and it appears in the list of System Components displayed on the Application Server Control Console.

You can enable and disable components using the Application Server Control Console. On the Application Server Home page, click **Enable/Disable Components.**

From the resulting page, you can select which components to enable or disable. Notice that components that are dependent on each other are grouped, and are enabled or disabled together.

When you enable or disable components, consider the following restrictions and additional information:

- If you use the backup and recovery procedures documented in this book, you must run `bkp_restore.pl -m config` after you enable or disable components so the proper components are registered with the OracleAS Backup and Recovery Tool. See Chapter 19 for more information.

- You cannot disable or enable components that are part of an Oracle Application Server Cluster. As a result, the **Enable/Disable Components** button is not available on the Application Server Home page when you are managing an instance that belongs to an OracleAS Cluster.

  See Also:

  - *Distributed Configuration Management Administrator's Guide* for information about using Distributed Configuration Management to create and manage an OracleAS Cluster
  - "About Managing OracleAS Clusters" in the Application Server Control Console online Help

3.5 Starting and Stopping an Oracle Application Server Environment

This section provides procedures for starting and stopping an Oracle Application Server environment. An environment can consist of multiple OracleAS Infrastructure and middle-tier instances distributed across multiple hosts. These instances are dependent on each other and it is important to start and stop them in the proper order.

You can follow these procedures when you need to completely shut down your Oracle Application Server environment. For example, when preparing to perform a complete backup of your environment, or apply a patch.

3.5.1 Starting an Oracle Application Server Environment

To start an Oracle Application Server environment:

1. Start any OracleAS Infrastructure that contains only OracleAS Metadata Repository.

   If your environment has OracleAS Infrastructure installations that contain only OracleAS Metadata Repository, start those in any order. Note that for these installation types, you only need to start OracleAS Metadata Repository. You do not need to start any processes with `opmnctl` and you do not need to start the Application Server Control Console. See Section 3.2.1 for more information.

2. Start the OracleAS Infrastructure that contains Oracle Identity Management.

   If your environment uses Oracle Identity Management, start the OracleAS Infrastructure that contains Oracle Internet Directory. If this OracleAS Infrastructure contains OracleAS Metadata Repository, start that before you start Oracle Internet Directory. See Section 3.2.1 for more information.

3. Start OracleAS Clusters.

   If your environment has middle-tier instances that are part of an OracleAS Cluster, start the OracleAS Clusters in any order.

   See Also: *Oracle Application Server High Availability Guide*

4. Start middle-tier instances.

   If your environment contains middle-tier instances that are not part of an OracleAS Cluster, start them in any order. See Section 3.2.3 for more information.

3.5.2 Stopping an Oracle Application Server Environment

To stop all processes in an Oracle Application Server environment:

1. Stop OracleAS Clusters.

   If your environment has middle-tier instances that are part of OracleAS Clusters, stop the clusters in any order.

   See Also: *Oracle Application Server High Availability Guide*

2. Stop middle-tier instances.

   If your environment contains middle-tier instances that are not part of an OracleAS Cluster, stop them in any order. See Section 3.2.4 for more information.

3. Stop the OracleAS Infrastructure that contains Oracle Identity Management.

   If your environment uses Oracle Identity Management, stop the OracleAS Infrastructure that contains Oracle Internet Directory. If this OracleAS Infrastructure contains OracleAS Metadata Repository, stop that as well. See Section 3.2.2.

4. Stop any OracleAS Infrastructure instances that contain only OracleAS Metadata Repository as described in Section 3.2.2.

   If your environment has OracleAS Infrastructure installations that contain only OracleAS Metadata Repository, stop those in any order.

3.6 Starting and Stopping: Special Topics

This section contains the following special topics about starting and stopping Oracle Application Server:

- Starting and Stopping Log Loader
- Starting and Stopping in High Availability Environments
- Resolving OC4J Errors When Starting Multiple Instances
- Forcing a Shut Down of OracleAS Metadata Repository

3.6.1 Starting and Stopping Log Loader

The method for starting and stopping Oracle Application Server Log Loader is different from other components.

Log Loader is not started when you issue the `opmnctl startall` command or when you perform a **Start All** operation in the Application Server Control Console. You can start Log Loader in the following ways:

- Using the following command:

  ```
  (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=LogLoader
  (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=LogLoader
  ```

- By clicking **Start** on the Log Loader page in the Application Server Control Console. See Section 5.5.1, "Starting and Stopping Log Loader" for more information.

Log Loader is stopped when you issue the `opmnctl stopall` command; however it is not stopped when you issue a **Stop All** operation in the Application Server Control Console. In the latter case, you can stop Log Loader in the following ways:

- Using the following command:

  ```
  (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=LogLoader
  ```

```
(Windows) ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=LogLoader
```

- By clicking **Stop** on the Log Loader page in the Application Server Control Console. See Section 5.5.1, "Starting and Stopping Log Loader" for more information.

3.6.2 Starting and Stopping in High Availability Environments

There are special considerations and procedures for starting and stopping High Availability environments such as:

- DCM-Managed Oracle Application Server Cluster
- Manually Managed Oracle Application Server Cluster
- Oracle Application Server Cold Failover Cluster
- Oracle Application Server Disaster Recovery (includes starting and stopping the DSA component)

> **See:** *Oracle Application Server High Availability Guide* for information about starting and stopping in high-availability environments

3.6.3 Resolving OC4J Errors When Starting Multiple Instances

If you have multiple Oracle Application Server installations on one host and you start them at the same time (for example, to start an OracleAS Cluster), OPMN may return an error like the following:

```
<process-type id="my_OC4J_instance">
  <process-set id="default_island">
    <process id="93388820" pid="24711" status="Stopped" index="1"
     log="/disk1/oracleas/opmn/logs/OC4J~my_OC4J_instance~default_island~1"
 operation="request" result="failure">
      <msg code="-21" text="failed to restart a managed process
        after the maximum retry limit">
      </msg>
```

This error indicates that an OC4J instance (`my_OC4J_instance`) failed to start. The problem could be caused by two different Oracle homes on the same host using the same port ranges for RMI, JMS, and AJP ports, and an OC4J instance in one Oracle home trying to use the same port as an OC4J instance in another Oracle home.

For example, assume you have two Oracle Application Server installations on one host that reside in `ORACLE_HOME1` and `ORACLE_HOME2`. Each installation contains one or more OC4J instances, and each OC4J instance is assigned a port range for AJP, RMI, and JMS ports.

You can check OC4J port range assignments by examining the `opmn.xml` file in both Oracle homes:

```
ORACLE_HOME1/opmn/conf/opmn.xml
ORACLE_HOME2/opmn/conf/opmn.xml
```

In each file, locate the OC4J instance entries, which start with a line like the following:

```
<process-type id="home" module-id="OC4J" ... >
```

Within each entry, locate the RMI, JMS, and AJP port ranges, which looks like this:

```
<port id="ajp" range="12501-12600"/>
```

```
<port id="rmi" range="12401-12500"/>
<port id="jms" range="12601-12700"/>
```

Table 3–1 illustrates the problem of having the same OC4J port assignments in two Oracle homes—the AJP, RMI, and JMS port ranges in `ORACLE_HOME1` are identical to the AJP, RMI, and JMS port ranges in `ORACLE_HOME2`. (Note that this example only lists the relevant lines from the `opmn.xml`.)

Table 3–1 Example of Identical Port Ranges in Two Oracle Homes

OC4J Port Ranges in *ORACLE_HOME1*/opmn/conf/opmn.xml	**OC4J Port Ranges in *ORACLE_HOME2*/opmn/conf/opmn.xml**

`<ias-component id="OC4J">`
`...`
`<process-type id="home" ... >`
`...`
`<port id="ajp" range="12501-12600"/>`
`<port id="rmi" range="12401-12500"/>`
`<port id="jms" range="12601-12700"/>`
`...`
`</process-type>`
`<process-type id="OC4J_SECURITY" ... >`
`...`
`<port id="ajp" range="12501-12600"/>`
`<port id="rmi" range="12401-12500"/>`
`<port id="jms" range="12601-12700"/>`
`</process-type>`
	`<ias-component id="OC4J">`
	`...`
	`<process-type id="home" ... >`
	`...`
	`<port id="ajp" range="12501-12600"/>`
	`<port id="rmi" range="12401-12500"/>`
	`<port id="jms" range="12601-12700"/>`
	`...`
	`</process-type>`
	`<process-type id="OC4J_SECURITY" ... >`
	`...`
	`<port id="ajp" range="12501-12600"/>`
	`<port id="rmi" range="12401-12500"/>`
	`<port id="jms" range="12601-12700"/>`
	`</process-type>`

Port allocation for all OC4J instances within an Oracle Application Server instance is controlled by OPMN. So, having overlapping port ranges within a single `opmn.xml` file is not a problem. However, when two OPMNs on a host start processes at the same time, there is no coordination between them on port usage.

The algorithm OPMN uses to assign a port is:

1. Choose a port from the port range that is not currently marked as allocated to any processes managed by the OPMN in the local instance.
2. Before assigning the port, check to see if the port is in use by binding to it.
3. If the port is not in use (that is, OPMN could bind to it), then unbind and assign the port to a process (such as an OC4J instance) so it can bind to it, updating internal data structures with this assignment information.

In between the time that OPMN unbinds from the port and the assigned process binds to the port, it is possible for another process to bind to the port. This could be another OPMN on the host, or any other process that happens to try to bind to the same port number.

If your port range assignments are the same across Oracle homes, and you received the error shown at the beginning of this section, then probably two OPMN processes tried to bind the same port for their OC4J instances. There is no way to eliminate this problem completely (because there is a rare chance that a non-OPMN process could try to bind to the port at the same time) but you can reconfigure OPMN to reduce the chance of encountering it.

There are two options for addressing this problem:

- Option 1: Assign Unique Port Ranges to Each Oracle Home
- Option 2: Increase the Maximum Number of Retries for Starting OC4J Instances

Option 1: Assign Unique Port Ranges to Each Oracle Home

You can assign unique OC4J port ranges to each Oracle home, as shown in Table 3–2. Then, the OPMN in ORACLE_HOME1 and the OPMN in ORACLE_HOME2 will not attempt to use the same port numbers when assigning OPMN ports, and will not attempt to bind to the same port.

Table 3–2 Example of Using Unique Port Ranges in Two Oracle Homes

OC4J Port Ranges in *ORACLE_HOME1*/opmn/conf/opmn.xml	**OC4J Port Ranges in *ORACLE_HOME2*/opmn/conf/opmn.xml**
<ias-component id="OC4J"> ... <process-type id="home" ... > ... <port id="ajp" range="**12501-12600**"/> <port id="rmi" range="**12401-12500**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_SECURITY" ... > ... <port id="ajp" range="**12501-12600**"/> <port id="rmi" range="**12401-12500**"/> <port id="jms" range="**12601-12700**"/> </process-type>	<ias-component id="OC4J"> ... <process-type id="home" ... > ... <port id="ajp" range="**4601-4700**"/> <port id="rmi" range="**4701-4800**"/> <port id="jms" range="**4801-4900**"/> ... </process-type> <process-type id="OC4J_SECURITY" ... > ... <port id="ajp" range="**4601-4700**"/> <port id="rmi" range="**4701-4800**"/> <port id="jms" range="**4801-4900**"/> </process-type>

To do this:

1. Choose unique port ranges for AJP, RMI, and JMS.
2. Edit *ORACLE_HOME2*/opmn/conf/opmn.xml.
3. For each OC4J instance in the file, change AJP, RMI, and JMS to use the new unique port ranges. For example:

   ```
   <port id="ajp" range="4601-4700"/>
   <port id="rmi" range="4701-4800"/>
   <port id="jms" range="4801-4900"/>
   ```

4. Save and close the file.
5. Reload OPMN:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl reload
   (Windows) ORACLE_HOME\opmn\bin\opmnctl reload
   ```

Option 2: Increase the Maximum Number of Retries for Starting OC4J Instances

OPMN attempts to start processes a certain number of times before declaring failure. For process types with port ranges, if the failure to start the process is due to the process not being able bind to the assigned port number, OPMN attempts to start the process with a different port number in the specified range. You can have identical port ranges in two Oracle homes, and increase the number of times OPMN attempts to restart a process, so eventually OPMN will choose a port that works. This does not completely eliminate the problem, because there is a chance that OPMN will not find a port that works in 10 tries, but it does reduce the chance of encountering the problem.

The parameter that controls the number of retries is retry. The default value is 2. You can increase the parameter to a higher number, for example, 10, by following these steps in each Oracle home:

1. Edit *ORACLE_HOME*/opmn/conf/opmn.xml.
2. For each OC4J instance in the file, increase the retry value for start and restart. For example:

```
<start timeout="600" retry="10"/>
<restart timeout="720" retry="10"/>
```

3. Save and close the file.
4. Reload OPMN:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl reload
(Windows) ORACLE_HOME\opmn\bin\opmnctl reload
```

Table 3–3 shows an example of the opmn.xml file in two Oracle homes on the same host after the retry count has been increased to 10.

Table 3–3 Example of Increasing the Retry Count in Two Oracle Homes

OC4J Port Ranges in *ORACLE_HOME1*/opmn/conf/opmn.xml	**OC4J Port Ranges in *ORACLE_HOME2*/opmn/conf/opmn.xml**
<ias-component id="OC4J"> ... <process-type id="home" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="ajp" range="**12501-12600**"/> <port id="rmi" range="**12401-12500**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_SECURITY" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="ajp" range="**12501-12600**"/> <port id="rmi" range="**12401-12500**"/> <port id="jms" range="**12601-12700**"/> </process-type>	<ias-component id="OC4J"> ... <process-type id="home" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="ajp" range="**12501-12600**"/> <port id="rmi" range="**12401-12500**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_SECURITY" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="ajp" range="**12501-12600**"/> <port id="rmi" range="**12401-12500**"/> <port id="jms" range="**12601-12700**"/> </process-type>

3.6.4 Forcing a Shut Down of OracleAS Metadata Repository

If you find that the OracleAS Metadata Repository instance is taking a long time to shut down, you can use the following command to force an immediate shutdown:

```
SQL> SHUTDOWN IMMEDIATE;
```

Immediate database shutdown proceeds with the following conditions:

- No new connections are allowed, nor are new transactions allowed to be started, after the statement is issued.
- Any uncommitted transactions are rolled back. (If long uncommitted transactions exist, this method of shutdown might not complete quickly, despite its name.)
- Oracle does not wait for users currently connected to the database to disconnect. Oracle implicitly rolls back active transactions and disconnects all connected users.

The next startup of the database will not require any instance recovery procedures.

See Also: *Oracle Database Administrator's Guide* in the Oracle Database 10*g* documentation library

Part II

Basic Administration

This part describes basic administration tasks.

It contains the following chapters:

- Chapter 4, "Managing Ports"
- Chapter 5, "Managing Log Files"
- Chapter 6, "Managing an OracleAS Metadata Repository"

4

Managing Ports

This chapter describes how to view and change Oracle Application Server port numbers. It contains the following topics:

4.1 About Managing Ports

Many Oracle Application Server components and services use ports. As an administrator, it is important to know the port numbers used by these services, and to ensure that the same port number is not used by two services on your host.

Most port numbers are assigned during installation. Every component and service has an allotted port range, which is the set of port numbers Oracle Application Server attempts to use when assigning a port. Oracle Application Server starts with the lowest number in the range and performs the following checks:

- Is the port used by another Oracle Application Server installation on the host?

  The installation may be up or down at the time; Oracle Application Server can still detect if the port is used.

- Is the port used by a process that is currently running?

  This could be any process on the host, even a non-Oracle Application Server process.

If the answer to any of the preceding questions is yes, Oracle Application Server moves to the next highest port in the allotted port range and continues checking until it finds a free port.

You can override this behavior for some ports, and specify a port number assignment during installation. To do this, you edit a template file called `staticports.ini`, and launch Oracle Universal Installer with special options.

> **See Also:** Appendix D for a complete list of allotted port ranges. Refer to *Oracle Application Server Installation Guide* for directions on overriding port assignments during installation with `staticports.ini`.

4.2 Viewing Port Numbers

You can view port numbers on the Application Server Control Console Ports page. Click the **Ports** tab on the Application Server Home page. The Ports page displays the current port numbers and is updated any time you change a port number. For selected components, it also provides links to pages that allow you to change port numbers.

Note: Immediately after installation, you can view port number assignments in:

`(UNIX) ORACLE_HOME/install/portlist.ini`
`(Windows) ORACLE_HOME\install\portlist.ini`

If you change a port number, it is not updated in this file, so you can only rely on this file immediately after installation. In addition, this file is not valid after you upgrade Oracle Application Server. Use Application Server Control Console to view the port numbers.

4.3 Changing Middle-Tier Ports

This section provides complete instructions for changing port numbers in middle-tier instances. The instructions explain how to change the port number, and update any other components that might be affected.

See Also: Appendix D for more information on port numbers

Note: You can change a port number to any number you want, as long as it is an unused port. You do not have to use a port in the allotted port range for the component.

This section contains the following topics:

- Changing Oracle Enterprise Manager Ports
- Changing OC4J Ports
- Changing the Oracle HTTP Server Listen Ports
- Changing the Oracle HTTP Server Diagnostic Port
- Changing OracleAS Web Cache Ports
- Changing the DCM Discovery Port
- Changing the Java Object Cache Port
- Changing the Log Loader Port
- Changing OPMN Ports (ONS Local, Request, and Remote)
- Changing the Port Tunneling Port
- Changing the OracleAS Portal Port
- Changing the OracleAS Wireless Port
- Changing OracleBI Discoverer Ports
- Changing the OracleAS Forms Services Port
- Changing OracleAS Reports Services Ports

4.3.1 Changing Oracle Enterprise Manager Ports

After you have installed Oracle Application Server, you can change the following Oracle Enterprise Manager 10*g* ports associated with your Oracle Application Server instance:

- The Oracle Management Agent port, which is used for communications with the Management Agent
- The Application Server Control Console port, which is used in the Application Server Control Console URL. For example, on UNIX:

  ```
  http://appserver1.acme.com:1156
  ```

- The Oracle Application Server Containers for J2EE (OC4J) Remote Method Invocation (RMI) port, which is used by the Application Server Control OC4J instance

To view the current port values for these components, as well as the valid port number range for each component, navigate to the Ports page from the Application Server Home page for the instance.

However, you cannot modify the Enterprise Manager port numbers from the Ports page. Instead, use the following procedure to change the Application Server Control ports:

1. Change directory to the `bin` directory in the Oracle Application Server Oracle home.
2. Stop the Application Server Control Console.

   On UNIX systems, enter the following command:

   ```
   ORACLE_HOME/bin/emctl stop iasconsole
   ```

   On Windows systems, use the Services control panel to stop the Application Server Control service.
3. Use the following command to change one of the Enterprise Manager port values:

   ```
   (UNIX) ORACLE_HOME/bin/emctl config {agent port | iasconsole {port | rmiport}}
   port_number
   (Windows) ORACLE_HOME\bin\emctl config {agent port | iasconsole {port |
   rmiport}} port_number
   ```

   For example, to change the port used by the Application Server Control Console on UNIX:

   ```
   ORACLE_HOME/bin/emctl config iasconsole port 1812
   ```

4. Start Application Server Control.

   On UNIX systems, enter the following command:

   ```
   ORACLE_HOME/bin/emctl start iasconsole
   ```

   On Windows systems, use the Services control panel to start the Application Server Control service.

Table 4–1 describes the configuration changes that are automatically performed when you use the `emctl config` command to change an Application Server Control port number.

Table 4–1 Changing Application Server Control Ports Using the emctl Command Line

Port	Command Line	Actions Performed
Application Server Control port	`emctl config iasconsole port` *port_number*	Changes the port value assigned to the StandaloneConsoleURL property in following configuration file: (UNIX) *ORACLE_HOME*/sysman/emd/targets.xml (Windows) *ORACLE_HOME*\sysman\emd\targets.xml Changes the port value assigned to the `web-site` tag in the following configuration file: (UNIX) *ORACLE_HOME*/sysman/j2ee/config/emd-web-site.xml (Windows)*ORACLE_HOME*\sysman\j2ee\config\emd-web-site.xml
Oracle Management Agent port	`emctl config agent port` *port_number*	Changes the value assigned to the EMD_URL property in the following configuration file: (UNIX) *ORACLE_HOME*/sysman/config/emd.properties (Windows) *ORACLE_HOME*\sysman\config\emd.properties
OC4J Remote Method Invocation (RMI) port	`emctl config iasconsole rmiport` *port_number*	Changes the port values in the following configuration files: (UNIX) *ORACLE_HOME*/sysman/j2ee/config/rmi.xml (Windows) *ORACLE_HOME*\sysman\j2ee\config\rmi.xml (UNIX) *ORACLE_HOME*/bin/emctl.pl (Windows) *ORACLE_HOME*\bin\emctl.pl

4.3.2 Changing OC4J Ports

This section describes how to change the following OC4J port numbers:

- AJP
- JMS
- RMI
- IIOP
- IIOPS1 (Server only)
- IIOPS2 (Server and client)

By default, Oracle Application Server does not use a single port number for each type of OC4J port. Instead, it uses a port range for each type of OC4J port and that range is the same for all OC4J instances on the host. During runtime, each OC4J instance on the host is assigned a single free port from the range. For example, if the default AJP range for every OC4J instance on a host is 12501-12600, then each OC4J instance is assigned a single free port from that range for its AJP port.

When changing an OC4J port number, you typically specify a new port range. The range may be a simple port range (12501-12600), a comma separated list of ports (12501, 12504, 12507), or a combination of both (12501-12580, 12583, 12590-12600). By default, the ranges contain 100 ports. If you specify a range that is too narrow, you may encounter problems when starting OC4J instances. The AJP and RMI port ranges are required; the others are optional.

Note: Note that because the IIOP, IIOPS1, and IIOPS2 ports are not configured by default, they may not be listed in the Ports page of Application Server Control Console or in opmn.xml. To configure them, you must manually add them to the opmn.xml file.

See the J2EE Interoperability chapter of the *Oracle Application Server Containers for J2EE Services Guide* for more information.

You can change OC4J port ranges using the Application Server Control Console or manual steps:

- Using the Application Server Control Console:
  1. Navigate to the Application Server instance Home page.
  2. Click **Ports.**
  3. On the Ports page, locate the OC4J Instance and OC4J port range you want to change. Click the icon in the **Configure** column.
  4. On the Server Properties page, enter the new port range in the appropriate field. Click **Apply.**
  5. On the Confirmation page, click **Yes,** you want to restart now.
- Using manual steps:
  1. Open the `opmn.xml` file:

     ```
     (UNIX) ORACLE_HOME/opmn/conf/opmn.xml
     (Windows) ORACLE_HOME\opmn\conf\opmn.xml
     ```

  2. Locate the element for the OC4J instance that contains the port range you want to change. For example, if you want to change a port range for the `home` instance, locate this element:

     ```
     <process-type id="home" ...>
     ```

  3. Within the OC4J instance element, there is a `port` element for each type of port. For example:

     ```
     <port id="ajp" range="12501-12600"/>
     <port id="rmi" range="12401-12500"/>
     <port id="jms" range="12601-12700"/>
     <port id="iiop" range="13301-13400"/>
     <port id="iiops1" range="13401-13500"/>
     <port id="iiops2" range="13501-13600"/>
     ```

  4. Modify the range parameter for the port you want to change, and then save the file.
  5. Reload OPMN:

     ```
     (UNIX) ORACLE_HOME/opmn/bin/opmnctl reload
     (Windows) ORACLE_HOME\opmn\bin\opmnctl reload
     ```

  6. Start the OC4J instance that contains the port number you changed:

     ```
     (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc process-type=OC4J_instance
     (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc process-type=OC4J_instance
     ```

     For example, if you changed a port number in the `home` instance on UNIX:

     ```
     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home
     ```

  7. Run the following command:

     ```
     (UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig
     (Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig
     ```

4.3.3 Changing the Oracle HTTP Server Listen Ports

To change the Oracle HTTP Server Listen ports, you change the Oracle HTTP Server Listen directive. When you do this, there are often dependencies that must also be set. For example, if you are using OracleAS Web Cache to improve the performance of your Oracle Application Server instance, you must modify the OracleAS Web Cache origin server settings whenever you modify the Oracle HTTP Server listen ports.

To be sure the port dependencies are modified correctly, you can use a single command to change the Oracle HTTP Server listen port. The `portconfig` command automatically modifies the necessary configuration files within the Oracle home and optionally restarts the required components within the Oracle home.

The following topics describe how to define the `portconfig` command and then use it to modify the Oracle HTTP Server HTTP or HTTPS listen port:

- Task 1: Enable Oracle HTTP Server to Run as Root for Ports Set to Less Than 1024 (Unix Only)
- Task 2: Use the portconfig Command to Change the Oracle HTTP Server Listen Ports
- Task 3: Update the portlist.ini File
- Task 4: Restart Oracle HTTP Server

Task 1: Enable Oracle HTTP Server to Run as Root for Ports Set to Less Than 1024 (Unix Only)

If you are on a UNIX system and you are changing the Listen port to a number less than 1024, perform these steps before you change the Oracle HTTP Server Listen port.

By default, Oracle HTTP Server runs as a non-root user (the user that installed Oracle Application Server). On UNIX systems, if you change the Oracle HTTP Server Listen port number to a value less than 1024, you must enable Oracle HTTP Server to run as root, as follows:

1. Log in as root.
2. Run the following commands in the middle-tier Oracle home:

```
cd ORACLE_HOME/Apache/Apache/bin
chown root .apachectl
chmod 6750 .apachectl
```

Task 2: Use the portconfig Command to Change the Oracle HTTP Server Listen Ports

Use the following procedure to change the Oracle HTTP Server HTTP or HTTPS listen port:

1. Set the ORACLE_HOME environment variable to the home directory of the Oracle Application Server instance where the Oracle HTTP Server resides.

   For example:

```
(UNIX) setenv ORACLE_HOME /dev0/private/oracle/appserv1/
(Windows) set ORACLE_HOME=D:\oracle\appserv1\
```

2. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.

3. Create an alias (on UNIX systems) or a DOSKEY macro (on Windows systems) to represent the `portconfig` command.

   For example, to execute the command as an alias on UNIX systems, enter the following command:

```
alias portconfig '$ORACLE_HOME/jdk/bin/java -cp
$ORACLE_HOME/sysman/webapps/emd/WEB-INF/lib/emd.jar:
$ORACLE_HOME/dcm/lib/dcm.jar:
$ORACLE_HOME/sso/lib/ossoreg.jar
oracle.sysman.ias.sta.tools.PortConfigCmdLine \!*'
```

   Similarly, to execute the command as DOSKEY macro on Windows systems, enter the following at the DOS command line:

```
doskey portconfig=%ORACLE_HOME%\jdk\bin\java -cp
%ORACLE_HOME%/sysman/webapps/emd/WEB-INF/lib/emd.jar;
%ORACLE_HOME%/dcm/lib/dcm.jar;
%ORACLE_HOME%/sso/lib/ossoreg.jar
oracle.sysman.ias.sta.tools.PortConfigCmdLine $*
```

4. Use the newly created `portconfig` command as follows:

```
portconfig -oracleHome ORACLE_HOME
-oldPort old_port
-newPort new_port
[-sso -url http://sso_host:port -user http_server_admin_user
  [-site name_of_sso_partner_application]
  [-admin mod_osso_admin_user]
  [-vHost path_to_mod_osso_configuration_file]]
[-webCache] [-debug]
{-start | -restart}
```

   For example, on UNIX systems:

```
portconfig -oracleHome $ORACLE_HOME -oldPort 7777 -newPort 7778 -webCache
```

   For example, on Windows systems:

```
portconfig -oracleHome %ORACLE_HOME% -oldPort 80 -newPort 7778 -webCache
```

Table 4–2 describes the arguments available when you use the `portconfig` command to automatically change the Oracle HTTP Server Listen port.

Table 4–2 Arguments for the portconfig Command

Argument	Description
`-oracleHome`	The Oracle home of the Oracle Application Server instance. The `portconfig` command modifies only components that are part of the selected Oracle home. You can use an environment variable to represent the Oracle home.
`-oldPort`	The old (current) value of the Oracle HTTP Server Listen port.
`-newPort`	The new value for the Oracle HTTP Server Listen port.
`-webCache`	Use this optional argument if you are using OracleAS Web Cache to improve the performance and reliability of your Web server. When this argument is included on the command line, the dependent OracleAS Web Cache port assignment is changed automatically. Specifically, the port number of the origin server is updated automatically so that it points to the new Oracle HTTP Server listen port. **Note:** The `portconfig` command updates the OracleAS Web Cache instance only if it resides in the current Oracle home.

Table 4–2 (Cont.) Arguments for the portconfig Command

Argument	Description
`-start`	Use this optional argument to stop and start the application server instance after the `portconfig` command performs the configuration changes. The Oracle Application Server instance must be stopped and started—or restarted—before the port changes take effect. Note that during startup, all enabled components of the application server are started, even those that were originally down before you ran the `portconfig` command to change the Oracle HTTP Server Listen port. Compare with the `-restart` argument. Each time you run the command you can use the `-restart` or `-start` options, but not both.
`-restart`	Use this optional argument to stop and start the application server instance after the `portconfig` command performs the configuration changes. The Oracle Application Server instance must be restarted—or stopped and started—before the port changes take effect. With this option, only already running components are restarted after the configuration changes are complete. Components that were down before you ran the `portconfig` command to change the Oracle HTTP Server Listen port will remain down. Compare with the `-start` argument. Each time you run the command you can use the `-restart` or `-start` options, but not both.
`-debug`	Use this optional argument to display debugging information as the command executes. This argument can be useful if you are troubleshooting a problem or working with Oracle Support.
`-sso`	Use this optional argument when the Listen port you are changing is protected by OracleAS Single Sign-On. The `portconfig` command re-registers `mod_osso` with the new Oracle HTTP Server Listen port value. When you use this argument, you must include the `-url` and `-user` arguments. In addition, you can optionally use the `-site`, `-admin`, and `-vHost` arguments. For more information about registering mod_osso, see "Configuring and Administering Partner Applications" in the *Oracle Application Server Single Sign-On Administrator's Guide*.
`-url`	This argument is required when you use the `-sso` argument. Use this argument to provide the new Oracle HTTP Server URL, which is also used by OracleAS Single Sign-On and uses the new Listen port. For example: `http://sso42.acme.com:7778` This URL is passed as the `-mod_osso_url` parameter in the `ssoreg.sh` and `ssoreg.bat` scripts.
`-user`	This argument is required when you use the `-sso` argument. Use this argument to enter the name of the account that is used to start Oracle HTTP Server. On UNIX systems, this is usually `root`. On Windows, it is usually `SYSTEM`. The value provided with this argument is passed as the `-u` parameter in the `ssoreg.sh` and `ssoreg.bat` scripts.
`-site`	This argument is optional; however, you can use it only when you use the `-sso` argument. Use this argument to enter the site name of OracleAS Single Sign-On partner application. The site name is displayed by the OracleAS Single Sign-On administration pages. The value of this argument is passed as the `-site_name` parameter in the `ssoreg.sh` and `ssoreg.bat` scripts. If the `-site` argument is not specified, the application server instance name is passed to `ssoreg.sh` and `ssoreg.bat` scripts as the value of the `-site_name` parameter.

Table 4–2 (Cont.) Arguments for the portconfig Command

Argument	Description
`-admin`	This argument is optional; however, you can use it only when you use the `-sso` argument. Use this argument to enter the account name of the `mod_osso` administrator. This value is displayed in the OracleAS Single Sign-On administration pages. In most cases, this value should be the same as the distinguished name (dn) of the user who installed Oracle Application Server. The value of this argument is passed as the `-admin_info` parameter in the `ssoreg.sh` and `ssoreg.bat` scripts.
`-vHost`	This argument is optional; however, you can use it only when you use the `-sso` argument. Use this argument to enter the path to the `osso.conf` file for the virtual host being configured. For example: `ORACLE_HOME/Apache/Apache/conf/osso/`*`vh_name`*`/osso.conf` Use this argument only when you are registering an HTTP virtual host with the OracleAS Single Sign-On server. The value of this argument is passed as the `-config_file` parameter, along with the `-virtualhost` parameter, in the `ssoreg.sh` and `ssoreg.bat` scripts.

Task 3: Update the portlist.ini File

After you change the Oracle HTTP Server Listen Port, you should update `portlist.ini` with the new port number. This will avoid potential problems if you later associate the middle tier with an OracleAS Infrastructure or change the OracleAS Infrastructure associated with the middle tier. The `portlist.ini` file is located in the following directory:

```
(UNIX) ORACLE_HOME/install
(Windows) ORACLE_HOME\install
```

Task 4: Restart Oracle HTTP Server

Restart the application server instance:

- On UNIX systems:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

- On Windows systems:

```
ORACLE_HOME\bin\emctl stop iasconsole
ORACLE_HOME\opmn\bin\opmnctl stopall
ORACLE_HOME\opmn\bin\opmnctl startall
ORACLE_HOME\bin\emctl start iasconsole
```

4.3.4 Changing the Oracle HTTP Server Diagnostic Port

To change the Oracle HTTP Server Diagnostics port number in any installation type:

1. Open the `dms.conf` file:

```
(UNIX) ORACLE_HOME/Apache/Apache/conf/dms.conf
(Windows) ORACLE_HOME\Apache\Apache\conf\dms.conf
```

2. Change the old port number to the new port number everywhere it appears in the file, and then save the file. This update includes the Listen directive, OpmnHostPort directive, Redirect directive, and the VirtualHost.
3. Restart Oracle HTTP Server:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server

(Windows) ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=HTTP_Server
(Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=HTTP_Server
```

4.3.5 Changing OracleAS Web Cache Ports

The following sections describe how to change the OracleAS Web Cache ports:

- Changing the OracleAS Web Cache Listen Ports
- Changing the OracleAS Web Cache Administration Port
- Changing the OracleAS Web Cache Invalidation Port
- Changing the OracleAS Web Cache Statistics Port

4.3.5.1 Changing the OracleAS Web Cache Listen Ports

This section describes how to change the OracleAS Web Cache HTTP or HTTPS listen port. It involves changing the OracleAS Web Cache port number and updating other components in the middle tier with the new port number. The tasks involved are:

- Task 1: Enable OracleAS Web Cache to Run as Root for Ports Less Than 1024 (UNIX Only)
- Task 2: Change the OracleAS Web Cache Listen Port
- Task 3: Change the OracleAS Web Cache Logical Site Port
- Task 4: Update the Oracle HTTP Server Port Directive
- Task 5: Update the Application Server Control Console
- Task 6: Update mod_osso
- Task 7: Update OracleAS Portal Configuration
- Task 8: Update Web Providers
- Task 9: Update OracleAS Wireless
- Task 10: Update OracleBI Discoverer
- Task 11: Update OracleAS Reports Services
- Task 12: Restart the Middle-Tier Instance

Task 1: Enable OracleAS Web Cache to Run as Root for Ports Less Than 1024 (UNIX Only)

Perform this task only if you are changing the port to a number less than 1024.

By default, OracleAS Web Cache runs as a non-root user (the user that installed Oracle Application Server). On UNIX systems, if you change the OracleAS Web Cache listen port number to a value less than 1024, you must enable OracleAS Web Cache to run as root, as follows:

1. Log in as the user that installed Oracle Application Server and stop OracleAS Web Cache:

```
ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=WebCache
```

2. Log in as root.
3. Run the following command in the middle-tier Oracle home:

   ```
   ORACLE_HOME/webcache/bin/webcache_setuser.sh setroot user_ID
   ```

   The parameter *user_ID* is the user ID associated with the OracleAS Web Cache processes. This is usually the user that installed Oracle Application Server. This user is listed on the Security page (**Web Cache Home** -> **Administration** tab -> **Security**) of Application Server Control Console.
4. Log in as the user that installed Oracle Application Server and start OracleAS Web Cache:

   ```
   ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=WebCache
   ```

Task 2: Change the OracleAS Web Cache Listen Port

Take the following steps:

1. Using the Application Server Control Console, navigate to the Web Cache Home page.
2. Click the **Administration** tab to display the Administration page.
3. On the Administration page, click **Ports** to display the Ports page.
4. In the **Listen Ports** section, locate the appropriate port that has HTTP or HTTPS in the **Protocol** column.
5. Enter the new port number in the **Port** field.
6. Click **OK** to apply changes.
7. When prompted, click **Restart Web Cache** to restart the cache.

Task 3: Change the OracleAS Web Cache Logical Site Port

If the OracleAS Web Cache listen port is the same as the logical site port, update the logical site port as follows:

1. On the Administration page, click **Sites** to display the Sites page.
2. Locate the sites that use the old port number. If there is no site using the old port number, then the OracleAS Web Cache listener and site do not share the same port number.
3. For each site using the old port number:
   a. Select the site and click **Edit.**
   b. In the Edit Named Site page or the Server Mapping for Unnamed Site page, enter the new port number in the **Port** field.
4. Click **OK** to apply changes.
5. When prompted, click **Restart Web Cache** to restart the cache.

Task 4: Update the Oracle HTTP Server Port Directive

If you are changing the OracleAS Web Cache HTTP listen port to be the same as the logical site port, update the Port directive in the Oracle HTTP Server `httpd.conf` file:

1. Open the `httpd.conf` file:

   ```
   (UNIX) ORACLE_HOME/Apache/Apache/conf/httpd.conf
   ```

```
(Windows) ORACLE_HOME\Apache\Apache\conf\httpd.conf
```

2. Update the Port directive with the new port number, and then save the file.

   Do not modify the Listen directive. The OracleAS Web Cache port must be the same as the Oracle HTTP Server Port directive.

3. Run the following command:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs
(Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct ohs
```

If you are changing the OracleAS Web Cache HTTPS listen port, update the Port directive in the Oracle HTTP Server `ssl.conf` file:

1. Open the `ssl.conf` file:

```
(UNIX) ORACLE_HOME/Apache/Apache/conf/ssl.conf
(Windows) ORACLE_HOME\Apache\Apache\conf\ssl.conf
```

2. Update the SSL Port directive with the new port number, and then save the file.

   Do not modify the Listen directive. The OracleAS Web Cache SSL port must be the same as the Oracle HTTP Server SSL Port directive.

3. Run the following command:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs
(Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct ohs
```

Task 5: Update the Application Server Control Console

Update the Application Server Control Console with the new port number:

1. Open the `targets.xml` file:

```
(UNIX) ORACLE_HOME/sysman/emd/targets.xml
(Windows) ORACLE_HOME\sysman\emd\targets.xml
```

2. Update each occurrence of the old OracleAS Web Cache listen port number with the new port number, and then save the file.

   Depending on your configuration, this file may not contain any occurrences of the OracleAS Web Cache listen port, or it may contain many occurrences. The listen port may occur as a parameter on its own, or it may be part of a URL. The easiest way to edit this file is to search for all occurrences of the old OracleAS Web Cache listen port number, and replace them with the new port number.

3. Reload the Application Server Control Console:

```
(UNIX) ORACLE_HOME/bin/emctl reload
(Windows) ORACLE_HOME\bin\emctl reload
```

Task 6: Update mod_osso

If you have registered your virtual host as an OracleAS Single Sign-On partner application, follow these steps to re-register your virtual host with the new port number:

1. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.

2. If you are changing the OracleAS Web Cache HTTP listen port, take the following steps:

   a. Re-register `mod_osso` with the new port number by running the following command in the middle-tier Oracle home:

      UNIX:

      ```
      $ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path middle_tier_oracle_home
      -site_name middle_tier_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      ```

      Windows:

      ```
      ORACLE_HOME\sso\bin\ssoreg.bat
      -oracle_home_path middle_tier_oracle_home
      -site_name middle_tier_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      ```

      For example, if you want to change the OracleAS Web Cache HTTP listen port to 7779 on middle-tier host `myhost` on UNIX:

      ```
      $ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path /disk1/oracleas
      -site_name myhost:7779
      -config_mod_osso TRUE
      -mod_osso_url http://myhost.mydomain:7779
      ```

      See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for more information on registering `mod_osso`

   b. Restart the Oracle HTTP Server:

      ```
      (UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
      (Windows) ORACLE_HOME\opmn\bin\opmnctl restartproc process-type=HTTP_Server
      ```

3. If you are changing the OracleAS Web Cache HTTPS listen port, perform the following steps:

   a. Re-register `mod_osso` with the new port number by running the following command in the middle-tier Oracle home:

      On UNIX:

      ```
      $ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path middle_tier_oracle_home
      -site_name middle_tier_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      -virtual_host
      -config_file path/osso-https.conf
      ```

      On Windows:

      ```
      ORACLE_HOME\sso\bin\ssoreg.bat
      -oracle_home_path middle_tier_oracle_home
      -site_name middle_tier_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      -virtual_host
      ```

```
-config_file path\osso-https.conf
```

For example, if you want to change the OracleAS Web Cache HTTPS listen port to 4445 on middle-tier host `myhost` on UNIX:

```
$ORACLE_HOME/sso/bin/ssoreg.sh
-oracle_home_path /disk1/oracleas
-site_name myhost:4445
-config_mod_osso TRUE
-mod_osso_url http://myhost.mydomain:4445
-virtual_host
-config_file $ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
```

See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for more information on registering `mod_osso`

b. Edit the `mod_osso.conf` file, which is located at:

```
(UNIX) ORACLE_HOME/Apache/Apache/conf/mod_osso.conf
(Windows) ORACLE_HOME\Apache\Apache\conf\mod_osso.conf
```

In the `mod_osso.conf` file, comment the following directive, if you have not previously done so:

On UNIX:

```
LoadModule osso_module libexec/mod_osso.so
```

On Windows:

```
LoadModule osso_module modules\ApacheModuleOsso.dll
```

c. In the `httpd.conf` file, which is found in the same (`conf`) directory, add the directive that you just commented in the preceding step (if you have not previously done so). In a default setup, place the directive right after:

```
LoadModule wchandshake_module libexec/mod_wchandshake.so
```

d. In the `ssl.conf` file, which is also in the `conf` directory, update VirtualHost to include the `osso.conf` file for the virtual host. Name the file `osso-https.conf` to avoid conflict with the default `osso.conf` file. Check that the file name matches the name used in the registration script.

```
<VirtualHost _default_:4445>
.
.
.
OssoConfigFile ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
OssoIpCheck off
<Location /your_protected_url_for_the virtual site>
AuthType basic
Require valid-user
</Location>
.
.
.
</VirtualHost>
```

e. Update the Distributed Cluster Management Repository. For example:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -v -d
```

```
(Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -v -d
```

f. Restart the Oracle HTTP Server. For example:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
(Windows) ORACLE_HOME\opmn\bin\opmnctl restartproc process-type=HTTP_Server
```

Task 7: Update OracleAS Portal Configuration

If you are changing the OracleAS Web Cache HTTP listen port in a configuration with OracleAS Portal, update OracleAS Portal configuration with the new port number:

1. Using the Application Server Control Console, navigate to the Portal Home page.
2. In the **Administration** section, click **Portal Web Cache** Settings.
3. In the **Listening Port** field, enter the new port number.
4. Click **Apply.**

If you are changing the OracleAS Web Cache HTTPS listen port in a configuration with OracleAS Portal, update OracleAS Portal configuration with the new port number:

1. Update OracleAS Portal configuration:
   a. Using the Application Server Control Console, navigate to the Portal Home page.
   b. In the **Administration** section, click **Portal Web Cache Settings.**
   c. In the **Listening Port** field, enter the new port number.
   d. From the **Listening Port SSL Enabled** list, select **Yes.**
   e. Click **Apply.**
2. Update the `httpsports` parameter in the following file:

   ORACLE_HOME/j2ee/OC4J_Portal/applications/portal/portal/WEB-INF/web.xml

3. Restart the OC4J_Portal:
   a. Using the Application Server Control Console, navigate to the OC4J: OC4J_Portal Home page.
   b. In the General section, click **Start.**

See Also: *Oracle Application Server Portal Configuration Guide* for more information on updating the Portal Web Cache Settings

Task 8: Update Web Providers

If you are using Web Providers with OracleAS Portal, you must update them as follows (note that locally hosted Web Providers run on the same middle-tier instance as OracleAS Portal):

1. Log in to OracleAS Portal as the administrator (for example, `portal`).
2. Click the **Administer** tab.
3. Click the **Portlets** sub-tab.
4. Repeat this step for all locally hosted Web Providers registered in your Portal:
   a. In the Remote Providers portlet, enter the provider name in the **Name** field. Click **Edit.**
   b. Click the **Connection** tab.

c. In the **URL** field, update the port to the new port number. Click **Apply.**

d. Click **OK.**

Task 9: Update OracleAS Wireless

If you have OracleAS Wireless configured, update OracleAS Wireless with the new port number:

1. Re-register OracleAS Wireless with OracleAS Single Sign-On by running the following command on the middle-tier host:

   ```
   (UNIX) ORACLE_HOME/wireless/bin/reRegisterSSO.sh new_wireless_url oracle_home
   administrator_dn
   (Windows) ORACLE_HOME\wireless\bin\reRegisterSSO.bat
   new_wireless_url oracle_home administrator_dn
   ```

   In the example:

   - *new_wireless_url*: Wireless HTTP URL with the new OracleAS Web Cache listen port.
   - *oracle_home*: Middle-tier Oracle home whose OracleAS Web Cache port you are changing.
   - *administrator_dn*: Oracle Internet Directory administrator.

   For example, if you have changed the OracleAS Web Cache listen port to 7779 on the middle-tier installation in `/home/oracle` on UNIX host `myhost`:

   ```
   ORACLE_HOME/wireless/bin/reRegisterSSO.sh http://myhost:7779/ptg/rm
   /home/oracle cn=orcladmin
   ```

2. Update the Wireless HTTP and HTTPS configuration information:

   a. Navigate to the Wireless Home page on the Application Server Control Console.

   b. Select the **Site Administration** link.

   c. In the **General Configuration** section, select the **HTTP, HTTPS Configuration** link.

   d. In the **URL** section, update each URL that contains the OracleAS Web Cache listen port with the new port number.

   e. Click **OK.**

3. Update the instance URLs:

   a. Navigate to the Wireless Home page on the Application Server Control Console.

   b. In the **Instance Configuration** section, select the **Instance URLs** link.

   c. On the Instance URLs page:

      – If **Use the Wireless Site URLs** is selected, you do not need to make any changes to this page.

      – If **Use the Wireless Instance URLs** is selected, update each URL that contains the OracleAS Web Cache listen port with the new port number.

   d. Click **OK**.

Task 10: Update OracleBI Discoverer

If you have OracleBI Discoverer configured, and you are using the port for the URL of the Discoverer Portlet Provider, edit the URL of the Discoverer Portlet Provider to use the new port number.

> **See Also:** Section "How to Edit Discoverer Portlet Provider" in *Oracle Business Intelligence Discoverer Configuration Guide*.

Task 11: Update OracleAS Reports Services

You do not need to make any configuration changes to Reports Service to reflect the change. However, if you have built any Web pages that contain links to the middle-tier Reports Service, you need to update those Web pages with the new port number.

Task 12: Restart the Middle-Tier Instance

Restart the middle-tier instance:

- On UNIX systems:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

- On Windows systems:

```
ORACLE_HOME\bin\emctl stop iasconsole
ORACLE_HOME\opmn\bin\opmnctl stopall
ORACLE_HOME\opmn\bin\opmnctl startall
ORACLE_HOME\bin\emctl start iasconsole
```

4.3.5.2 Changing the OracleAS Web Cache Administration Port

The tasks to change the OracleAS Web Cache administration port are:

- Task 1: Change the OracleAS Web Cache Administration Port
- Task 2: Update OracleAS Portal

Task 1: Change the OracleAS Web Cache Administration Port

To change the OracleAS Web Cache administration port on any installation type:

1. Using the Application Server Control Console, navigate to the Web Cache Home page.
2. Click the **Administration** tab to display the Administration page.
3. On the Administration page, click **Ports** to display the Ports page.
4. In the **Operation Ports** section, locate the **Administration** row.
5. Enter the new port number in the **Port** field.
6. Click **OK** to apply changes.
7. When prompted, click **Restart Web Cache** to restart the cache.

Task 2: Update OracleAS Portal

If you have OracleAS Portal configured, update OracleAS Portal configuration with the new port number:

1. Using the Application Server Control Console, navigate to the Portal Home page.

2. In the **Administration** section, click **Portal Web Cache Settings.**
3. In the **Administration Port** field, enter the new port number.

> **See Also:** *Oracle Application Server Portal Configuration Guide* for more information on updating the Portal Web Cache Settings

4.3.5.3 Changing the OracleAS Web Cache Invalidation Port

The tasks to change the OracleAS Web Cache invalidation port are:

- Task 1: Change the OracleAS Web Cache Invalidation Port
- Task 2: Update OracleAS Portal
- Task 3: Update Web Providers

Task 1: Change the OracleAS Web Cache Invalidation Port

To change the OracleAS Web Cache invalidation port on any installation type:

1. Using the Application Server Control Console, navigate to the Web Cache Home page.
2. Click the **Administration** tab to display the Administration page.
3. On the Administration page, click **Ports** to display the Ports page.
4. In the **Operation Ports** section, locate the **Invalidation** row.
5. Enter the new port number in the **Port** field.
6. Click **OK** to apply changes.
7. When prompted, click **Restart Web Cache** to restart the cache.

Task 2: Update OracleAS Portal

If you have OracleAS Portal configured, update OracleAS Portal with the new port number:

1. Using the Application Server Control Console, navigate to the Portal Home page.
2. In the **Administration** section, click **Portal Web Cache Settings.**
3. In the **Invalidation Port** field, enter the new port number.

> **See Also:** *Oracle Application Server Portal Configuration Guide* for more information on updating the Portal Web Cache Settings

Task 3: Update Web Providers

If you are using Web Providers with OracleAS Portal, you must update them to use the new port as follows:

1. Open the `cache.xml` file:

```
(UNIX) ORACLE_HOME/portal/conf/cache.xml
(Windows) ORACLE_HOME\portal\conf\cache.xml
```

2. Update the `port` attribute to the new port, and then save the file.
3. Restart OC4J_Portal:
   a. Using the Application Server Control Console, navigate to the OC4J: OC4J_Portal Home page.
   b. In the **General** section, click **Start.**

4.3.5.4 Changing the OracleAS Web Cache Statistics Port

To change the OracleAS Web Cache statistics port on any installation type:

1. Using the Application Server Control Console, navigate to the Web Cache Home page.
2. Click the **Administration** tab to display the Administration page.
3. On the Administration page, click **Ports** to display the Ports page.
4. In the **Operation Ports** section, locate the **Statistics** row.
5. Enter the new port number in the **Port** field.
6. Click **OK** to apply changes.
7. When prompted, click **Restart Web Cache** to restart the cache.

If you change the statistics protocol to HTTPS, it is not possible to view performance statistics in Enterprise Manager until a certificate named b64InternetCertificate.txt is uploaded in Base64 format to *ORACLE_HOME*/sysman/config on UNIX and *ORACLE_HOME*\sysman\config on Windows.

4.3.6 Changing the DCM Discovery Port

To change the DCM Discovery port number in any installation type:

1. Open the dcmCache.xml file:

   ```
   (UNIX) ORACLE_HOME/dcm/config/dcmCache.xml
   (Windows) ORACLE_HOME\dcm\config\dcmCache.xml
   ```

2. Under the <communication> element, update the discovery-port parameter in the <coordinator> element with the new port number, and then save the file.

   For example:

   ```
   <coordinator discovery-port="7110" original="true" />
   ```

3. In every instance in the farm, stop the Application Server Control Console and stop the DCM daemon:

   - On UNIX systems:

     ```
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=dcm-daemon
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\bin\emctl stop iasconsole
     ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=dcm-daemon
     ```

   It is important that you make sure all Application Server Control Console instances and DCM daemons in the farm are stopped before you proceed to the next step.

4. In every instance in the farm, start the DCM daemon and the Application Server Control Console:

   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=dcm-daemon
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=dcm-daemon
ORACLE_HOME\bin\emctl start iasconsole
```

4.3.7 Changing the Java Object Cache Port

To change the Java Object Cache port number in any installation type:

1. Open the `javacache.xml` file:

   ```
   (UNIX) ORACLE_HOME/javacache/admin/javacache.xml
   (Windows) ORACLE_HOME\javacache\admin\javacache.xml
   ```

2. Under the `<communication>` element, update the `discovery-port` parameter in the `<coordinator>` element with the new port number, and then save the file.

   For example:

   ```
   <coordinator discovery-port="7010" />
   ```

3. Restart all OC4J instances which contain J2EE applications that use JavaCache:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl restart -co OC4J_INSTANCE
   (Windows) ORACLE_HOME\dcm\bin\dcmctl restart -co OC4J_INSTANCE
   ```

4.3.8 Changing the Log Loader Port

To change the Log Loader port in any installation type:

1. Stop the Log Loader:
   a. Using the Application Server Control Console, navigate to the Home page for the instance whose Log Loader port you want to change.
   b. Click **Logs** in the upper-right corner.
   c. On the View Logs page, click **Search Log Repository.**
   d. On the View Logs page, click **Log Loader.**
   e. On the Log Loader page, click **Stop.**
2. Change the Log Loader port number:
   a. On the Log Loader page, in the Administration section, click **Log Loader Properties.**
   b. On the Log Loader Properties page, enter the new port number in the **Log Loader Port** field.
   c. Click **Apply.**
3. Start the Log Loader:
   a. At the top of the Log Loader Properties page, click **Log Loader** to get back to the Log Loader page.
   b. On the Log Loader page, click **Start.**

4.3.9 Changing OPMN Ports (ONS Local, Request, and Remote)

This section describes how to change any of the following port numbers:

- ONS Local port
- ONS Request port
- ONS Remote port

To change these ports:

1. Stop the Application Server Control Console, OPMN and all OPMN-managed processes:
   - On UNIX systems:

     ```
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\bin\emctl stop iasconsole
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ```

2. Open the `opmn.xml` file:

   ```
   (UNIX) ORACLE_HOME/opmn/conf/opmn.xml
   (Windows) ORACLE_HOME\opmn\conf\opmn.xml
   ```

3. Under the `<notification-server>` element, modify the `local`, `remote`, or `request` parameter, as desired, in the `<port>` element, and then save the file.

   For example:

   ```
   <port local="6101" remote="6201" request="6004"/>
   ```

4. Start OPMN:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl start
   (Windows) ORACLE_HOME\opmn\bin\opmnctl start
   ```

5. Reload OPMN:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl reload
   (Windows) ORACLE_HOME\opmn\bin\opmnctl reload
   ```

6. If this is an Infrastructure with Oracle Internet Directory, start Oracle Internet Directory:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OID
   (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=OID
   ```

7. Start the rest of the processes:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl startall
     ORACLE_HOME\bin\emctl start iasconsole
     ```

8. Update DCM:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct opmn
   (Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct opmn
   ```

4.3.10 Changing the Port Tunneling Port

To change the Port Tunneling port number in any installation type:

1. Open the `opmn.xml` file:

```
(UNIX) ORACLE_HOME/opmn/conf/opmn.xml
(Windows) ORACLE_HOME\opmn\conf\opmn.xml
```

2. Under the `<ias-component id="IASPT">` element, update the range parameter in the `<port>` element with the new range. For example:

   ```
   <port id="ajp" range="7501-7503"/>
   ```

   Note that the port number range specified in `opmn.xml` overrides any port number specified in `iaspt.conf`. So you only need to update the port number in `opmn.xml`.

3. Reload OPMN, then stop and restart all OPMN processes and the Application Server Control Console:

   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl reload
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl reload
     ORACLE_HOME\bin\emctl stop iasconsole
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ORACLE_HOME\opmn\bin\opmnctl startall
     ORACLE_HOME\bin\emctl start iasconsole
     ```

4.3.11 Changing the OracleAS Portal Port

OracleAS Portal uses the OracleAS Web Cache HTTP listen port on the instance. See Section 4.3.5.1 for information about changing the OracleAS Web Cache HTTP listen port.

4.3.12 Changing the OracleAS Wireless Port

OracleAS Wireless uses the OracleAS Web Cache HTTP listen port on the instance. See Section 4.3.5.1 for information about changing the OracleAS Web Cache HTTP listen port.

4.3.13 Changing OracleBI Discoverer Ports

The OracleBI Discoverer Preferences port cannot be changed after installation. Other OracleBI Discoverer services use the OracleAS Web Cache HTTP listen port on the instance. See Section 4.3.5.1 for information about changing the OracleAS Web Cache HTTP listen port.

4.3.14 Changing the OracleAS Forms Services Port

OracleAS Forms Services uses the OracleAS Web Cache HTTP listen port on the instance. See Section 4.3.5.1 for information about changing the OracleAS Web Cache HTTP listen port.

4.3.15 Changing OracleAS Reports Services Ports

The following sections describe how to change OracleAS Reports Services ports:

- Changing the OracleAS Reports Services Bridge Port
- Changing the OracleAS Reports Services Network Port
- Changing the OracleAS Reports Services SQL*Net Port

4.3.15.1 Changing the OracleAS Reports Services Bridge Port

To change the OracleAS Reports Services bridge port, take the following steps:

1. Stop the OracleAS Reports Services bridge, using the following command:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=bridge_name
   (Windows) ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=bridge_name
   ```

   Alternatively, you can use the following command:

   ```
   (UNIX) rwbridge.sh name=bridge_name shutdown=immediate
   (Windows) rwbridge.bat name=bridge_name shutdown=immediate
   ```

2. Edit the following file:

   ```
   (UNIX) ORACLE_HOME/reports/conf/repbrg_bridge_name.conf
   (Windows) ORACLE_HOME\reports\conf\repbrg_bridge_name.conf
   ```

3. In the file, update the port attribute, specifying the new port number. The following example shows the new port number to be 14012:

   ```
   <bridge version="10.1.2" port="14012" timeout="1000">
   ```

4. Save and close the file.
5. Start the OracleAS Reports Services bridge, using the following command:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=bridge_name
   (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=bridge_name
   ```

   Alternatively, you can use the following command:

   ```
   (UNIX) rwbridge.sh name=bridge_name
   (Windows) rwbridge.bat name=bridge_name
   ```

   See Also: *Oracle Application Server Reports Services Publishing Reports to the Web* for more information about the OracleAS Reports Services bridge configuration file.

4.3.15.2 Changing the OracleAS Reports Services Network Port

To change the OracleAS Reports Services network port, which is used for service discovery, take the following steps:

1. Stop the Reports Server, using the following command:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=reports_server
   (Windows) ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=reports_server
   ```

2. Stop the OC4J_BI_Forms instance. From the Application Server Home page, select the check box next to **OC4J_BI_Forms** and click **Stop.**
3. Edit the following file:

   ```
   (UNIX) ORACLE_HOME/reports/conf/rwnetwork.conf
   (Windows) ORACLE_HOME\reports\conf\rwnetwork.conf
   ```

Because new network configuration files are generated based on the `rwnetwork.template` file, also edit that file if you want all newly generated network configuration files to use the new port. The `rwnetwork.template` file is located in the same directory as the `rwnetwork.conf` file.

4. In the `rwnetwork.conf` file, and optionally in the `rwnetwork.template` file, change the port attribute of the multicast element, specifying the new port number. The following example sets the new port to be 14022:

   ```
   <multicast channel="228.5.6.7" port="14022" timeout="1000" retry="3"/>
   ```

5. Save and close the file.

6. Start the Reports Server, using the following command:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=reports_server
   (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=reports_server
   ```

7. Start the OC4J_BI_Forms instance. From the Application Server Home page, select the check box next to **OC4J_BI_Forms** and click **Start.**

4.3.15.3 Changing the OracleAS Reports Services SQL*Net Port

To change the OracleAS Reports Services SQL*Net port number:

1. On the Reports Services host, edit the `tnsnames.ora` file. The default location is:

   ```
   (UNIX) ORACLE_HOME/network/admin/tnsnames.ora
   (Windows) ORACLE_HOME\network\admin\tnsnames.ora
   ```

   In the `REP_`*`HOSTNAME`* entry, update the `PORT` parameter with the new port number.

2. On all client hosts, edit the `tnsnames.ora` file. In the `REP_`*`HOSTNAME`* entry, update the `PORT` parameter with the new port number.

4.4 Changing Infrastructure Ports

This section contains the following topics:

- Changing the OracleAS Metadata Repository Net Listener Port
- Changing Oracle Internet Directory Ports
- Changing the HTTP Server Port on an Identity Management Installation
- Changing OracleAS Certificate Authority Ports

4.4.1 Changing the OracleAS Metadata Repository Net Listener Port

First, determine if it is necessary to change the OracleAS Metadata Repository listener port number. If you are concerned about the fact that you have another database on your host using the same port, it is possible that the OracleAS Metadata Repository and the other database can use the same port.

The following are guidelines for port usage by multiple databases on the same host:

- Multiple Oracle9*i* and Oracle Database10*g* databases can share the same Oracle Net listener port. If you install a OracleAS Metadata Repository on a host that contains Oracle9*i* and Oracle Database 10*g* databases, they can all use port 1521. There is no need to change the OracleAS Metadata Repository port number.

- If the other databases on your system are Oracle8*i* databases running the Net8 listener, then the OracleAS Metadata Repository must use a different port. They cannot share the same port.

> **Note:** If you want to run two listeners that use the same key value on one host, refer to Section 4.4.1.1, "Changing the KEY Value for an IPC Listener"

If you determine that you want to change the OracleAS Metadata Repository Listener Port, follow the steps in this section. An OracleAS Metadata Repository may be used in several different ways. Use the following table to determine the steps that are required for changing your type of OracleAS Metadata Repository:

If the Metadata Repository is used as follows:	Follow these tasks to change its Oracle Net listener port:
■ Identity Management Repository, Product Metadata Repository, and Management (DCM) Repository ■ Registered with Oracle Internet Directory	Task 1: Stop Middle-tier Instances Task 2: Change the OracleAS Metadata Repository Oracle Net Listener Port Task 3: Update Oracle Internet Directory Task 4: Update OracleAS Single Sign-On Task 5: Update OracleAS Certificate Authority Task 6: Update the Application Server Control Console Task 7: Update Middle-Tier Instances
■ Identity Management Repository only ■ Registered with Oracle Internet Directory	Task 1: Stop Middle-tier Instances Task 2: Change the OracleAS Metadata Repository Oracle Net Listener Port Task 3: Update Oracle Internet Directory Task 4: Update OracleAS Single Sign-On Task 5: Update OracleAS Certificate Authority Task 6: Update the Application Server Control Console
■ Product Metadata and Management (DCM) Repository ■ Registered with Oracle Internet Directory	Task 1: Stop Middle-tier Instances Task 2: Change the OracleAS Metadata Repository Oracle Net Listener Port Task 3: Update Oracle Internet Directory Task 7: Update Middle-Tier Instances
■ Management (DCM) Repository only ■ Not registered with Oracle Internet Directory	Task 2: Change the OracleAS Metadata Repository Oracle Net Listener Port Task 8: Update J2EE and Web Cache Instances

Task 1: Stop Middle-tier Instances

Stop all middle-tier instances that use the Metadata Repository by running the following command in each middle-tier Oracle home:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows) ORACLE_HOME\opmn\bin\opmnctl stopall
```

Task 2: Change the OracleAS Metadata Repository Oracle Net Listener Port

On the OracleAS Metadata Repository host:

1. Make sure your ORACLE_HOME and ORACLE_SID environment variables are set.
2. If OPMN is running, stop it:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
   (Windows) ORACLE_HOME\opmn\bin\opmnctl stopall
   ```

3. Stop the OracleAS Metadata Repository listener:

   ```
   lsnrctl stop
   ```

4. Open the `listener.ora` file, which is located at:

   ```
   (UNIX) ORACLE_HOME/network/admin/listener.ora
   (Windows) ORACLE_HOME\network\admin\listener.ora
   ```

   a. Under the `LISTENER` entry, update the value for `PORT`.

   b. Add the following `SID_DESC` entry to the `SID_LIST_LISTENER` entry:

      ```
      (SID_DESC =
        (GLOBAL_DBNAME = service_name)
        (ORACLE_HOME = oracle_home_path)
        (SID_NAME = sid)
      )
      ```

5. Edit the `tnsnames.ora` file. The default location is:

   ```
   (UNIX) ORACLE_HOME/network/admin/tnsnames.ora
   (Windows) ORACLE_HOME\network\admin\tnsnames.ora
   ```

   a. Update the `PORT` value in each entry that applies to OracleAS Metadata Repository.

   b. Add an entry like the following:

      ```
      newnetport =
        (DESCRIPTION =
          (ADDRESS = (PROTOCOL = tcp) (HOST = hostname) (PORT = port)))
      ```

      In the example, *`hostname`* is the fully-qualified hostname and *`port`* is the new port number.

6. Start the OracleAS Metadata Repository listener:

   ```
   lsnrctl start
   ```

7. Using SQL*Plus, log in to the OracleAS Metadata Repository as the SYSTEM user with SYSDBA privileges and run the following command:

   ```
   SQL> alter system set local_listener='newnetport' scope=spfile;
   ```

8. Using SQL*Plus, restart OracleAS Metadata Repository:

   ```
   SQL> SHUTDOWN
   SQL> STARTUP
   ```

9. Start the directory server:

   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OID
     ```

- On Windows systems:

```
ORACLE_HOME\opmn\bin/opmnctl start
ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=OID
```

Task 3: Update Oracle Internet Directory

On the Identity Management host, update Oracle Internet Directory with the new Oracle Net listener port number:

1. Start Oracle Directory Manager:
   - On UNIX, use the following command:

     ```
     ORACLE_HOME/bin/oidadmin
     ```

   - On Windows, navigate to Oracle Directory Manager (**Start** > **Programs** > **Oracle Application Server** *Oracle_Home* > **Integrated Management Tools** > **Oracle Directory Manager.**)
2. Log in to Oracle Directory Manager.
3. In the System Objects frame:
   a. Expand **Entry Management.**
   b. Expand **cn=Oracle Context.**
   c. Select the DBName for the OracleAS Metadata Repository. For example, if the DBName is the default, `orcl`, select **cn=ORCL**.
4. On the Properties tab, update the `PORT` parameter in the **orclnetdescstring** field with the new port number.
5. Click **Apply.**
6. Start OPMN in the Oracle Internet Directory Oracle home:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME\opmn\bin\opmnctl startall
```

Task 4: Update OracleAS Single Sign-On

On the OracleAS Single Sign-On host:

1. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
2. Update OracleAS Single Sign-On with the new repository port number by running the following command in the OracleAS Single Sign-On Oracle home:

```
$ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoca.jar reassoc -repos
$ORACLE_HOME
```

3. In the OracleAS Single Sign-On Oracle home, restart OC4J:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc ias-component=OC4J
(Windows) ORACLE_HOME\opmn\bin\opmnctl restartproc ias-component=OC4J
```

Task 5: Update OracleAS Certificate Authority

If the Identity Management installation has OracleAS Certificate Authority:

1. Run the following command:

```
(UNIX) ORACLE_HOME/oca/bin/ocactl updateconnection
```

```
(Windows) ORACLE_HOME\oca\bin\ocactl updateconnection
```

2. Restart OracleAS Certificate Authority:

```
(UNIX) ORACLE_HOME/oca/bin/ocactl stop
(UNIX) ORACLE_HOME/oca/bin/ocactl start

(Windows) ORACLE_HOME\oca\bin\ocactl stop
(Windows) ORACLE_HOME\oca\bin\ocactl start
```

If you are not sure if OracleAS Certificate Authority is configured, examine the Application Server Control Home page to see if it is listed in the Components section.

Task 6: Update the Application Server Control Console

Update the Application Server Control Console with the new port number:

1. In the Identity Management Oracle home, edit the following file:

```
(UNIX) ORACLE_HOME/sysman/emd/targets.xml
(Windows) ORACLE_HOME\sysman\emd\targets.xml
```

2. Update the old OracleAS Metadata Repository port number with the new port number, and then save the file.

   Locate the `oracle_ldap` target and update the `PORT` parameter in the `ConnectDescriptor` value with the new port number. The easiest way to find this is to search the file for the old port number.

3. Reload the Application Server Control Console:

```
(UNIX) ORACLE_HOME/bin/emctl reload
(Windows) ORACLE_HOME\bin\emctl reload
```

Task 7: Update Middle-Tier Instances

In each middle-tier Oracle home that uses OracleAS Metadata Repository:

1. Update the following file with the new Oracle Net listener port number:

```
(UNIX) ORACLE_HOME/network/admin/tnsnames.ora
(Windows) ORACLE_HOME\network\admin\tnsnames.ora
```

2. Check the following file:

```
(UNIX) ORACLE_HOME/Apache/modplsql/conf/dads.conf
(Windows) ORACLE_HOME\Apache\modplsql\conf\dads.conf
```

   Locate the line that begins with `PlsqlDatabaseConnectString`.

   - If the line ends with `ServiceNameFormat` or `SIDFormat`, update the line with the new OracleAS Metadata Repository port number, save the file, and restart Oracle HTTP Server.
   - If the line ends with `NetServiceNameFormat`, you do not need to do anything.

3. Start the middle-tier instance:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME\opmn\bin\opmnctl startall
```

Task 8: Update J2EE and Web Cache Instances

If the Metadata Repository is not registered with Oracle Internet Directory and is used as an OracleAS Database-Based Farm, you must update each J2EE and Web Cache instance that uses the Metadata Repository as follows:

1. Using the Application Server Control Console, navigate to the Home page for the J2EE and Web Cache instance.
2. Click the **Infrastructure** link.
3. On the Infrastructure page, in the OracleAS Farm Repository Management section, click **Change.**
4. Select **Existing Database.**
5. Follow the steps in the wizard for supplying the new Metadata Repository port number.
6. When the wizard is finished, navigate to the instance Home page and start your instance by clicking **Start All.**

4.4.1.1 Changing the KEY Value for an IPC Listener

It is not possible to run two listeners at the same time that are configured to use the same KEY value in their IPC protocol address. By default, the OracleAS Metadata Repository listener has its IPC KEY value set to EXTPROC. Hence, if your computer has another IPC listener that uses the EXTPROC key, you should configure the OracleAS Metadata Repository listener to use some other key value such as EXTPROC1.

To change the KEY value of an IPC listener:

1. Stop the listener (make sure your ORACLE_HOME environment variable is set first):

   ```
   lsnrctl stop
   ```

2. Edit the `listener.ora` and `tnsnames.ora` files. In each file, find the following line:

   ```
   (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC))
   ```

   Change it to the following:

   ```
   (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1))
   ```

3. Restart the listener:

   ```
   lsnrctl start
   ```

4.4.2 Changing Oracle Internet Directory Ports

This section describes how to change the Oracle Internet Directory HTTP or HTTPS port on an Identity Management installation. When you change this port number, you must update any middle-tier instances that use the Identity Management installation.

The following tasks describe how to update the Oracle Internet Directory port number on Identity Management, including updating other components in the Infrastructure and updating the middle-tier instances that use the port:

- Task 1: Prepare the Middle-Tier Instances
- Task 2: Prepare the Infrastructure Instances

Task 1: Prepare the Middle-Tier Instances

Perform this task only if the Identity Management installation is being used by middle-tier instances. On each middle-tier instance that uses Identity Management, stop the middle-tier instance as follows:

1. On the Application Server Home page of the Application Server Control Console, click **Stop All.**
2. Leave the Application Server Control Console running.

It is important that you leave the Application Server Control Console running in each of the middle-tier instances while you perform this procedure.

Task 2: Prepare the Infrastructure Instances

Prepare the Infrastructure instances by taking these steps:

1. Make sure that Identity Management and its associated OracleAS Metadata Repository are started on the Infrastructure whose port number you are changing.
2. If any middle-tier instances use a different OracleAS Metadata Repository for their product metadata and DCM repositories, make sure those repositories are started. In short, make sure all Metadata Repositories in your environment are started.

Task 3: Change the Oracle Internet Directory Port

Change the Oracle Internet Directory port by taking these steps:

1. On the Oracle Internet Directory host:
   a. Create a file named `mod.ldif` with the following contents. You can create the file in any directory.

      For HTTP:

      ```
      dn:cn=configset0, cn=osdldapd, cn=subconfigsubentry
      changetype:modify
      replace:orclnonsslport
      orclnonsslport:new_nonssl_port_number
      ```

      For HTTPS:

      ```
      dn:cn=configset0, cn=osdldapd, cn=subconfigsubentry
      changetype:modify
      replace:orclsslport
      orclsslport:new_ssl_port_number
      ```

   b. Run the following command:

      For HTTP:

      ```
      ldapmodify -D cn=orcladmin -w password -p oid_port -f mod.ldif
      ```

      For HTTPS:

      ```
      ldapmodify -D cn=orcladmin -w password -p oid_port -U SSLAuth -f mod.ldif
      ```

Note that *oid_port* is the old Oracle Internet Directory port number. If you are changing the HTTPS port, provide the additional `-U` argument to specify the SSL authentication mode. Use one of the following values for *SSLAuth*: `1` for no authentication required; `2` for one-way authentication required; `3` for two-way authentication required.

2. On the Oracle Internet Directory host, stop the entire instance that contains Oracle Internet Directory, as well as the Application Server Control Console:
   - On UNIX systems:

     ```
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\bin\emctl stop iasconsole
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ```

3. Perform this step in the Oracle Internet Directory Oracle home. If you have OracleAS Metadata Repository installed in other Oracle homes that are registered with this Oracle Internet Directory, perform this step in each of those Oracle homes as well.
   a. Open the `ldap.ora` file:

      ```
      (UNIX) ORACLE_HOME/ldap/admin/ldap.ora
      (Windows) ORACLE_HOME\ldap\admin\ldap.ora
      ```

   b. Modify the following line to contain the new port number, and then save the file:

      ```
      DIRECTORY_SERVERS=(myhost.myco.com:non_ssl_port:ssl_port)
      ```

   c. Open the `ias.properties` file:

      ```
      (UNIX) ORACLE_HOME/config/ias.properties
      (Windows) ORACLE_HOME\config\ias.properties
      ```

   d. Change the value of `OIDport` (for an HTTP port change) or `OIDsslport` (for an HTTPS port change) to the new port number, and then save the file.
4. On the Oracle Internet Directory host, start the instance that contains Oracle Internet Directory, and start the Application Server Control Console:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl startall
     ORACLE_HOME\bin\emctl start iasconsole
     ```

5. Perform this step in the OracleAS Single Sign-On Oracle home:
   a. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
   b. Run the following command in the OracleAS Single Sign-On Oracle home:

```
$ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoca.jar reassoc
-repos $ORACLE_HOME
```

Task 4: Reconfigure OracleAS Certificate Authority

Perform this task if you are using OracleAS Certificate Authority:

1. If OracleAS Certificate Authority is running in a different Oracle home, do the following steps in the OracleAS Certificate Authority Oracle home:
   a. Open the `ias.properties` file:

      (UNIX) *ORACLE_HOME*/config/ias.properties
      (Windows) *ORACLE_HOME*\config\ias.properties

   b. Change the value of `OIDport` (for an HTTP port change) or `OIDsslport` (for an HTTPS port change) to the new port number, and then save the file.
2. Update OracleAS Certificate Authority with the new Oracle Internet Directory port number by running the following command in the OracleAS Certificate Authority Oracle home:

   (UNIX) *ORACLE_HOME*/oca/bin/ocactl changesecurity -server_auth_port *portnum*
   (Windows) *ORACLE_HOME*\oca\bin\ocactl changesecurity -server_auth_port *portnum*

   In the example, *portnum* is the OracleAS Certificate Authority Server Authentication Virtual Host (SSL) port; the default is 6600.

   See Also: *Oracle Application Server Certificate Authority Administrator's Guide* for more information

Task 5: Restart the Identity Management Instance

Restart the Identity Management instance:

- On UNIX systems:

  ORACLE_HOME/bin/emctl stop iasconsole
  ORACLE_HOME/opmn/bin/opmnctl stopall
  ORACLE_HOME/opmn/bin/opmnctl startall
  ORACLE_HOME/bin/emctl start iasconsole

- On Windows systems:

  ORACLE_HOME\bin\emctl stop iasconsole
  ORACLE_HOME\opmn\bin\opmnctl stopall
  ORACLE_HOME\opmn\bin\opmnctl startall
  ORACLE_HOME\bin\emctl start iasconsole

Task 6: Update the Middle-Tier Instances to Use the New Port Number

On each middle-tier instance that uses the Identity Management installation, run the Change Identity Management Services wizard and start the instance:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the middle-tier instance.
2. Click the **Infrastructure** link.
3. On the Infrastructure page, in the Identity Management section, click **Change.**
4. Follow the steps in the wizard for supplying the new Identity Management information (the new port number).

5. When the wizard is finished, navigate to the Application Server Home page and start the middle-tier instance by clicking **Start All.**

4.4.3 Changing the HTTP Server Port on an Identity Management Installation

This section describes how to change the Oracle HTTP Server HTTP or HTTPS listen port on an Identity Management installation. When you change this port number, you also effectively change the OracleAS Single Sign-On port number. This means you must update any middle-tier instances that use the OracleAS Single Sign-On port.

The following tasks describe how to update the Oracle HTTP Server port number on Identity Management, including updating other components in the Infrastructure and updating the middle-tier instances that use the port:

- Task 1: Prepare the Middle-Tier Instances
- Task 2: Prepare the Infrastructure Instances
- Task 3: Modify the Oracle HTTP Server Listen and Port Directives
- Task 4: Enable Oracle HTTP Server to Run as Root for Ports Less Than 1024 (UNIX Only)
- Task 5: Update the Application Server Control Console
- Task 6: Update OracleAS Single Sign-On
- Task 7: Re-register mod_osso
- Task 8: Update Oracle Delegated Administration Services
- Task 9: Update OracleAS Certificate Authority
- Task 10: Restart the Identity Management Instance
- Task 11: Restart OracleAS Certificate Authority
- Task 12: Update the Middle-Tier Instances to Use the New Port Number

Task 1: Prepare the Middle-Tier Instances

Perform this task only if the Identity Management installation is being used by middle-tier instances. On each middle-tier instance that uses Identity Management, stop the middle-tier instance as follows:

1. On the Application Server Home page of the Application Server Control Console, click **Stop All.**
2. Leave the Application Server Control Console running.

It is important that you leave the Application Server Control Console running in each of the middle-tier instances while you perform this procedure.

Task 2: Prepare the Infrastructure Instances

Prepare the Infrastructure by taking the following steps:

1. Make sure that Identity Management and its associated OracleAS Metadata Repository are started on the Infrastructure whose port number you are changing.
2. If any middle-tier instances use different Metadata Repositories for their product metadata and DCM repositories, make sure those are started. In short, make sure all Metadata Repositories in your environment are started.

Task 3: Modify the Oracle HTTP Server Listen and Port Directives

If you are changing the HTTP port, change both the Listen and Port directives to the new port number in the Oracle HTTP Server `httpd.conf` file. You can perform this task using the Application Server Control Console or manual steps.

- Using the Application Server Control Console:
  1. Navigate to the Application Server Home page and click **Ports.**
  2. On the Ports page, locate the Oracle HTTP Server Listen port and click the icon in the **Configure** column.
  3. On the Server Properties page:
     - Enter the new port number in the **Default Port** field. This is for the Port directive.
     - Enter the new port number in the **Listening Port** column. This is for the Listen directive. There may be more than one listening port listed. The only way to tell which is the non-SSL listen port is to choose the one with the old non-SSL listen port value.
  4. At the bottom of the page, click **Apply.**
  5. On the Confirmation page, click **No,** you would not like to restart now.
- Using manual steps:
  1. Open the `httpd.conf` file:

     ```
     (UNIX) ORACLE_HOME/Apache/Apache/conf/httpd.conf
     (Windows) ORACLE_HOME\Apache\Apache\conf\httpd.conf
     ```

  2. Update the non-SSL Listen and Port directives with the new port number, and then save the file.

     The value for Listen and Port must be the same port number, for example, to change the listener port to 7779:

     ```
     Listen 7779
     Port 7779
     ```

     There may be multiple Listen and Port directives in this file. Modify the Listen and Port directives that are not enclosed in an SSL virtual host container. The easiest way to locate the proper Listen and Port directives is to search the file for the old port number.

  3. Run the following command:

     ```
     (UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs
     (Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct ohs
     ```

If you are changing the HTTPS port, change both the SSL Listen and Port directives to the new port number in the Oracle HTTP Server `ssl.conf` file. You must do this using the following manual steps:

1. Edit the `ssl.conf` file, located at:

   ```
   (UNIX) ORACLE_HOME/Apache/Apache/conf/ssl.conf
   (Windows) ORACLE_HOME\Apache\Apache\conf\ssl.conf
   ```

2. Update the SSL Listen and SSL Port directives with the new port number, and then save the file.

The value for Listen and Port must be the same port number, for example, to change the listener port to 4445:

```
Listen 4445
Port 4445
```

Save and close the file.

3. Run the following command:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs
(Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct ohs
```

Task 4: Enable Oracle HTTP Server to Run as Root for Ports Less Than 1024 (UNIX Only)

Perform this task if you are changing the port to a value less than 1024 on UNIX.

By default, Oracle HTTP Server runs as a non-root user (the user that installed Oracle Application Server). On UNIX systems, if you change the Oracle Application Server non-SSL listen port number to a value less than 1024, you must enable Oracle HTTP Server to run as root, as follows:

1. Log in as root.
2. Run the following commands in the Infrastructure Oracle home:

```
cd ORACLE_HOME/Apache/Apache/bin
chown root .apachectl
chmod 6750 .apachectl
```

Task 5: Update the Application Server Control Console

Update the Application Server Control Console with the new port number:

1. Open the `targets.xml` file:

```
(UNIX) ORACLE_HOME/sysman/emd/targets.xml
(Windows) ORACLE_HOME\sysman\emd\targets.xml
```

2. Update each occurrence of the old Oracle HTTP Server listen port number with the new port number, and then save the file.

   Depending on your configuration, this file may not contain any occurrences of the Oracle HTTP Server listen port, or it may contain many occurrences. The listen port may occur as a parameter on its own, or it may be part of a URL. The easiest way to edit this file is to search for all occurrences of the old Oracle HTTP Server listen port number, and replace them with the new port number.

3. Reload the Application Server Control Console:

```
(UNIX) ORACLE_HOME/bin/emctl reload
(Windows) ORACLE_HOME\bin\emctl reload
```

Task 6: Update OracleAS Single Sign-On

Perform this task if OracleAS Single Sign-On is configured to use the Oracle HTTP Server HTTP listen port in the installation where you are changing the port.

1. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.

2. Run one or both of the following commands in the OracleAS Single Sign-On Oracle home:

   To change the non-SSL port:

   ```
   (UNIX) ORACLE_HOME/sso/bin/ssocfg.sh http hostname new_non_ssl_port_number
   (Windows) ORACLE_HOME\sso\bin\ssocfg.bat http hostname new_non_ssl_port_number
   ```

   To change the SSL port:

   ```
   (UNIX) ORACLE_HOME/sso/bin/ssocfg.sh https hostname new_ssl_port_number
   (Windows) ORACLE_HOME\sso\bin\ssocfg.bat https hostname new_ssl_port_number
   ```

   In the examples:

   - *hostname* is the host on which OracleAS Single Sign-On is running.
   - *new_non_ssl_port_number* is the new non-SSL Oracle HTTP Server listen port number.
   - *new_ssl_port_number* is the new SSL Oracle HTTP Server listen port number.

Task 7: Re-register mod_osso

Re-register `mod_osso` as follows:

1. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
2. On Windows systems, set the path, for example: `PATH=%PATH%;%ORACLE_HOME%\bin;%ORACLE_HOME%\lib`.
3. If you are changing the Oracle HTTP Server listen port, take the following steps:
   a. Re-register `mod_osso` to take care of the default partner applications by running the following command in the Identity Management Oracle home:

      UNIX:

      ```
      ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path identity_management_oracle_home
      -site_name identity_management_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      ```

      Windows:

      ```
      ORACLE_HOME\sso\bin\ssoreg.bat
      -oracle_home_path middle_tier_oracle_home
      -site_name identity_management_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      ```

      For example, if you want to change the Oracle HTTP Server listen port to 7779 on host `myhost` on UNIX:

      ```
      $ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path /disk1/oracleas
      -site_name myhost:7779
      -config_mod_osso TRUE
      -mod_osso_url http://myhost.mydomain:7779
      ```

4. If you are changing the Oracle HTTP Server SSL listen port, perform the following steps.

   a. Re-register `mod_osso` with the new port number by running the following command in the middle-tier Oracle home:

      UNIX:

      ```
      ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path identity_management_oracle_home
      -site_name identity_management_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      -config_file path/osso-https.conf
      ```

      Windows:

      ```
      ORACLE_HOME\sso\bin\ssoreg.bat
      -oracle_home_path identity_management_oracle_home
      -site_name identity_management_hostname:new_port_number
      -config_mod_osso TRUE
      -mod_osso_url mod_osso_url
      -config_file path\osso-https.conf
      ```

      For example, if you want to change the Oracle HTTP Server SSL listen port to 7778 on `myhost` on UNIX:

      ```
      $ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path /disk1/oracleas
      -site_name myhost:4445
      -config_mod_osso TRUE
      -mod_osso_url http://myhost.mydomain:7778
      -config_file $ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
      ```

      See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for more information on registering `mod_osso`

   b. Edit the `mod_osso.conf` file, which is located at:

      ```
      (UNIX) ORACLE_HOME/Apache/Apache/conf/mod_osso.conf
      (Windows) ORACLE_HOME\Apache\Apache\conf\mod_osso.conf
      ```

      In the `mod_osso.conf` file, comment the following directive, if you have not previously done so:

      On UNIX:

      ```
      LoadModule osso_module libexec/mod_osso.so
      ```

      On Windows:

      ```
      LoadModule osso_module modules\ApacheModuleOsso.dll
      ```

   c. In the `httpd.conf` file, which is found in the same (`conf`) directory, add the directive that you just commented in the preceding step (if you have not previously done so). In a default setup, place the directive right after:

      ```
      LoadModule wchandshake_module libexec/mod_wchandshake.so
      ```

5. Restart the Oracle HTTP Server:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
   ```

```
(Windows) ORACLE_HOME\opmn\bin\opmnctl restartproc process-type=HTTP_Server
```

6. If you have configured or modified any additional partner applications, you must also re-register those.

   See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for more information on registering `mod_osso`

Task 8: Update Oracle Delegated Administration Services

If you have Oracle Delegated Administration Services configured, and Oracle Delegated Administration Services uses the new port number, follow these steps to update the Oracle Delegated Administration Services URL entry in Oracle Internet Directory.

You can find out what port Oracle Delegated Administration Services uses with the following command:

```
ldapsearch -h oid_host -p oid_port -D "cn=orcladmin"
-w "password" -b "cn=OperationURLs,cn=DAS,cn=Products,cn=OracleContext"
-s base "objectclass=*" orcldasurlbase
```

To update Oracle Delegated Administration Services:

1. Create a file named mod.ldif with the following contents (you can create the file in any directory):

   ```
   dn:cn=OperationURLs,cn=DAS,cn=Products,cn=OracleContext
   changetype:modify
   replace:orcldasurlbase
   orcldasurlbase:http://hostname:new_http_port_number/
   ```

   Note the slash at the end of the `orcldasurlbase` URL.

2. Run the following command:

   ```
   ldapmodify -D cn=orcladmin -w password -p oid_port -f mod.ldif
   ```

Task 9: Update OracleAS Certificate Authority

If you are using OracleAS Certificate Authority:

1. Re-register OracleAS Certificate Authority with the OracleAS Single Sign-On server by running the following command in the OracleAS Certificate Authority Oracle home:

   ```
   (UNIX) ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port portnum
   (Windows) ORACLE_HOME\oca\bin\ocactl changesecurity -server_auth_port portnum
   ```

   In the example, *portnum* is the OracleAS Certificate Authority Server Authentication Virtual Host (SSL) port; the default is 6600.

   See Also: *Oracle Application Server Certificate Authority Administrator's Guide*

2. If OracleAS Certificate Authority is located in a different Oracle home than the OracleAS Single Sign-On server, restart Oracle HTTP Server and the `oca` instance in the OracleAS Certificate Authority Oracle home:

   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
     ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=oca
     ```

```
ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=oca
```

- On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl stopproc ias-component=HTTP_Server
ORACLE_HOME\opmn\bin\opmnctl stopproc process-type=oca
ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=HTTP_Server
ORACLE_HOME\opmn\bin\opmnctl startproc process-type=oca
```

Task 10: Restart the Identity Management Instance

Restart the Identity Management instance:

- On UNIX systems:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

- On Windows systems:

```
ORACLE_HOME\bin\emctl stop iasconsole
ORACLE_HOME\opmn\bin\opmnctl stopall
ORACLE_HOME\opmn\bin\opmnctl startall
ORACLE_HOME\bin\emctl start iasconsole
```

Task 11: Restart OracleAS Certificate Authority

If OracleAS Certificate Authority is configured in this instance, restart it:

```
(UNIX) ORACLE_HOME/oca/bin/ocactl start
(Windows) ORACLE_HOME\oca\bin\ocactl start
```

Task 12: Update the Middle-Tier Instances to Use the New Port Number

Now that you have changed the Oracle HTTP Server port on the Identity Management installation, you must update all middle-tier instances to use the new port number.

1. Update each middle-tier instance using the Change Identity Management wizard in the Application Server Control Console. Note that the wizard does not prompt you for the new port number; it retrieves the port number internally.

   On each middle-tier instance that uses Identity Management:

   a. Using the Application Server Control Console, navigate to the Application Server Home page for the middle-tier instance.

   b. Click the **Infrastructure** link.

   c. On the Infrastructure page, in the **Identity Management** section, click **Change.**

   d. Follow the steps in the wizard.

   e. When the wizard is finished, navigate to the Application Server Home page and start the middle-tier instance by clicking **Start All.**

2. Refresh the Oracle Internet Directory cache in your applications:

   a. Log in to the Portal.

   b. Click the Administrator tab.

   c. Click the global settings link.

   d. Click the **SSO/OID** tab.

e. Check the refresh Oracle Internet Directory cache settings and click **Apply**.

4.4.4 Changing OracleAS Certificate Authority Ports

This section describes how to change the following port numbers:

- OracleAS Certificate Authority Server Authentication Virtual Host (SSL)
- OracleAS Certificate Authority Mutual Authentication Virtual Host (SSL)

To change either of these port numbers:

1. Open the `ocm_apache.conf` file in the Oracle home of the Infrastructure that contains OracleAS Certificate Authority:

   ```
   (UNIX) ORACLE_HOME/Apache/Apache/conf/ocm_apache.conf
   (Windows) ORACLE_HOME\Apache\Apache\conf\ocm_apache.conf
   ```

   a. Modify the `Server` or `Mutual` port, or both, and then save the file.

      Note that each port number is listed in the file in two places:

      – As a Listen directive

      – As a default virtual host

      The easiest way to find these is to search for the old port number.

   b. Run the following command:

      ```
      (UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs
      (Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct ohs
      ```

2. Run the following command (make sure your ORACLE_HOME environment variable is set first):

   ```
   sqlplus oca/oca_admin_password @$ORACLE_HOME/oca/sql/ocaportchg
   ```

   a. Enter the Server Authentication Only port when prompted. If you do not want to change this port number, enter the old port number.

   b. Enter the Mutual Authentication port when prompted. If you do not want to change this port number, enter the old port number.

3. Re-register OracleAS Certificate Authority with the OracleAS Single Sign-On server by running the following command in the OracleAS Certificate Authority Oracle home:

   ```
   (UNIX) ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port portnum
   (Windows) ORACLE_HOME\oca\bin\ocactl changesecurity -server_auth_port portnum
   ```

   In the example, *portnum* is the OracleAS Certificate Authority Server Authentication Virtual Host (SSL) port; the default is 6600.

   See Also: *Oracle Application Server Certificate Authority Administrator's Guide*

4. Restart Oracle HTTP Server:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc type=ohs
   (Windows) ORACLE_HOME\opmn\bin\opmnctl restartproc type=ohs
   ```

5. Restart the OracleAS Certificate Authority OC4J instance:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc type=oc4j instancename=oca
   ```

```
(Windows) ORACLE_HOME\opmn\bin\opmnctl restartproc type=oc4j instancename=oca
```

6. Start Oracle Application Server Certificate Authority:

```
(UNIX) ORACLE_HOME/oca/bin/ocactl start
(Windows) ORACLE_HOME\oca\bin\ocactl start
```

4.5 Changing OracleAS Developer Kit Ports

OracleAS Developer Kits use the same ports as the J2EE and Web Cache installation type. To change port numbers in an OracleAS Developer Kit installation, refer to the instructions in Section 4.3.

4.6 Changing Oracle Content Management Software Development Kit Ports

To change the port numbers in an Oracle Content Management Software Development Kit, refer to the instructions in Section D.1.7.

5

Managing Log Files

Oracle Application Server components generate log files containing messages that record all types of events, including startup and shutdown information, errors, warning messages, access information on HTTP requests, and additional information. This chapter describes how to view and manage log files to assist in monitoring system activity and in diagnosing system problems.

It contains the following topics:

5.1 Introduction to Oracle Application Server Logging

The Application Server Control Console lets you list and search log files across Oracle Application Server components. You can view log files from the Application Server Control Console pages or download a log file to your local client and view the log files using another tool.

This section covers the following topics:

5.1.1 Understanding Log File Data and Naming

Several Oracle Application Server components use Oracle Diagnostic Logging (ODL). Using ODL, log file naming and the format of the contents of log files conforms to an Oracle standard and the diagnostic messages are written in XML. Some Oracle Application Server components do not use ODL, and write their diagnostic messages using a component specific text format.

Regardless of the format of the messages that are stored in log files, ODL or text based, you can view log files using the Application Server Control Console, or you can download log files to your local client and view them using another tool (for example a text editor, or another file viewing utility).

This section covers the following topics:

- ODL Message Formatting and ODL Log File Naming
- Log File Messages by Component

> **Note:** Some Oracle Application Server components do not support ODL. Other components support ODL, but do not enable ODL by default.

5.1.1.1 ODL Message Formatting and ODL Log File Naming

When Oracle Application Server components run and produce ODL messages, the messages are written to diagnostic log files using XML format. Each ODL message includes a HEADER element containing fields with information about the message, optionally a CORRELATION_DATA element containing information to assist in correlating messages across components, and a PAYLOAD element containing the message text, including optional arguments and associated values.

Using ODL, Oracle Application Server components write diagnostic log files to a logging directory and determine the names for logging directories using a component specific naming convention.

See Also:

- Section 5.6.2, "Understanding ODL Messages and ODL Log Files"
- Section 5.4.1, "Correlating Messages Across Log Files and Components"

5.1.1.2 Log File Messages by Component

Table 5–1 lists the supported message formats for each Oracle Application Server component. Several components optionally support ODL format, where ODL is not the default format.

Table 5–1 Diagnostic Message Format by Component

Component	Default Format	ODL Support	Location[1]
Oracle Application Development Framework	ODL	Yes	*ORACLE_HOME*/bc4j/logs/*OC4J_Name*
BPEL Process Manager	Text	No	*ORACLE_HOME*/integration/orabpel/system/logs
CM SDK	Text	No	*ORACLE_HOME*/ifs/cmsdk/log/*cmsdk_domain_name*/Domain Controller.log *ORACLE_HOME*/ifs/cmsdk/log/*cmsdk_domain_name*/*node_name*.log
DCM	ODL	Yes	*ORACLE_HOME*/dcm/logs
Discoverer	Text	No	*ORACLE_HOME*/discoverer/logs Discoverer Viewer is an OC4J application. The log file is named application.log and is found in the directory *ORACLE_HOME*/j2ee/OC4J_BI_FORMS.
Enterprise Manager	Text	No	*ORACLE_HOME*/sysman/log
Forms	Text	No	*ORACLE_HOME*/j2ee/OC4J_BI_FORMS/application-deployments/formsapp/*island*/application.log

Table 5–1 (Cont.) Diagnostic Message Format by Component

Component	Default Format	ODL Support	Location[1]
HTTP Server	Text	Yes	*ORACLE_HOME*/Apache/Apache/logs/error_log.*time*
InterConnect	Text	No	*ORACLE_HOME*/oai/10.1.2/adapters/*adapter_name*/logs
Integration B2B	ODL	Yes	*ORACLE_HOME*/ip/log
BPEL Process Analytics	Text	No	*ORACLE_HOME*/integration/bam/log
Log Loader	ODL	Yes	*ORACLE_HOME*/diagnostics/logs
OC4J *instance_name*	Text	Yes	*ORACLE_HOME*/j2ee/*instance_name*/log *ORACLE_HOME*/j2ee/*instance_name*/application-deployments/*application_name*/application.log
OCA	Text	No	From the command line, for administrator use only, messages are stored at: *ORACLE_HOME*/oca/logs/admin.log Logging for user and administrator usage, other than command line, is stored in the database and accessed through the Oracle Application Server Certificate Authority (OCA) Administrator web interface.
Oracle Internet Directory	Text	No	*ORACLE_HOME*/ldap/log
OPMN	Text	No	*ORACLE_HOME*/opmn/logs *ORACLE_HOME*/opmn/logs/*component_type~*...
Port Tunneling	Text	No	*ORACLE_HOME*/iaspt/logs
Reports Server	Text	No	*ORACLE_HOME*/reports/logs
Single Sign-On	Text	No	*ORACLE_HOME*/sso/log
TopLink	Text	No	The log file location is specified with the `log path` configuration option in the OracleAS TopLink installation directory, for example: `config/toplink.xml`
Universal Installer	Text	No	*ORACLE_HOME*/cfgtoollogs
Web Cache	Text	No	*ORACLE_HOME*/webcache/logs
Wireless	Text	Yes	*ORACLE_HOME*/wireless/logs

[1] Locations are shown in UNIX format. Invert the slashes for Windows format.

5.1.2 Using a Log Repository

The Application Server Control Console supports viewing diagnostic messages from a Log Repository. A **Log Repository** can be file-based or stored in a database, and contains messages collected from multiple diagnostic log files across components. The Log Repository does not contain messages from access or trace log files because access logs and trace logs are verbose and do not contain diagnostic information.

The Oracle Application Server Log Loader component initializes and updates the data in a Log Repository. After the Log Loader starts, it stores information from diagnostic log files to the Log Repository at regular intervals.

Using a Log Repository consolidates Oracle Application Server log file data. Then, you can use the Application Server Control Console to easily search and view log file data generated by multiple components. Using a Log Repository can speed up the diagnostic process and reduce the resources required to support Oracle Application Server.

> **Note:** By default, the Log Loader is not started. Use the Application Server Control Console or OPMN to start Log Loader. See Section 5.5.1, "Starting and Stopping Log Loader".

5.1.3 Configuring Component Logging Options

Administrators configure logging options to manage and limit the logging information that Oracle Application Server components generate and save.

> **Note:** The Application Server Control Console does not directly support configuring logging options. In many cases, to configure component logging options, you need to use the Application Server Control Console Advanced Server Properties page to edit the values in configuration files.

The logging configuration options include:

- Specifying log file names and pathnames: Most Oracle Application Server components let you specify the directory for storing diagnostic log files. Specifying the diagnostic logging directory allows administrators to manage system and network resources.
- Limiting log file size: As Oracle Application Server components run and generate diagnostic messages, the size of the log files increases. Oracle Application Server components use one of several strategies to deal with log file size. Some components allow log files to keep increasing in size; in this case, it is the administrator's responsibility to monitor and clean up the log files. Other components, including OC4J, let you specify configuration options that limit how much log file data is collected and saved.
- Using log file archiving: Certain Oracle Application Server components let you specify configuration options to control the size of diagnostic logging directories. This lets you determine a maximum size for the directories containing a component's log files. When the maximum size is reached, older logging information is deleted before newer logging information is saved.
- Setting component logging levels: Certain Oracle Application Server components, including Oracle HTTP Server, allow administrators to configure logging levels. By configuring logging levels, the number of messages saved to diagnostic log files can be reduced. For example, you can set the logging level so that the system only reports and saves critical messages.

> **See Also:** Oracle Application Server component documentation for information on setting logging configuration options

5.2 Listing and Viewing Log Files with Application Server Control

Use Application Server Control Console to list log files by selecting the **Logs** link on the Application Server Control Console. This displays the View Logs page.

> **See Also:** Section 5.6.1, "Using the printlogs Tool to View Log Messages" for information on a command-line tool for viewing log files

This section covers the following:

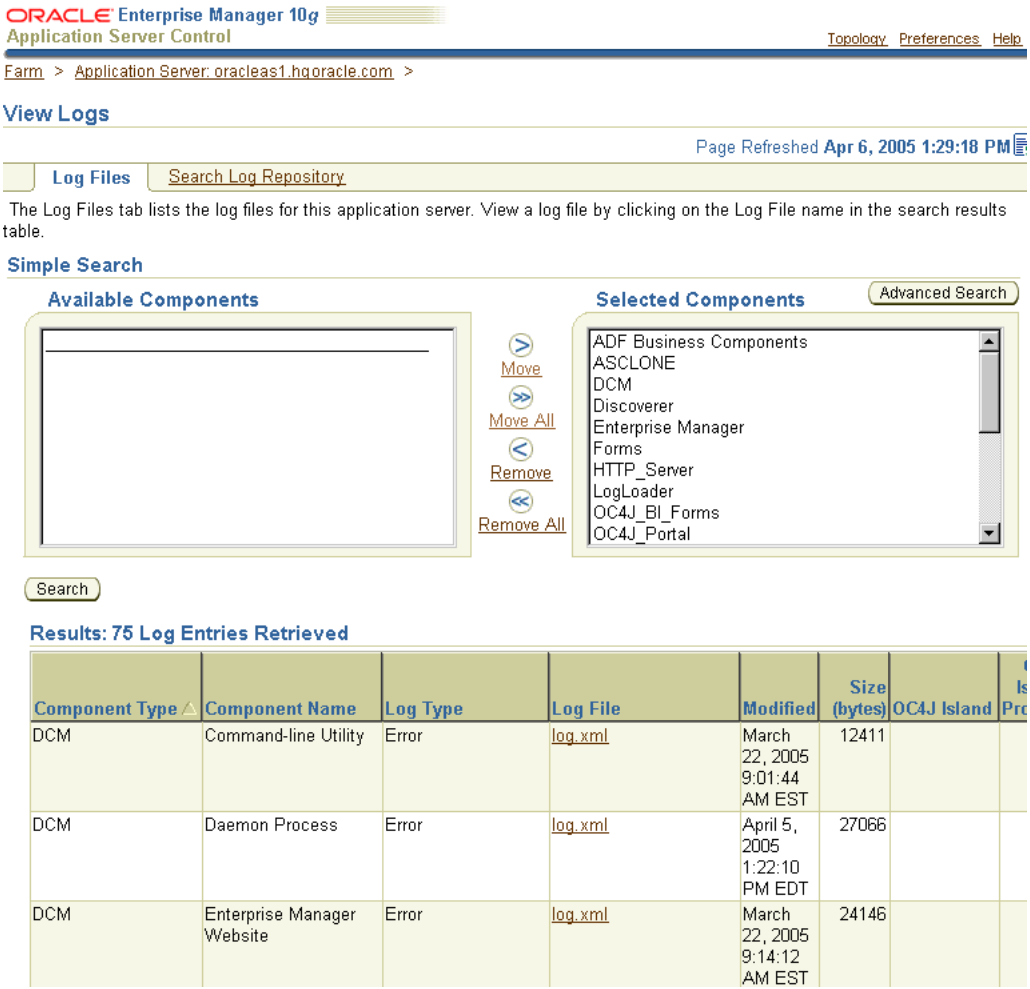
5.2.1 Listing Log Files for Components

You can list the log files for individual components from the Application Server Control Console. To list the log files, perform the following steps:

1. Select the **Logs** link on the Application Server Control Console. The View Logs page is displayed.
2. To view all components, click **Move All** to move all available components to **Selected Components**. To view some components, select them in the **Available Components** box and click **Move.**
3. Click **Search** to list the log files for the selected components.
4. After the search returns, the **Results** section shows log file information such as the name of the component associated with a log file and a link to the log file.

Figure 5–1 shows the Application Server Control Console View Logs page after a search.

Figure 5–1 Enterprise Manager View Logs Search Results



5.2.2 Listing Log Files from Oracle Application Server Components Pages

After you select a system component link on the Application Server Control Console Home page, you can view the log files for the selected component by clicking **Logs** at the top of the page. When you click **Logs**, the Application Server Control Console searches for the log files associated with the current component. Then, you can view the log files on the resulting View Logs page by selecting the links in the **Log Files** column in the **Results** table.

For example, if you click **Logs** on the HTTP Server Home page, Enterprise Manager searches for the log files associated with the Oracle HTTP Server and displays the View Logs page with a list of Oracle HTTP Server log files in the Results table.

When you select the **Logs** link from a component page, the log file pages include a **Return to** link at the bottom of each page. The **Return to** link returns you to the component page from which you selected the **Logs** link.

5.2.3 Using Log Files Advanced Search

You can filter the search for log files by certain log file attributes by using the Advanced Search page of the Application Server Control Console.

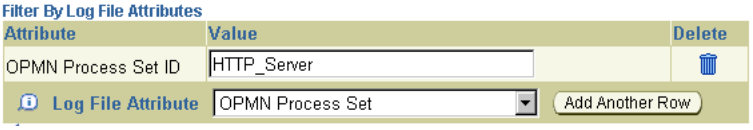
You can list log files using a search filter by performing the following steps:

1. Select the **Logs** link on an Application Server Control Console page. The View Logs page is shown.
2. Click **Advanced Search** to display the View Logs Advanced Search page. The Advanced Search page lets you list log files for Oracle Application Server components and enables you to filter the search for log files by certain log file attributes.
3. Select the desired components from the **Available Components** box and click **Move** or **Move All** to move components to the **Selected Components** box.
4. Select a field from the **Log File Attribute** list.
5. Click **Add Row** to add a row for the selected log file attribute.
6. Enter the desired search value in the **Value** field.
7. If you want to select additional fields with values, click **Add Another Row** and enter additional values.
8. Click **Search** to perform the search. When the search returns, the **Results** section shows log files with matching fields.

To obtain more information on filtering using log file attributes, click the information icon next to the **Log File Attribute** list.

Figure 5–2 shows the Advanced Search Filter By Log File Attributes selection box, with the **Log File Attribute** list and the **Add Another Row** button.

Figure 5–2 Log Files Advanced Search Filter By Log File Attributes



5.3 Searching Diagnostic Messages in a Log Repository

The Application Server Control Console lets you search through diagnostic messages in a Log Repository containing messages collected from several Oracle Application Server components. The advantage of using a Log Repository is that you can search, view, and correlate diagnostic messages in a uniform way across multiple Oracle Application Server components.

This section covers the following topics:

- Getting Started with Log Repository
- Searching Log Repository with Simple Search
- Searching Log Repository with Advanced Search
- Viewing Repository Log Entry Details
- Using Regular Expressions with Log Repository Search

5.3.1 Getting Started with Log Repository

The Log Repository must contain diagnostic messages before you can search the Log Repository. The Log Repository is initialized and updated by the Log Loader component, and the Log Loader must be started before you can search the Log Repository. The Log Loader is not started automatically; you must start the Log Loader to make sure that the Log Repository is updated.

> **See:** Section 5.5, "Using Oracle Application Server Log Loader" for information on starting and using Log Loader.

5.3.2 Searching Log Repository with Simple Search

To search for diagnostic log entries in the Log Repository using simple search criteria, do the following:

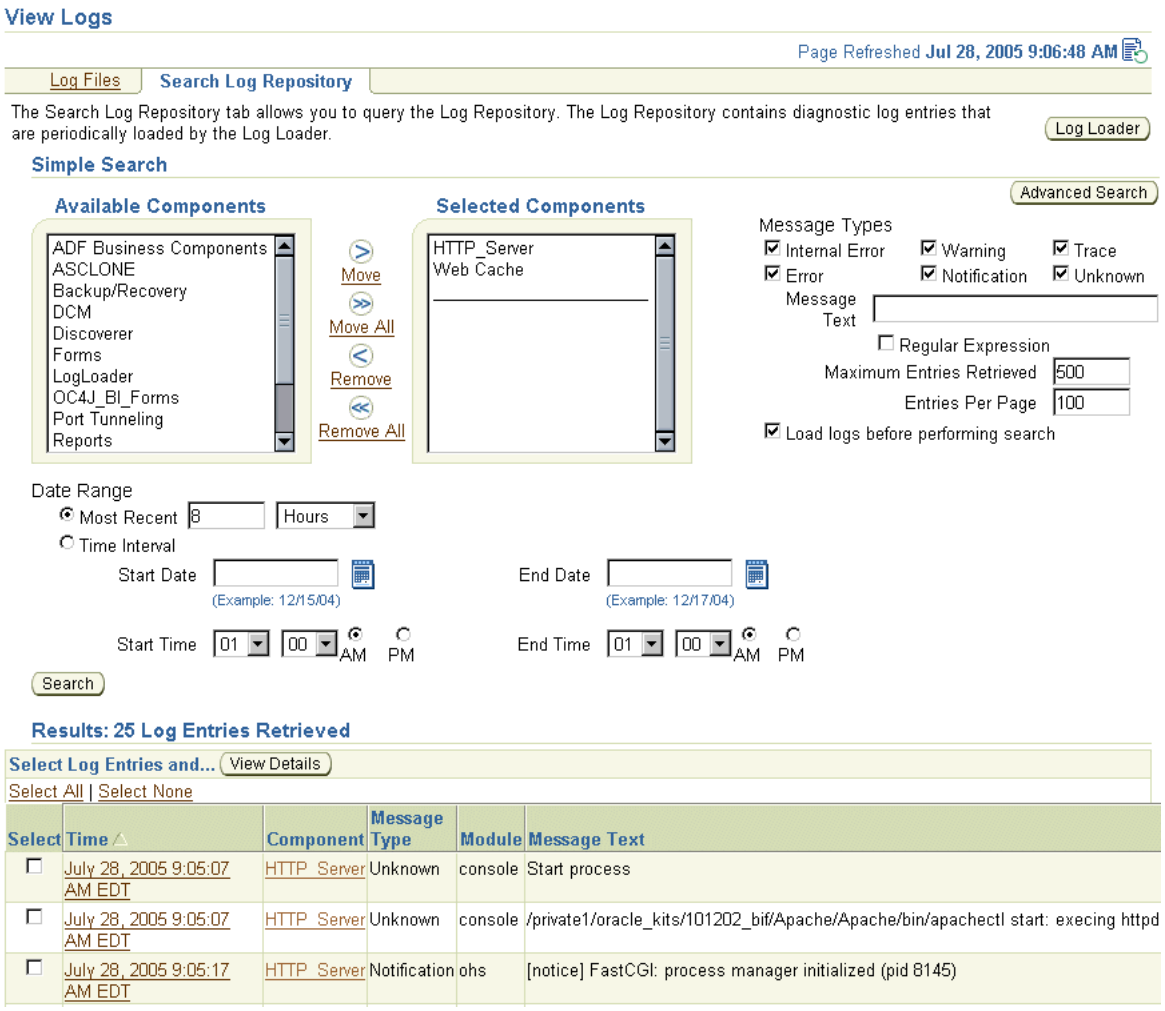
1. Select the **Logs** link on an Application Server Control Console page. The View Logs page is shown.
2. From the View Logs page, click **Search Log Repository.**
3. Select components from the **Available Components** box, then click **Move** or **Move All** to move the selected components to the **Selected Components** box. This step is optional.
4. Use the default selections, or select the available search and result display options. The online help describes the available search and display options for the Search Log Repository page.

> **Note:** The **Message Type** selection box includes the Unknown option. Some components do not include a message type when the component writes log file entries. These messages are loaded into the Log Repository with Unknown specified as the message type.

5. Click **Search** to search for messages in the Log Repository that match the constraints you specify. When the search returns, the Results section shows the matching diagnostic log messages from the Log Repository.

   Figure 5–3 shows the Search Log Repository page.

Figure 5–3 Search Log Repository Page



See Section 5.3.4, "Viewing Repository Log Entry Details" for information about viewing the log entry details.

5.3.3 Searching Log Repository with Advanced Search

With the Advanced Search feature, you can select from a set of repository query fields that can restrict the list of log message entries to those that apply to your criteria.

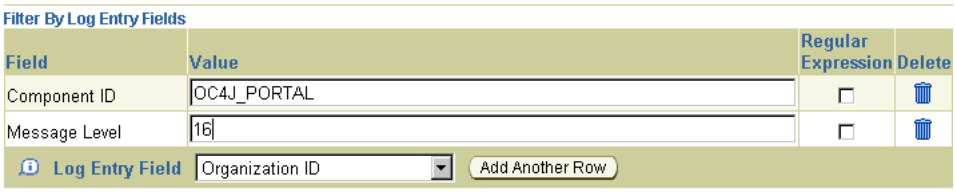
Take the following steps:

1. Select the **Logs** link on an Application Server Control Console page. The View Logs page is shown.
2. From the View Logs page, click **Search Log Repository,** then click **Advanced Search.**
3. On the Search Log Repository Advanced Search page, use the **Filter By Log Entry Fields** box to select log message fields and values to search. Select multiple fields by clicking **Add Another Row.** When you specify values for multiple fields, the search only returns results that match all of the specified constraints. The online help describes the available search and display options for the Search Log Repository page.

4. Use the default selections, or specify search and result date range and message type options by making selections and entering constraints.
5. Click **Search** to search for messages in the Log Repository that match the selection constraints. When the search returns, the Results section shows the matching log entries.

Figure 5–4 shows the Advanced Search Log Repository **Filter By Log Entry Fields** section.

Figure 5–4 Search Log Repository Advanced Search Filter By Log Entry Fields



5.3.4 Viewing Repository Log Entry Details

To view a log entry, either select the link shown in the **Time** field of the Results area on the View Logs page, or select entries in the **Select** field and then click **View Details**. The Log Entry Details page is displayed, as shown in Figure 5–5. It displays information about the log entry, including the Message Type, Component, the Message Text, and optionally the Execution Context ID (ECID).

Figure 5–5 Log Repository Log Entry Details Page

See Also: Section 5.4, "Diagnosing Problems and Correlating Messages" for information on Execution Context IDs

5.3.5 Using Regular Expressions with Log Repository Search

Regular expression matching is applied when the check box in the Regular Expression field is selected on the Log Repository Simple Search or Advanced Search page. On the Simple Search page, the Regular Expression check box is under the **Message Text** field. On the Advanced Search page, the Regular Expression check box is in the **Filter by**

Log Entry Fields box. Using a regular expression in a search enables you to enter a pattern description to match strings for a Log Repository search.

The Log Repository search uses the Apache Jakarta regular expression engine, which uses "*" for a string of characters, "?" for a single character, and supports boundary matches, including "^" for a match only at the beginning of an entry, and "$" for a match only at the end of an entry, and special characters, including "\t" for Tab, "\n" for newline, "\r" for return, and "\f" for form feed.

> **See Also:** http://jakarta.apache.org/regexp for more information on supported regular expressions

5.4 Diagnosing Problems and Correlating Messages

Generally, administrators and others view log file data to diagnose, monitor, and search for component errors or problems that may cause component errors. The Application Server Control Console supports a unified architecture and provides cross-component tools that can assist you in these tasks.

This section covers the following topics:

- Correlating Messages Across Log Files and Components
- Diagnosing Component Problems

5.4.1 Correlating Messages Across Log Files and Components

Certain Oracle Application Server components provide **message correlation** information for diagnostic messages. Message correlation information helps those viewing diagnostic messages determine relationships between messages across components. The Execution Context ID (ECID) is a globally unique identifier associated with a thread of execution. The ECID helps you to use log file entries to correlate messages from one application or across application server components. By searching related messages using the message correlation information, multiple messages can be examined and the component that first generates a problem can be identified (this technique is called **first-fault component isolation**). Message correlation data can help establish a clear path for a diagnostic message across components, within which errors and related behavior can be understood.

When you view an entry on the Log Entry Details page in the Application Server Control Console, if the Execution Context ID field is available, it displays the Execution Context ID as a link. Selecting the **Execution Context ID** link shows you all the diagnostic messages in the Log Repository with the same execution context ID.

You can use the ECID to track requests as they move through Oracle Application Server.

The ECID takes the following format:

```
request_id, sequence_number
```

- The *request_id* is a unique string that is associated with each request.
- The *sequence_number* represents the hop number of the request, as it passes through Oracle Application Server (or through the component).

  For example, OracleAS Web Cache assigns an initial sequence number of 0 to a request (when OracleAS Web Cache handles the request). After that, the sequence number is incremented as the request moves through Oracle Application Server components.

Table 5–2 lists the Oracle Application Server components that provide message correlation information (using an ECID), and if a component supports message correlation, but it is not enabled by default.

Table 5–2 Oracle Application Server Components Supporting Message Correlation

Component	Message Correlation Configuration Reference
DCM	Supports message correlation.
OC4J	Supports message correlation when ODL logging is enabled and when the property `oracle.dms.transtrace.ecidenabled` is set to the value `true` (by default this is `false`). This property is set on the OC4J command line. **See also:** Section 5.6.5 and *Oracle Application Server Containers for J2EE User's Guide* for details on enabling ODL logging in OC4J
HTTP Server	Supports message correlation. **See also:** Section 5.6.5
Portal	Supports message correlation. OracleAS Portal outputs the ECID with error messages in the Portal Repository Diagnostics log file. **See also:** "Diagnosing OracleAS Portal Problems" in the *Oracle Application Server Portal Configuration Guide*.
Web Cache	Supports message correlation. **See also:** The section, "Oracle-ECID Request-Header Field" in Chapter 2, "Caching Concepts" in the *Oracle Application Server Web Cache Administrator's Guide*

5.4.2 Diagnosing Component Problems

When an Oracle Application Server component has a problem, you can isolate and determine the cause of the problem by viewing the diagnostic messages. The following general techniques can assist you in accomplishing this task:

- Search for errors, or warnings, related to the problem
- Correlate the errors across components
- Correlate the errors across a time interval
- Perform component-based analysis

Using a Log Repository can make searching for the root cause of a problem much easier. A Log Repository consolidates log file data and enables you to easily search, correlate, and view log file data that is generated by multiple Oracle Application Server components. A Log Repository correlates cross-component information by time, and correlates events that occur in a cascading fashion. Once a problem is isolated to a particular component in the repository, then, if needed, the problem can be further analyzed by examining the component-specific diagnostic files.

5.5 Using Oracle Application Server Log Loader

The Oracle Application Server Log Loader component is a process that periodically updates a Log Repository. A Log Repository stores diagnostic messages read from multiple log files across Oracle Application Server components in a single Oracle home. After the Log Loader starts, at regular intervals it reads the contents of log files incrementally and writes the contents to the Log Repository.

This section covers the following topics:

- Starting and Stopping Log Loader

5.5.1 Starting and Stopping Log Loader

You can use the controls on the Application Server Control Console Log Loader page to start and stop the Log Loader.

Note: By default, when Oracle Application Server is installed, the Log Loader is stopped.

To start the Log Loader, perform the following steps:

1. Select the **Logs** link on any Application Server Control Console page.
2. From the View Logs page, select the **Search Log Repository** link.
3. Select **Log Loader.**
4. On the Log Loader page, click **Start.**
5. On the confirmation page, you can click one of the following:
   - **Cancel:** Cancels the operation.
   - **Start:** Starts the Log Loader, but it does not load any existing log messages from component log files. Only messages that are added to the component logs after the Log Loader is started are added to the Log Repository.
   - **Start and Load Existing Logs:** Starts the Log Loader and loads all existing log messages from component log files. Any messages that are added to the component logs after the Log Loader is started are also added to the Log Repository.

5.5.2 Enabling and Disabling Log Loader

On the Log Loader page, the **Enable** button enables the Log Loader. By default, when you first install Oracle Application Server, the Log Loader is enabled, but not started. When you disable the Log Loader, Enterprise Manager stops the Log Loader and the Log Loader component does not appear in the list of components on the View Logs page.

When you enable the Log Loader, the Log Loader component appears in the components list on the View Logs page, but it is not started.

5.5.3 Updating the Log Configuration

When the Log Loader starts, it loads configuration information about the component log files it will use as sources for the diagnostic messages that are stored in the Log Repository. (This includes information on the location and format of the log files).

Most log configuration files are installed when Oracle Application Server components are configured. The log configuration files for HTTP Server, OPMN, OC4J, and the Log Loader are generated when the Log Loader is initially started.

If configuration changes are made that effect the location of diagnostic log files for these components, click **Update Log Configuration** on the Log Loader page to regenerate the log configuration files for these components. This ensures the Log Loader is loading the correct set of logs into the Log Repository.

> **See Also:** Section 5.6.4, "Component Diagnostic Log File Registration"

5.5.4 Setting Log Loader Properties

You can set Log Loader properties from the Log Loader page. To navigate to the Log Loader page:

1. Select the **Logs** link on any Application Server Control Console page.
2. From the View Logs page, select the **Search Log Repository** link.
3. Click **Log Loader.**
4. Select the **Log Loader Properties** link in the Administration section. The Log Loader Properties page includes fields showing the current values for the Log Loader properties.

To change the Log Loader properties, perform the following steps:

1. Enter updated values in the appropriate fields on the Log Loader Properties page.
2. Click **Apply** to apply the new values.

Figure 5–6 shows the Application Server Control Console Log Loader Properties page.

Figure 5–6 Log Loader Properties Page

The Application Server Control Console online help includes detailed information on the Log Loader Properties fields.

5.5.5 Understanding Log Loader Diagnostic Messages

The Log Loader logs its diagnostic messages, including errors, to its log file. Diagnostic messages might include errors encountered due to an incorrect configuration, or errors that occur while the Log Loader is reading data from a log file or is writing data to the log repository.

The common Log Loader problems include:

- Errors in the Log Loader configuration file (*ORACLE_HOME*/diagnostics/config/logloader.xml). Errors in the configuration file usually prevent the Log Loader from running. Such errors need to be corrected before the Log Loader can work properly.
- Configuration errors that occur when a component's registration file contains errors (*ORACLE_HOME*/diagnostics/config/registration/*.xml). Errors in the registration files do not prevent the Log Loader from running but may prevent the contents of certain log files from being loaded in the repository. Typically, there are two common types of registration file errors:
  - XML syntax errors that prevent the file from being parsed. If such errors are encountered, the Log Loader completely ignores the contents of the file.
  - A wrong path specified for a configuration file. If the Log Loader cannot find a log file at the specified path, it issues a Warning level diagnostic message. This does not always indicate an error. For example, it is possible that the component that generates that log was not active when the Log Loader started and the log file had not been created yet. The Log Loader continues to look for the log file and starts reading messages when the log file is created.
- Errors may occur while the Log Loader is reading messages from a log file. If the log file includes contents that cannot be read or parsed, then the Log Loader issues a log message indicating that it cannot read part of the contents of the file. In this case, the Log Loader attempts to recover from the error and continue to read the Log File.
- Errors may occur when writing messages to the repository (for example, a disk error). This type of error may indicate a problem that may require attention from the system administrator to correct the problem.
- The Log Loader produces an error message when it skips reading log files because a log file exceeds the currently specified maximum load size. The maximum load size can be specified on the Log Loader properties page.

  In this case, the Log Loader logs an error message in the following format:

  ```
  Size of data to be read from log /logfile exceeds threshold of x bytes.
  Skipping y_skipped bytes and moving to end of log.
  ```

  This message indicates the size of data to be read exceeds the specified maximum load size *x*, and that the Log Loader is skipping to the end of the log file. The error message provides information on the name of the log file */logfile*, and the number of bytes skipped *y_skipped*.

5.6 Advanced Logging Topics

This section covers the following topics:

- Using the printlogs Tool to View Log Messages
- Understanding ODL Messages and ODL Log Files

- Understanding Log Loader Log File Format Conversion
- Component Diagnostic Log File Registration
- Configuring Components to Produce ODL Messages and ECIDs
- Creating and Managing a Diagnostic Message Database Repository
- Limitations and Configuration

5.6.1 Using the printlogs Tool to View Log Messages

The `printlogs` tool is a command-line alternative to the Application Server Control Console for viewing log messages. The `printlogs` tool supports a variety of options for gathering and filtering log messages, and prints the results to standard output in a single format. For example, you can use `printlogs` to:

- Read log messages from the Log Repository or individual log files
- Filter log messages according to timestamp or log field value
- Print log messages in ODL or text format
- Sort log messages by field
- Report the number of log messages of a specified type
- Run in a continuous loop, printing log reports and sleeping for a specified amount of time

> **See Also:** Appendix F for more information about the `printlogs` tool

5.6.2 Understanding ODL Messages and ODL Log Files

This section covers the following topics:

- ODL Message Contents
- ODL Log File Naming

5.6.2.1 ODL Message Contents

Using ODL, diagnostic messages are written to log files using XML format and each message includes a HEADER element containing information about the message, optionally a CORRELATION_DATA element containing information to assist in correlating messages across components, and a PAYLOAD element containing the message text including optional arguments and associated values.

Example 5–1 shows a sample ODL format message that includes the optional CORRELATION_DATA element.

Example 5–1 Sample ODL Message Content

```
<MESSAGE>
  <HEADER>
    <TSTZ_ORIGINATING>2002-04-01T18:38:48.058-08:00</TSTZ_ORIGINATING>
    <ORG_ID>oracle.com</ORG_ID>
    <COMPONENT_ID>OHS</COMPONENT_ID>
    <HOSTING_CLIENT_ID>0.0.255.255</HOSTING_CLIENT_ID>
    <MSG_TYPE TYPE="ERROR"></MSG_TYPE>
    <MSG_LEVEL>17</MSG_LEVEL>
    <HOST_ID>test-perf9</HOST_ID>
    <HOST_NWADDR>0.0.255.255</HOST_NWADDR>
```

```
    <MODULE_ID>apache_core</MODULE_ID>
    <PROCESS_ID>5713</PROCESS_ID>
  </HEADER>
  <CORRELATION_DATA>
    <EXEC_CONTEXT_ID>
      <UNIQUE_ID>1017715128:255..255.255.88:5713:0:1</UNIQUE_ID>
      <SEQ>1</SEQ>
    </EXEC_CONTEXT_ID>
  </CORRELATION_DATA>
  <PAYLOAD>
    <MSG_TEXT>File does not exist:
 /files/Apache/docs/images/java-apache-project.gif
    </MSG_TEXT>
 </PAYLOAD>
</MESSAGE>
```

Table 5–3 describes the contents of an ODL message header. For any given component that produces ODL format messages, the optional header fields may not be present in the generated diagnostic messages.

Table 5–3 ODL Format Message Header Fields

Header Field Name	Description	Required
COMPONENT_ID	The product or component ID for the component that originated the message.	Required
HOST_ID	The DNS host network ID.	Optional
HOST_NWADDR	The IP or other network address for the originating host.	Optional
HOSTING_CLIENT_ID	The ID of the client or security group that the message relates to.	Optional
MODULE_ID	The ID for the module that originated the message.	Optional
MSG_GROUP	The name of the group the message belongs to, for purposes of selecting similar messages.	Optional
MSG_ID	The message ID. The message ID uniquely identifies the message.	Optional
MSG_LEVEL	An integer value that qualifies the message type (MSG_TYPE). Lower level values are for higher severity errors. Possible values are 1 through 32.	Optional
MSG_TYPE	The type of the message. Possible values are: INTERNAL_ERROR, ERROR, WARNING, NOTIFICATION, TRACE, UNKNOWN. If MSG_TYPE is included, the TYPE attribute is required when MSG_TYPE is included in the message header.	Required
ORG_ID	The organization ID for the originating component. This is usually the domain name for the organization.	Optional
PROCESS_ID	The process ID for the process, or execution unit associated with the message. Java components may use this field to specify the process ID and the thread ID, or only the thread ID.	Optional
TSTZ_NORMALIZED	The timestamp normalized for clock drift across hosts. This field is used when the diagnostic message is copied to a repository in a different hosts.	Optional
TSTZ_ORIGINATING	The timestamp with local time zone. This specifies the date and time when the message was generated.	Required
USER_ID	The User ID associated with the message.	Optional

5.6.2.2 ODL Log File Naming

Using ODL provides the following benefits:

- ODL limits the total amount of diagnostic information saved.

- Older segment files are removed and newer segment files are saved in chronological fashion.
- Components can remain active, and do not need to be shutdown, when diagnostic logging files are cleaned.

Using ODL, Oracle Application Server components write diagnostic log files to a logging directory. Components determine the names for logging directories using a component-specific naming convention.

An **ODL log** is a set of log files that includes: the current ODL log file, typically named `log.xml`, and zero or more **ODL Archives (segment files)** that contain older messages. As the log file grows, new information is added to the end of the log file, `log.xml`. When the log file reaches the maximum segment size, it is renamed and a new log file, `log.xml` is created. (You can specify the maximum ODL segment size using component-specific configuration options.)

> **Note:** Some Oracle Application Server components, in particular Oracle HTTP Server, do not support the ODL log file naming mechanism that this section describes. In Oracle HTTP Server, ODL diagnostic messages are written to a file, `log.xml`, that does not have a configurable size limit.

Segment files are created when the ODL log file `log.xml` reaches the maximum segment size. That is, the `log.xml` is renamed to `log`*n*`.xml`, where *n* is an integer, and a new `log.xml` file is created when the component generates new diagnostic messages.

To limit the size of the ODL log, components use a configuration option specifying the maximum size of the logging directory. Whenever the sum of the sizes of all of the files in the directory reaches the maximum, the oldest archive is deleted to keep the total size under the specified limit.

> **Note:** The most recent segment file is never deleted.

For example, when the maximum directory size is reached, with the starting segment file named `log9872`, the following files could be present in the log file directory:

```
File                      Size

log.xml                   10002
log9872.xml               15000
log9873.xml               15000
log9874.xml               15000
log9875.xml               15000
log9876.xml               15000
```

In this case, when `log.xml` fills up, `log9872.xml` is removed and `log.xml` is moved to the new file `log9877.xml`. New diagnostic messages then are written to a new `log.xml`.

5.6.3 Understanding Log Loader Log File Format Conversion

The Log Loader reads logs in several different formats and it converts the contents of non-ODL logs to ODL format. In most cases, the resulting ODL log record will contain only a timestamp and the message text from the original log entry. Values for other

ODL message fields, such as COMPONENT_ID and MODULE_ID can be provided in the log registration file for each log, so that these values are set to all log records parsed from the log. The Log Loader attempts to determine the severity or level of each non-ODL log and generate an appropriate ODL message type. However, in many cases, if the severity or level cannot be determined, the resulting ODL log record will have the message type set to `UNKNOWN`.

The Log Loader can even read unformatted logs, which may not even contain timestamp values. This is the case for several logs in the *`ORACLE_HOME`*`/opmn/logs` directory which contain redirected output from Oracle Application Server processes managed by Oracle Process Manager and Notification Server (OPMN). When log entries do not contain a timestamp, the Log Loader sets the timestamp to the value of the "last known timestamp" for that log. The value of the last known timestamp is determined according to the following rules:

1. The initial value of the last known timestamp is zero. Note that whenever adding a log record to the repository, a zero value timestamp is converted to the current time.
2. If the Log Loader finds an OPMN generated timestamp, it sets the last known timestamp with its value.
3. When the Log Loader reaches the end of the log, it sets the last known timestamp with the current time. If the Log Loader is running regularly, such as once every five minutes, this results in timestamps that are approximate to the actual time the message was written, within a five minute range. If the Log Loader is not run frequently, the value of these timestamps could be inaccurate.

Note: The OC4J redirected logs found in the *`ORACLE_HOME`*`/opmn/logs` directory are not treated as unformatted logs, because each line in the OC4J logs contains a timestamp. Most other logs in this directory are treated as unformatted logs, and have timestamps assigned according to the preceding rules.

5.6.4 Component Diagnostic Log File Registration

The Application Server Control Console and the Log Loader read Oracle Application Server component diagnostic registration files to determine names, locations, and additional configuration information about diagnostic log files. The directory *`ORACLE_HOME`*`/diagnostics/config/registration` contains the diagnostic log file registration files.

Oracle Application Server components may have multiple registration files in the configuration registration directory.

The format for the registration files includes a Oracle Application Server component ID, and extension, `.xml`. Table 5–4 lists the Oracle Application Server Components and their associated Component IDs.

Note: Components are responsible for creating the component diagnostic registration files. Normally, Oracle Application Server administrators should not modify these files.

Table 5–4 Component IDs for Diagnostic Log File Configuration

Component Name	Component ID
ADF	ADFBC
B2B	INTEGRAT
BPEL Process Manager	INTEGBPM
DCM	DCM
Discoverer	DISCOVER
Enterprise Manager	EM
Forms Services	FORMS
HTTP Server	OHS
Infrastructure Database	RDBMS
InterConnect	INTEGIC
Internet Directory	OID
Listener for Infrastructure Database	LISTENER
Log Loader	LOGLOADER
OC4J	OC4J
OPMN	OPMN
Port Tunneling	IASPT
Portal	PORTAL
Report Services	REPORTS
Single Sign-On	SSO
TopLink	TOPLINK
Ultra Search	ULTRSRCH
Universal Installer	OUI
Web Cache	WEBCACHE
Wireless	WIRELESS

5.6.5 Configuring Components to Produce ODL Messages and ECIDs

Table 5–5 lists the Oracle Application Server components that support ODL messages but that generate text messages by default. By making configuration changes, you can configure these components to produce ODL messages and for OC4J, an ECID.

This section covers the following topics:

- Configuring Oracle HTTP Server to Produce ODL Messages
- Configuring OC4J to Produce ODL Messages
- Configuring OC4J to Produce ECIDs

See Table 5–1 for the complete list of Oracle Application Server components that produce ODL messages.

Table 5–5 Oracle Application Server Components with Options for Supporting ODL

Component	Default Format	ODL Support	Location[1]
HTTP Server	Text	Yes	*ORACLE_HOME*/Apache/Apache/logs
OC4J Instance	Text	Yes	*ORACLE_HOME*/j2ee/*instance_name*/log *ORACLE_HOME*/j2ee/application-deployments/*application_name*/application.log

[1] Locations are shown in UNIX format. Invert the slashes for Windows format.

5.6.5.1 Configuring Oracle HTTP Server to Produce ODL Messages

To configure the Oracle HTTP Server to produce ODL messages, perform the following steps:

1. Add a directory named `oracle` where the Oracle HTTP Server ODL messages will be stored. The directory should be located at the following location:

   ```
   (UNIX) ORACLE_HOME/Apache/Apache/logs
   (Windows) ORACLE_HOME\Apache\Apache\logs
   ```

2. Using the Application Server Control Console or the `dcmctl` command line utility, modify the `httpd.conf` file to set the value of the OraLogMode and OraLogSeverity directives. Using the Application Server Control Console, from the Administration section of the HTTP_Server page, select the **Advanced Server Properties** link. Specify the OraLogMode and OraLogSeverity directives in `httpd.conf`.

   For example:

   ```
   OraLogMode oracle
   OraLogSeverity NOTIFICATION
   ```

3. Using the Application Server Control Console, restart the HTTP Server.

> **See Also:** *Oracle HTTP Server Administrator's Guide* for details on using the OraLogMode and OraLogSeverity directives

5.6.5.2 Configuring OC4J to Produce ODL Messages

The supplied configuration files for OC4J include commented out specifications for ODL logging. Enabling ODL logging in OC4J involves uncommenting the ODL configuration options and restarting the associated OC4J instance.

To change the ODL logging configuration for OC4J, use the Application Server Control Console. Select the **Administration** link for the OC4J instance for which you want to enable ODL logging. Then, select the **Advanced Properties** link to show the Advanced Server Properties page. On this page, edit the configuration files and uncomment the lines that contain the `<odl>` element.

> **See Also:** Chapter 3, "Advanced Configuration Development, and Deployment" in *Oracle Application Server Containers for J2EE User's Guide*

5.6.5.3 Configuring OC4J to Produce ECIDs

OC4J supports generating an Execution Context ID (ECID) for its log file entries. You can use the ECID to track requests as they move through Oracle Application Server, or through OC4J. By default, ECID generation is disabled in OC4J.

To enable ECID generation in OC4J, set the Java command-line option `-Doracle.dms.transtrace.ecidenabled=true`.

To modify Java command-line options using the Application Server Control Console, do the following:

1. Select the **Administration** link on the OC4J Home page of the application server instance of interest.
2. Select **Server Properties** in the Instance Properties area.
3. Scroll down to the Multiple VM Configuration section. This section defines the ports and the command line options for OC4J and for the JVM that runs OC4J processes.
4. In the Command Line Options section, add the following at the end of the **Java Options** text field:

   `-Doracle.dms.transtrace.ecidenabled=true`
5. Click **Apply.**

Note the following when setting the `oracle.dms.transtrace.ecidenabled` property:

- The default value for `oracle.dms.transtrace.ecidenabled` is `false`.
- The property applies for the entire OC4J instance and it cannot be set to different values for different applications running on OC4J.
- When ODL is enabled for OC4J and the value for `oracle.dms.transtrace.ecidenabled` is `false`, OC4J uses an ECID that is generated from within OC4J, rather than receiving the ECID from Oracle HTTP Server. When ODL is enabled for OC4J, all log messages should include an ECID.

> **See Also:** "Advanced Configuration Development, and Deployment" in *Oracle Application Server Containers for J2EE User's Guide*

5.6.6 Creating and Managing a Diagnostic Message Database Repository

You can use SQL scripts to create and manage a database repository for diagnostic messages. By creating a database repository for diagnostic messages, you can search, view, and correlate diagnostic messages across multiple Oracle Application Server instances.

Use the SQL scripts described in the following sections to create and manage a repository for diagnostic messages. The scripts are located in the following directory:

- On UNIX:

  ORACLE_HOME`/diagnostics/admin`
- On Windows:

  ORACLE_HOME`\diagnostics\admin`

The database that hosts the Log Repository can be an Oracle9*i* database or an Oracle Database 10*g* database.

5.6.6.1 Creating a Diagnostic Message Database Repository

To create a diagnostic message database repository, take the following steps:

1. Make sure the ORACLE_HOME and ORACLE_SID environment variables are set.
2. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you have to set depend on the type of your UNIX operating system.
3. Choose an existing tablespace or create a new tablespace for the repository.

   To create a new tablespace, connect to an Oracle database as an administrator and run the script `dmrep_tablespace.sql`. This script requires two arguments: the name of the tablespace to be created and the location of the tablespace datafile, for example:

   ```
   SQL> connect SYS as SYSDBA;
   ...
   SQL> @ORACLE_HOME/diagnostics/admin/dmrep_tablespace.sql dmrep
   ORACLE_HOME/diagnostics/repository/dmrep.dbf
   ```

4. Choose an existing user or create a new user.

   To create a new user, connect to the Oracle database containing the tablespace for the repository as an administrator and run the script `dmrep_user.sql`. This script requires three arguments: name of the user, user password, and the default user tablespace. Use the tablespace you designated for the repository for the default user tablespace, for example:

   ```
   SQL> @ORACLE_HOME/diagnostics/admin/dmrep_user.sql dmrepusr dmreppw dmrep
   ```

5. Create the diagnostic message repository schema.

   To create the diagnostic message repository schema, run the script `dmrep_create.sql`. Connect to the new or existing tablespace as the designated user, for example:

   ```
   SQL> connect dmrepusr
   ...
   SQL> @ORACLE_HOME/diagnostics/admin/dmrep_create.sql
   ```

6. Change the Log Loader configuration to use the diagnostic message repository.

   In order for the Log Loader to load diagnostic messages into the repository, you must update the repository element in the `loglogader.xml` file. To edit the repository element, you must know the JDBC URL for the database hosting the diagnostic message repository. Replace the contents of the repository element with the following:

   ```
   <repository>
      <database_repository
         url="jdbc:oracle:thin:@DB host:DB port:DB instance"
         user="dmrepusr"/>
   </repository>
   ```

   Replace the variables in the preceding example with the values for your installation.

   Store the repository password for your installation in a wallet in the Log Loader configuration directory, using the following command:

   ```
   (UNIX) ORACLE_HOME/diagnostics/bin/logloader -storePassword -user dmrepusr -pwd
   dmreppw
   (Windows) ORACLE_HOME\diagnostics\bin\logloader -storePassword -user dmrepusr
   -pwd dmreppw
   ```

If your installation is part of an OracleAS Cluster, update the Log Loader configuration in one instance of the cluster and then run the following command to propagate the changes to the other instances in the cluster:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct logloader
(Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct logloader
```

5.6.6.2 Removing Old Messages from the Diagnostic Message Repository

Use the script `dmrep_drop.sql` to delete messages that are older than a specified number of days, hours, minutes, or seconds. The script takes two arguments:

- *N*, which is the number of units
- *Unit*, which must be one of the following: DAY, HOUR, MINUTE, or SECOND

The following is an example of the script with arguments:

```
SQL> @ORACLE_HOME/diagnostics/admin/dmrep_cleanup.sql 7 DAY
```

5.6.6.3 Deleting the Diagnostic Message Repository

Use the script `dmrep_drop.sql` to delete the schema for the diagnostic message repository. The following is an example of deleting the DMREP schema:

```
SQL> connect dmrepusr
...
SQL> @ORACLE_HOME/diagnostics/admin/dmrep_drop.sql
```

To delete the user and tablespace, connect to the database as administrator and run the SQL commands for dropping a user and dropping a tablespace. The following example drops a user and tablespace, including contents and datafiles:

```
SQL> connect sys as sysdba
...
SQL> drop user dmrepusr;
SQL> drop tablespace dmrep including contents and datafiles;
```

5.6.6.4 Reconfiguring Log Loader to Use a File-Based Repository

If you want to use a file-based repository instead of a database repository for the diagnostic message repository, you must update the repository element in the logloader.xml file. Replace the contents of the repository element with the following:

```
<repository>
   <xml_repository
      path="diagnostics/repository"
      encoding="utf-8" segmentSize="5242880" maxSize="52428800"/>
</repository>
```

If your installation is part of an OracleAS Cluster, update the Log Loader configuration in one instance of the cluster and then run the following command to propagate the changes to the other instances in the cluster:

```
(UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct logloader
(Windows) ORACLE_HOME\dcm\bin\dcmctl updateConfig -ct logloader
```

5.6.7 Limitations and Configuration

Note the following limitations and configuration issues with log files:

- The Logs link in the Application Server Control Console gives you an integrated view of many Oracle Application Server component log files. However, certain log files are only available at the component level. Oracle Application Server components use the directory *ORACLE_HOME*/diagnostics/config/registration to make their log files visible to the Application Server Control Console. Some Oracle Application Server component log files are not exposed through Application Server Control Console pages.
- If you shut down the Log Loader after the database is shutdown, and you restart Log Loader after the database is restarted, some messages may be reloaded to the repository database twice.

  Usually, Log Loader saves its state after each load. However, if the database is shutdown first, Log Loader does not save its state. When it restarts, it resets its state to the end of the last successful load and will repeat any load that was unsuccessful. If the repository database was shutdown in the middle of that load, some of the records may have been written to the repository database, but the entire load will be repeated.

6

Managing an OracleAS Metadata Repository

This chapter provides information on managing OracleAS Metadata Repository.

It contains the following topics:

6.1 Frequently Asked Questions About OracleAS Metadata Repository

OracleAS Metadata Repository is an Oracle Database 10*g* database and can be managed using standard database procedures and tools. However, there are some considerations for managing OracleAS Metadata Repository within the Oracle Application Server environment. This section answers frequently asked questions about managing the Metadata Repository.

- **What is a Metadata Repository?**

  A Metadata Repository is a database. It is pre-seeded with schemas to support Oracle Application Server components and services. See Appendix E for information on the schemas that are pre-seeded in the Metadata Repository.

- **When is a Metadata Repository required?**

  A Metadata Repository is required by the following installations:

  - An Identity Management installation requires one for Identity Management schemas.
  - A J2EE and Web Cache installation that is part of an OracleAS Cluster managed by a database repository requires one for the Management (DCM) schema.
  - A Portal and Wireless installation requires one for product metadata schemas.
  - A Business Intelligence and Forms installation requires one for product metadata schemas.

- **How can I obtain a Metadata Repository?**

  You can obtain a Metadata Repository in either of the following ways:

  - You can install a Metadata Repository as part of an Infrastructure installation with Oracle Universal Installer. This installs the Metadata Repository from scratch.

    See Also: *Oracle Application Server Installation Guide*

  - You can install a Metadata Repository into an existing database using the Oracle Application Server Metadata Repository Creation Assistant.

    See Also: *Oracle Application Server Metadata Repository Creation Assistant User's Guide*

- **Are there any tools for managing the Metadata Repository?**

  You can use Oracle Enterprise Manager. See Section 2.5, "Managing the OracleAS Metadata Repository Database with Database Control" for more information.

- **Can I use the Metadata Repository to deploy applications?**

  No. The Metadata Repository is not supported for deploying applications.

- **Are there any database features that are not supported by the Metadata Repository?**

  The following tablespace management features are not supported:

  - Using ALTER TABLESPACE to assign a different default tablespace to a user
  - Using ALTER TABLESPACE to reduce the number of tablespaces that were created when you initially created the Metadata Repository
  - Renaming a tablespace

  It is, however, possible to use ALTER TABLESPACE to do segment management using autoextend or any other feature.

- **Can a Metadata Repository coexist on a host with other databases?**

  Yes. As long as each database has a unique SID and global database identifier. The databases may be able to share a Net listener as follows:

  - Multiple Oracle9*i* and Oracle 10*g* databases can share the same Net listener port. If the other databases on your host are Oracle9*i* or Oracle 10*g* databases, the Metadata Repository can use the same Net listener port (for example, 1521) as the other databases.
  - If the other databases on your system are Oracle8*i* databases running Oracle Net8 listener, then the Metadata Repository must use a different port for its Net listener.

- **Can I change the Metadata Repository Net listener port after installation?**

  Yes. Refer to Section 4.4.1, "Changing the OracleAS Metadata Repository Net Listener Port" for more information.

- **Can I change the Metadata Repository SID and global database name after installation?**

  No. This is not supported.

- **Can I change the character set of the Metadata Repository?**

  Yes. Follow the instructions for changing the character set in the database documentation, then refer to Section 6.5, "Changing the Character Set of OracleAS Metadata Repository" for updates you need to make to Oracle Application Server.

- **Can I tune the Metadata Repository?**

  Yes, you can apply database tuning strategies to the Metadata Repository.

  One important point is that the processes and sessions parameters in the Oracle `init$SID.ora` configuration file should be tuned to allow the Metadata Repository to handle the maximum number of database sessions used by Oracle Application Server middle-tier installations, or other middle-tier installations accessing the Metadata Repository.

  The primary consumers of database sessions are OracleAS Portal and OracleAS Wireless. An `init$SID.ora` setting of processes=150 should support four middle-tier installations that include these components. Note that an OracleAS Portal best practice recommendation is to relocate the Portal instance out of the Infrastructure, which reduces the database connections requirement.

  > **See Also:** *Oracle Application Server Performance Guide* for a detailed description of the database connection usage of `mod_plsql` in an OracleAS Portal installation

- **Can I change Metadata Repository schema passwords?**

  Yes. However, you must make sure to use the correct procedure. Some schemas store their passwords in Oracle Internet Directory and you must change their passwords using the Application Server Control Console so the password is updated in Oracle Internet Directory and the database. See Section 6.4, "Changing OracleAS Metadata Repository Schema Passwords" for more information.

- **Can I delete schemas that I am not using from the Metadata Repository?**

  No. You should never delete any of the schemas are provided with the Metadata Repository.

- **Can I rename or relocate Metadata Repository datafiles after installation?**

  Yes. See Section 6.6, "Renaming and Relocating OracleAS Metadata Repository Datafiles" for more information.

- **Can I configure my Metadata Repository for high availability?**

  Yes. Oracle Application Server offers high availability options for the Metadata Repository, including:

  – Oracle Application Server Cold Failover Cluster

  – DCM-Managed Oracle Application Server Cluster

  – Oracle Application Server Disaster Recovery

  > **See Also:** *Oracle Application Server High Availability Guide*

- **Can I enable archive logging on the Metadata Repository?**

  Yes. This is part of the Oracle-recommended backup and recovery strategy. See Section 21.2.2, "Enabling ARCHIVELOG Mode" for more information.

- **How can I back up and recover the Metadata Repository?**

  Oracle provides a backup and recovery strategy for your entire Oracle Application Server environment, including the Metadata Repository. See Part V, "Backup and Recovery" for more information.

6.2 Postinstallation Status of OracleAS Metadata Repository Schemas

Table 6–1 shows the status of Metadata Repository schemas immediately after installation. The table contains the account status and initial password for each schema, depending on whether the Metadata Repository is registered with Oracle Internet Directory.

To unlock an account using SQL*Plus (be sure to set your ORACLE_HOME and ORACLE_SID environment variables before you run these commands):

```
ORACLE_HOME/bin/sqlplus "SYS/password_for_sys AS SYSDBA"
SQL> ALTER USER schema ACCOUNT UNLOCK;
```

To lock an account:

```
ORACLE_HOME/bin/sqlplus "SYS/password_for_sys AS SYSDBA"
SQL> ALTER USER schema ACCOUNT LOCK;
```

The method for changing passwords varies by schema. Refer to Section 6.4, "Changing OracleAS Metadata Repository Schema Passwords" to determine the proper way to change a password.

Table 6–1 Postinstallation Status of Schemas in a Metadata Repository

Schema	Account Status (Registered with Oracle Internet Directory)	Password (Registered with Oracle Internet Directory)	Account Status (Not Registered with Oracle Internet Directory)	Password (Not Registered with Oracle Internet Directory)
ANONYMOUS	OPEN	RANDOM	OPEN	RANDOM
B2B	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
BAM	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
CTXSYS	LOCKED	RANDOM	LOCKED	RANDOM
DBSNMP	OPEN	Set by user during installation	OPEN	Set by user during installation
DCM	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
DIP	LOCKED	EXPIRED	LOCKED	EXPIRED
DISCOVERER5	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
DMSYS	LOCKED	EXPIRED	LOCKED	EXPIRED
DSGATEWAY[1]	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
EXFSYS	LOCKED	EXPIRED	LOCKED	EXPIRED
INTERNET_APPSERVER_REGISTRY	LOCKED, NO CREATE SESSION	EXPIRED	LOCKED, NO CREATE SESSION	EXPIRED
IP[2]	LOCKED	EXPIRED	LOCKED	EXPIRED
MDDATA	LOCKED	EXPIRED	LOCKED	EXPIRED
MDSYS	LOCKED	EXPIRED	LOCKED	EXPIRED
MGMT_VIEW	OPEN	RANDOM	OPEN	RANDOM
OCA	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ODS	OPEN	Same as the `ias_admin` password supplied during installation	LOCKED	EXPIRED

Table 6–1 (Cont.) Postinstallation Status of Schemas in a Metadata Repository

Schema	Account Status (Registered with Oracle Internet Directory)	Password (Registered with Oracle Internet Directory)	Account Status (Not Registered with Oracle Internet Directory)	Password (Not Registered with Oracle Internet Directory)
OEM_REPOSITORY	OPEN	RANDOM - Stored in Oracle Internet Directory	OPEN	RANDOM
OLAPSYS	LOCKED	RANDOM	LOCKED	RANDOM
ORABPEL	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORAOCA_PUBLIC	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORASSO	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORASSO_DS	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORASSO_PA	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORASSO_PS	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORASSO_PUBLIC	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
ORDPLUGINS	LOCKED	EXPIRED	LOCKED	EXPIRED
ORDSYS	LOCKED	EXPIRED	LOCKED	EXPIRED
OUTLN	LOCKED	EXPIRED	LOCKED	EXPIRED
OWF_MGR	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
PORTAL	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
PORTAL_APP	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
PORTAL_DEMO	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
PORTAL_PUBLIC	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
SCOTT	LOCKED	EXPIRED	LOCKED	EXPIRED
SI_INFORMTN_SCHEMA	LOCKED	EXPIRED	LOCKED	EXPIRED
SYS	OPEN	Set by user during installation	OPEN	Set by user during installation
SYSMAN	OPEN	Set by user during installation	OPEN	Set by user during installation
SYSTEM	OPEN	Set by user during installation	OPEN	Set by user during installation
UDDISYS	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
WCRSYS	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
WIRELESS	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
WK_TEST	LOCKED	EXPIRED	LOCKED	EXPIRED
WKPROXY	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED

Table 6–1 (Cont.) Postinstallation Status of Schemas in a Metadata Repository

Schema	Account Status (Registered with Oracle Internet Directory)	Password (Registered with Oracle Internet Directory)	Account Status (Not Registered with Oracle Internet Directory)	Password (Not Registered with Oracle Internet Directory)
WKSYS	OPEN	RANDOM - Stored in Oracle Internet Directory	LOCKED	EXPIRED
WMSYS	LOCKED	EXPIRED	LOCKED	EXPIRED
XDB	LOCKED	RANDOM	LOCKED	RANDOM

[1] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the DSGATEWAY schema is not used. It is provided for backward compatibility.

[2] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the IP schema does not contain any data. It is has been replaced by the B2B schema and is provided only for backward compatibility.

6.3 Viewing OracleAS Metadata Repository Schema Passwords

If a Metadata Repository is registered with Oracle Internet Directory, some schema passwords are stored in the directory and you can view them using Oracle Internet Directory tools.

You can view the passwords for the following schemas in Oracle Internet Directory:

B2B
BAM
DCM
DISCOVERER5
DSGATEWAY
OCA
ODS
OEM_REPOSITORY
ORABPEL
ORAOCA_PUBLIC
ORASSO
ORASSO_DS
ORASSO_PA
ORASSO_PS
ORASSO_PUBLIC
OWF_MGR
PORTAL
PORTAL_APP
PORTAL_DEMO
PORTAL_PUBLIC
UDDISYS
WCRSYS
WIRELESS
WKPROXY
WKSYS

You can view the passwords using the following procedures:

- Viewing OracleAS Metadata Repository Schema Passwords using Oracle Directory Manager
- Viewing OracleAS Metadata Repository Schema Passwords using ldapsearch

6.3.1 Viewing OracleAS Metadata Repository Schema Passwords using Oracle Directory Manager

To view Metadata Repository schema passwords using Oracle Directory Manager, take the following steps:

1. Start Oracle Directory Manager:
   - On UNIX, use the following command:

```
ORACLE_HOME/bin/oidadmin
```

- On Windows, navigate to Oracle Directory Manager (**Start** > **Programs** > **Oracle Application Server** *Oracle_Home* > **Integrated Management Tools** > **Oracle Directory Manager.**)

2. Log in to Oracle Directory Manager as the `orcladmin` user.
3. In the System Objects frame, expand **Entry Management**, expand **cn=OracleContext**, expand **cn=Products**, expand **cn=IAS**, expand **cn=IAS Infrastructure Databases**, and expand **orclReferenceName=*dbname*** for the Metadata Repository.
4. Select the **OrclResourceName=*schema_name*** entry for the schema whose password you want to view.
5. In the Properties tab, you can view the password in the **orclpasswordattribute** field.

6.3.2 Viewing OracleAS Metadata Repository Schema Passwords using ldapsearch

You can view Metadata Repository schema passwords by using `ldapsearch`, a command-line tool. The command uses the following format:

```
ORACLE_HOME/bin/ldapsearch -p oid_port -h oid_hostname -D "cn=orcladmin"
-w orcladmin_password -b "orclresourcename=schema_name,
 orclreferencename=metadata_rep_global_db_name, cn=ias infrastructure databases,
 cn=ias, cn=products, cn=oraclecontext" -s base "objectclass=*"
 orclpasswordattribute
```

The command returns several lines of output. The password is listed in the following line:

```
orclpasswordattribute=password
```

The following example uses the `ldapsearch` tool to request the ORASSO schema password.

```
ORACLE_HOME/bin/ldapsearch -p 13060 -h myhost -D "cn=orcladmin"
-w mypassword -b "orclresourcename=ORASSO,
orclreferencename=orcl.mycompany.com, cn=ias infrastructure databases,
cn=ias, cn=products, cn=oraclecontext" -s base "objectclass=*"
orclpasswordattribute
```

The command returns the ORASSO schema password, which is `Og23NI78` in this example:

```
OrclResourceName=ORASSO,orclReferenceName=orcl.mycompany.com
cn=IAS Infrastructure Databases,cn=IAS,cn=Products,cn=OracleContext
orclpasswordattribute=Og23NI78
```

6.4 Changing OracleAS Metadata Repository Schema Passwords

The method for changing schema passwords in the Metadata Repository varies by schema. Some schemas store their passwords in Oracle Internet Directory; you must change their passwords using the Application Server Control Console so that both Oracle Internet Directory and the database are updated. Other schemas do not store their passwords in Oracle Internet Directory; you change their passwords in the database using SQL*Plus. A few schemas require special steps for changing their passwords.

Table 6–2 lists the appropriate method for changing each Metadata Repository schema.

Table 6–2 Methods for Changing Oracle Metadata Repository Schema Passwords

Schema	Method for Changing Password
B2B	You must change the password in two places: ■ Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2. ■ Manually change the password in Oracle Internet Directory. Refer to Section 6.4.3.
BAM	You must change the password in two places: ■ Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2. ■ Manually change the password in Oracle Internet Directory. Refer to Section 6.4.3.
DCM	If the Metadata Repository is registered with Oracle Internet Directory, you must change the password in two places: ■ Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2. ■ Manually change the password in Oracle Internet Directory. Refer to Section 6.4.3. If the Metadata Repository is not registered with Oracle Internet Directory, you need to change the password only in the database using SQL*Plus.
DISCOVERER5	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
DSGATEWAY[1]	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
IP[2]	You must change the password in two places: ■ Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2. ■ Manually change the password in Oracle Internet Directory. Refer to Section 6.4.3.
OCA	This schema requires special steps. Refer to *Oracle Application Server Certificate Authority Administrator's Guide* for advanced topics in administration.
ODS	This schema requires special steps. Refer to *Oracle Internet Directory Administrator's Guide* for information on resetting the default password for the database.
ORABPEL	You must change the password in two places: ■ Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2. ■ Manually change the password in Oracle Internet Directory. Refer to Section 6.4.3.
ORAOCA_PUBLIC	This schema requires special steps. Refer to *Oracle Application Server Certificate Authority Administrator's Guide* for advanced topics in administration.
ORASSO	Use the Application Server Control Console. Navigate to the Application Server Home Page for the Infrastructure (Identity Management) installation and follow the instructions in Section 6.4.1. Then, in the Application Server Control Console, navigate to the Application Server Home page. Select the check box next to **HTTP_Server** and click **Restart.** Select the check box next to **OC4J_SECURITY** and click **Start.**

Table 6–2 (Cont.) Methods for Changing Oracle Metadata Repository Schema Passwords

Schema	Method for Changing Password
ORASSO_DS	Use the Application Server Control Console. Navigate to the Application Server Home Page for the Infrastructure (Identity Management) installation and follow the instructions in Section 6.4.1.
ORASSO_PA	Use the Application Server Control Console. Navigate to the Application Server Home Page for the Infrastructure (Identity Management) installation and follow the instructions in Section 6.4.1.
ORASSO_PS	Use the Application Server Control Console. Navigate to the Application Server Home Page for the Infrastructure (Identity Management) installation and follow the instructions in Section 6.4.1. Changing the ORASSO_PS password requires that the database link from all Portal schemas to the ORASSO_PS schema be re-created. To do this, run the following command for each affected Portal instance: `ORACLE_HOME/portal/conf/ptlconfig -dad dad_name -site [-pw PORTAL_schema_password]` Refer to *Oracle Application Server Portal Configuration Guide*.
ORASSO_PUBLIC	Use the Application Server Control Console. Navigate to the Application Server Home Page for the Infrastructure (Identity Management) installation and follow the instructions in Section 6.4.1.
OWF_MGR	You must change the password in two places: ■ Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2. ■ Manually change the password in Oracle Internet Directory. Refer to Section 6.4.3.
PORTAL	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1. After you change the password, restart Oracle HTTP Server and OC4J_Portal: In the home page for the instance, select **HTTP_Server** and **OC4J_Portal** and click **Restart.**
PORTAL_APP	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
PORTAL_DEMO	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
PORTAL_PUBLIC	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
SCOTT	Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2.
SYS	Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2.
SYSTEM	Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2.
UDDISYS	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
WCRSYS	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
WIRELESS	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.

Table 6–2 (Cont.) Methods for Changing Oracle Metadata Repository Schema Passwords

Schema	Method for Changing Password
WK_TEST	Use SQL*Plus to change the password directly in the database. Refer to Section 6.4.2.
WKPROXY	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.
WKSYS	Use the Application Server Control Console. Navigate to the Application Server Home Page for the middle-tier instance that uses this schema and follow the instructions in Section 6.4.1.

[1] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the DSGATEWAY schema is not used. It is provided for backward compatibility.

[2] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the IP schema does not contain any data. It has been replaced by the B2B schema and is provided only for backward compatibility.

6.4.1 Changing Schema Passwords Using the Application Server Control Console

Some schemas store their passwords in Oracle Internet Directory. You must change their passwords using the Application Server Control Console so the password is updated in both the database and Oracle Internet Directory.

To change a schema password using the Application Server Control Console:

1. Depending on the schema, navigate to the home page for the middle-tier instance or the Infrastructure. Refer to Table 6–2 to determine which home page to use.
2. On the home page, click **Infrastructure.**
3. On the Infrastructure page, click **Change Schema Password.**
4. On the Change Schema Password page, select the schema. Enter the new password in the **Password** and **Confirm Password** fields. Click **OK.**

6.4.2 Changing Schema Passwords Using SQL*Plus

You can change some schema passwords directly in the database using SQL*Plus. To do so, connect to the database as a user with SYSDBA privileges and issue the following command:

```
SQL> ALTER USER schema identified by new_password;
```

For example, to change the SCOTT schema password to `abc123`:

```
SQL> ALTER USER SCOTT IDENTIFIED BY abc123;
```

6.4.3 Changing Schema Passwords in Oracle Internet Directory

A few schemas (DCM, B2B, OWF_MGR) require you to manually update the password in the Metadata Repository and in Oracle Internet Directory. Use the following procedure to change these passwords:

1. Start Oracle Directory Manager:
   - On UNIX, use the following command:

     ```
     ORACLE_HOME/bin/oidadmin
     ```

   - On Windows, navigate to Oracle Directory Manager (**Start** > **Programs** > **Oracle Application Server** *Oracle_Home* > **Integrated Management Tools** > **Oracle Directory Manager.**)

2. Log in to Oracle Directory Manager as the `orcladmin` user.
3. In the System Objects frame, expand **Entry Management**, expand **cn=OracleContext**, expand **cn=Products**, expand **cn=IAS**, expand **cn=IAS Infrastructure Databases**, and expand **orclReferenceName=*dbname*** for the Metadata Repository.
4. Select the **OrclResourceName=*schema_name*** entry for the schema whose password you want to change.
5. In the Properties tab, update the password in the **orclpasswordattribute** field.
6. Click **Apply.**

6.5 Changing the Character Set of OracleAS Metadata Repository

You can change the character set of the OracleAS Metadata Repository by following the instructions for changing the character set in the database documentation.

Then, take the following steps to configure the middle-tier and infrastructure to work with OracleAS Metadata Repository after its character set has been changed:

1. Modify the character set of all Database Access Descriptors (DADs) accessing the metadata repository to the new database character set.
   a. Using the Application Server Control Console, navigate to the middle-tier instance home page.
   b. In the System Components section, click **HTTP_Server.**
   c. On the HTTP_Server home page, click **Administration.**
   d. On the HTTP_Server Administration page, click **PL/SQL Properties.** This opens the `mod_plsql` Services page.
   e. Scroll to the DADs section and click the name of the DAD that you want to configure. This opens the Edit DAD page.
   f. In the **NLS Language** field, type in a NLS_LANG value whose character set is the same as the new character set for OracleAS Metadata Repository.
   g. Click **OK.**
   h. Repeat steps e to g for all DADs accessing OracleAS Metadata Repository.
2. Reconfigure the Oracle Ultra Search index as follows:
   a. Connect to OracleAS Metadata Repository as WKSYS and invoke the following SQL script to reconfigure the default cache character set and index preference:

      ```
      (UNIX) ORACLE_HOME/ultrasearch/admin/wk0prefcheck.sql
      (Windows) ORACLE_HOME\ultrasearch\admin\wk0prefcheck.sql
      ```

   b. Connect to OracleAS Metadata Repository as the default user (WKTEST) and invoke the following SQL script:

      ```
      (UNIX) ORACLE_HOME/ultrasearch/admin/wk0idxcheck.sql
      (Windows) ORACLE_HOME\ultrasearch\admin\wk0idxcheck.sql
      ```

      The script asks you to enter the instance name (`WK_INST`). Enter `y` when prompted to proceed with the change.

This script reconfigures the instance (in this case, the default instance). It also truncates the Oracle Text index used by Oracle Ultra Search and you must force a recrawl to rebuild the index.

c. Repeat step b for all Oracle Ultra Search instances that were created before you changed the database character set. Invoke the script as the instance owner, and then force a recrawl of all data sources, if necessary.

3. If you installed OracleAS Portal, you must make sure that it is configured with the same languages as the OracleAS Infrastructure (which includes OracleAS Metadata Repository) to avoid problems with set language requests sent to OracleAS Single Sign-On. To configure additional languages for OracleAS Portal, you run the `ptlang` tool.

   See: The section "Configuring Language Support" in the *Oracle Application Server Portal Configuration Guide* for information on the `ptlang` tool

6.6 Renaming and Relocating OracleAS Metadata Repository Datafiles

When you install OracleAS Metadata Repository, you can choose the location for its datafiles. The default location is *`ORACLE_HOME`*`/oradata/`*`SID`*. After installation, you may want to relocate datafiles to a different directory. For example, you may want to move them to a directory on a file system with more space. Or, you may want to move them to a directory on a different disk for performance reasons. Another thing you may want to do is keep the datafiles in the same directory, but rename them.

This section provides a procedure for renaming or relocating datafiles. You can use this procedure on one or more datafiles, and the datafiles may be in multiple tablespaces.

This procedure applies to:

- The datafiles associated with Oracle Application Server schemas and tablespaces. See Appendix E for a list of schemas.
- The following standard Oracle database datafiles:
  - `drsys01.dbf`
  - `system01.dbf`
  - `temp01.dbf`
  - `users01.dbf`

The following example shows how to relocate two datafiles in two different tablespaces, as follows:

- Relocate the `oca.dbf` datafile in the `OCATS` tablespace from `/infra_home/oradata/orcl/oca.dbf` to `/new_directory/oca.dbf`
- Relocate the `dcm.dbf` datafile in the DCM schema from `/infra_home/oradata/orcl/dcm.dbf` to `/new_directory/dcm.dbf`

Before you start the procedure:

- Make sure you have a complete cold backup of OracleAS Metadata Repository. See Part V, "Backup and Recovery" for more information.
- Connect to OracleAS Metadata Repository as a user with administrator privileges. You must have the ALTER DATABASE system privilege to relocate datafiles.
- Read through the entire procedure before you start.

The procedure is as follows:

1. Verify the location of your datafiles.

   You can verify the location of datafiles in a particular tablespace by querying the data dictionary view DBA_DATA_FILES.

   For example, to query the location of datafiles in the OCATS and DCM tablespaces:

```
SQL> SELECT FILE_NAME, BYTES FROM DBA_DATA_FILES
WHERE TABLESPACE_NAME = 'OCATS' OR TABLESPACE_NAME = 'DCM';

FILE_NAME                                        BYTES
-----------------------------------------------  ------------
/infra_home/oradata/orcl/oca.dbf                 78643200
/infra_home/oradata/orcl/dcm.dbf                 96993280
```

2. Shut down all middle-tier instances that use OracleAS Metadata Repository.
3. Stop the Infrastructure that contains OracleAS Metadata Repository, then start an OracleAS Metadata Repository instance and mount the database without opening it, as follows:
   a. Stop the Application Server Control Console and OPMN-managed processes:
      * On UNIX systems:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
```

      * On Windows systems:

```
ORACLE_HOME\bin\emctl stop iasconsole
ORACLE_HOME\opmn\bin\opmnctl stopall
```

   b. Leave the Metadata Repository listener running.
   c. Stop the OracleAS Metadata Repository instance (make sure the ORACLE_HOME environment variable is set):

```
sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> SHUTDOWN
```

   d. Start an OracleAS Metadata Repository instance and mount the database without opening it:

```
SQL> STARTUP MOUNT
```

4. Move the datafiles to their new location using the operating system. For example:
   - On UNIX systems:

```
mv /infra_home/oradata/orcl/oca.dbf /new_directory/oca.dbf
mv /infra_home/oradata/orcl/dcm.dbf /new_directory/dcm.dbf
```

   - On Windows systems:

```
rename C:\infra_home\oradata\orcl\oca.dbf D:\new_directory\oca.dbf
rename C:\infra_home\oradata\orcl\dcm.dbf D:\new_directory\dcm.dbf
```

   Note: You can execute an operating system command to copy a file by using the SQL*Plus `HOST` command.

5. Use ALTER DATABASE to rename the file pointers in the database's control file:

```
SQL> ALTER DATABASE
      RENAME FILE          '/infra_home/oradata/orcl/oca.dbf',
                           '/infra_home/oradata/orcl/dcm.dbf'
      TO                   '/new_directory/oca.dbf',
                           '/new_directory/dcm.dbf';
```

   The new files must already exist; this statement does not create the files. Always provide complete filenames (including their full paths) to properly identify the old and new datafiles. In particular, specify the old datafile name exactly as it appears in the DBA_DATA_FILES view of the data dictionary.

6. Shut down OracleAS Metadata Repository, then perform a normal startup of the Infrastructure:

   a. Leave the OracleAS Metadata Repository listener running.

   b. Shut down OracleAS Metadata Repository:

```
SQL> SHUTDOWN
```

   c. Start OracleAS Metadata Repository:

```
SQL> STARTUP
```

   d. Start OPMN-managed processes and the Application Server Control Console:

      – On UNIX systems:

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

      – On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl startall
ORACLE_HOME\bin\emctl start iasconsole
```

7. Start the middle-tier instances that use the Infrastructure.

8. Verify the new location of your datafiles.

```
SQL> SELECT FILE_NAME, BYTES FROM DBA_DATA_FILES
WHERE TABLESPACE_NAME = 'OCATS' OR TABLESPACE_NAME = 'DCM';

FILE_NAME                                      BYTES
---------------------------------------------- ---------------
/new_directory/oca.dbf                         78643200
/new_directory/dcm.dbf                         96993280
```

9. Perform a complete cold backup of OracleAS Metadata Repository. After making any structural changes to a database, always perform an immediate and complete backup.

   See Also: Part V, "Backup and Recovery"

Part III

Advanced Administration

This part describes advanced administration tasks that involve reconfiguring Oracle Application Server.

It contains the following chapters:

7

Reconfiguring Application Server Instances

When you installed Oracle Application Server, you chose an installation type and the components you wanted to configure. For J2EE and Web Cache installations, you could choose if you wanted to use Infrastructure Services. After installation, you may want to add or delete components, or even change the installation type. Or, you may want to start using Infrastructure Services with your J2EE and Web Cache installation. This chapter describes how to make these types of changes.

It contains the following topics:

If you have disabled anonymous binds in Oracle Internet Directory, you must enable them before you make configuration changes. See Section 7.6, "Disabling and Enabling Anonymous Binds" for more information.

7.1 Expanding a Middle-Tier Installation

There are three types of middle-tier installations. The types are ordered in that each contains all of the components in the previous installation type, plus additional components. The installation types, in order from lowest to highest, are:

- J2EE and Web Cache
- Portal and Wireless (includes all components in J2EE and Web Cache)
- Business Intelligence and Forms (includes all components in J2EE and Web Cache, Portal and Wireless)

When you installed Oracle Application Server, you chose an installation type based on the components you required at the time. You may decide later that you want to use additional components that are available in a higher installation type. For example, you may have installed a J2EE and Web Cache, and then decide later that you want to use OracleAS Portal.

To accomplish this, you can expand your application server installation by installing a higher installation type in the same Oracle home using Oracle Universal Installer. Options for expanding a middle-tier installation are shown in Table 7–1.

Table 7–1 Options for Expanding a Middle-Tier Installation

You can expand this type of installation:	To this type of installation:	Result:
J2EE and Web Cache	Portal and Wireless	You are given the option of configuring Portal and Wireless.
Portal and Wireless	Business Intelligence and Forms	You are given the option of configuring Discoverer, Forms, Reports, and Personalization.

When you expand an installation:

- All of your current configured components are maintained.
- The disk files for the additional components in the higher installation type are installed in your Oracle home.
- You are given the option of configuring any of the additional components in the higher installation type.

Note the following:

- You must configure OracleAS Web Cache before you expand a J2EE and Web Cache instance. If you did not configure OracleAS Web Cache during installation, see Section 7.2.1, "Configuring OracleAS Web Cache After Installation" for instructions.
- You cannot reduce an installation by installing a lower installation type in the same Oracle home. For example, you cannot install a J2EE and Web Cache installation in an Oracle home that contains a Portal and Wireless installation. If you want to exclude certain components from your installation, you can disable them. See Section 3.4, "Enabling and Disabling Components" for more information.
- You can only expand middle-tier installations; you cannot expand an Infrastructure installation.

> **See Also:** *Oracle Application Server Installation Guide* for complete instructions on expanding a middle-tier installation

7.2 Configuring Additional Components After Installation

When you installed Oracle Application Server, you were allowed to select the components you wanted to configure. You may decide later you want to configure one of the components you did not select during installation. For example, if you installed J2EE and Web Cache and did not choose to configure OracleAS Web Cache, you can configure OracleAS Web Cache after installation.

You can configure components after installation using the Select Component page in the Application Server Control Console. From the Home page, click **Configure Component.** The Select Component page is displayed, as shown in Figure 7–1.

Figure 7–1 Configuring Components with Application Server Control Console

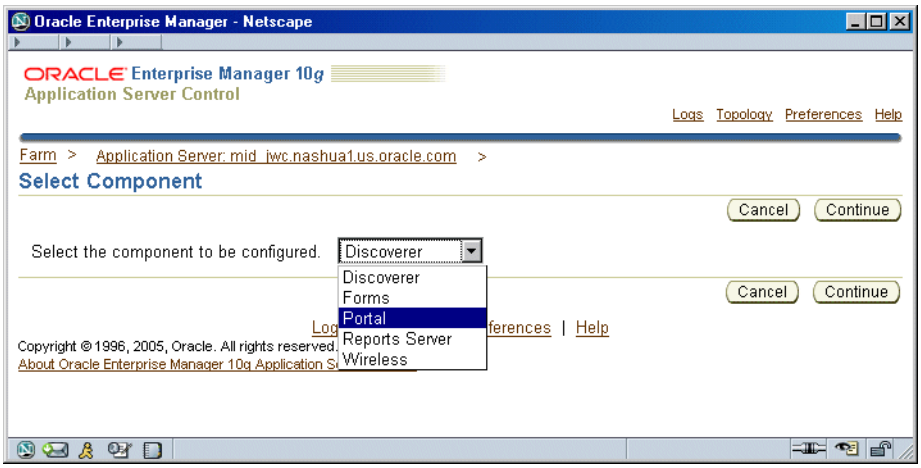


Table 7–2 lists which components can be configured after installation and provides pointers to instructions for using Application Server Control Console to configure and verify each component.

Table 7–2 Components That Can Be Configured After Installation

To configure this component:	In these Installation types:	For instructions, refer to:
Web Cache	J2EE and Web Cache	Section 7.2.1
Portal	Portal and Wireless Business Intelligence and Forms	Section 7.2.2
Wireless	Portal and Wireless Business Intelligence and Forms	Section 7.2.3
Discoverer	Business Intelligence and Forms	Section 7.2.4
Forms	Business Intelligence and Forms	Section 7.2.5
Reports Services	Business Intelligence and Forms	Section 7.2.6
Personalization	Business Intelligence and Forms	Section 7.2.7
Single Sign-On	Infrastructure	Section 7.2.8
Delegated Administration Service	Infrastructure	Section 7.2.9
Directory Integration and Provisioning	Infrastructure	Section 7.2.10

7.2.1 Configuring OracleAS Web Cache After Installation

This section describes how to configure OracleAS Web Cache after installation.

7.2.1.1 Things to Know Before You Start

During installation, port numbers were reserved for OracleAS Web Cache services. You can find the port numbers in the following file:

```
(UNIX) ORACLE_HOME/install/portlist.ini
(Windows) ORACLE_HOME\install\portlist.ini
```

The port numbers are listed as:

```
Web Cache HTTP Listen port = port_number
Web Cache HTTP Listen (SSL) port = port_number
Web Cache Administration port = port_number
Web Cache Invalidation port = port_number
Web Cache Statistics port = port_number
```

These port numbers will be used when you configure OracleAS Web Cache. If you want to use different port numbers, you can change them after you configure OracleAS Web Cache.

7.2.1.2 Configuring OracleAS Web Cache

To configure OracleAS Web Cache, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the instance in which you want to configure OracleAS Web Cache.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Web Cache** from the menu. Click **Continue.**
4. On the Login page, in the **Administration Password** field, enter the `ias_admin` password. Click **Finish.**

7.2.1.3 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. Verify that **Web Cache** is listed in the System Components section. It will have a status of Down. Select the check box next to **Web Cache** and click **Start.**
3. When the operation completes, verify that **Web Cache** shows a status of Up. Then, click **Web Cache** and verify that the Web Cache Home page is displayed.
4. On the Web Cache Home page, click **Administration** to set up OracleAS Web Cache.

> **See Also:** *Oracle Application Server Web Cache Administrator's Guide* for basic setup and configuration tasks

7.2.2 Configuring OracleAS Portal After Installation

This section describes how to configure OracleAS Portal after installation.

Note that you must configure OracleAS Web Cache before you expand a J2EE and Web Cache instance. If you did not configure OracleAS Web Cache during installation, see Section 7.2.1 for instructions. If you expand the instance before configuring OracleAS Web Cache, see Section 7.2.2.4.

7.2.2.1 Things to Know Before You Start

Before you configure OracleAS Portal, make sure that the `sqlnet.ora` file contains the following line, and that LDAP is listed in the line:

```
NAMES.DIRECTORY_PATH= (TNSNAMES, LDAP, ONAMES, HOSTNAME)
```

The sqlnet.ora file is located in the following directory:

```
(UNIX) Oracle_Home/network/admin
```

```
(Windows) Oracle_Home\network\admin
```

7.2.2.2 Configuring OracleAS Portal

To configure OracleAS Portal, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the instance in which you want to configure OracleAS Portal.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Portal** from the menu. Click **Continue.**
4. On the Login page, in the **Administration Password** field, enter the `ias_admin` password. Click **Finish.**

7.2.2.3 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. Verify that **OC4J_Portal** and **Portal:portal** are listed in the System Components section.
3. Restart Oracle HTTP Server and start OC4J_Portal:
   a. In the System Components table, select **HTTP_Server**, and click **Restart**.
   b. Select **OC4J_Portal**, and then click **Start**.

      The **home** OC4J instance will be Down after configuring OracleAS Portal through Application Server Control Console. If you wish to start this service, click the **home** component in the System Components section, then click **Start.**
4. Verify that OC4J_Portal and Portal:portal both have a status of Up:
   a. Click **OC4J_Portal** and verify that the OC4J_Portal page is displayed.
   b. Click **Portal:portal** and verify that the Portal page is displayed.

      Initially, the Portal:portal status may be displayed as Down. This is normal. The status should be updated approximately five minutes after configuration.
5. If this is the *first* instance of OracleAS Portal to use this OracleAS Metadata Repository, run the following command in the middle-tier Oracle home (make sure the ORACLE_HOME environment variable is set before you run this command):
   - On UNIX:

     ```
     ORACLE_HOME/portal/conf/ptlconfig -dad portal [-pw PORTAL_schema_password]
     ```

   - On Windows:

     ```
     ORACLE_HOME\portal\conf\ptlconfig -dad portal [-pw PORTAL_schema_password]
     ```

   This script writes OracleAS Portal configuration entries into the OracleAS Metadata Repository. Do not run this script if there are other OracleAS Portal instances using the OracleAS Metadata Repository, because this script will overwrite any existing OracleAS Portal configuration entries in the OracleAS Metadata Repository.

Note: The PORTAL schema password is stored in the Oracle Internet Directory and the entry may be viewed by an administrator using the `oidadmin` utility with the following path under Entry Management:

`OrclResourceName=PORTAL,orclReferenceName=iasdb.`*`myhost.mycompany.com`*`,`
`cn=IAS Infrastructure Databases,cn=IAS,cn=Products,cn=OracleContext`

6. Verify that you can access OracleAS Portal at the following URL:

   `http://`*`hostname.domain`*`:`*`port`*`/pls/portal`

   In the URL, *`hostname.domain`* is the OracleAS Portal host and *`port`* is the OracleAS Web Cache HTTP Listen port for the OracleAS Portal instance. For example:

   `http://myhost.mycompany:7777/pls/portal`

   You can log in to OracleAS Portal as the user `portal`.

   - If this is the first OracleAS Portal instance to use the OracleAS Metadata Repository, the password is the original `ias_admin` password you supplied for this middle tier during installation. The original `ias_admin` password is required, even if you changed the `ias_admin` password after installation.
   - If this is not the first OracleAS Portal instance to use the OracleAS Metadata Repository, the password is either:
     - The original `ias_admin` password for the first middle tier associated with the OracleAS Metadata Repository
     - The current `portal` password, if the administrator changed the `portal` user password after the first OracleAS Portal instance was installed

   See Also: *Oracle Application Server Portal Configuration Guide* for more information on configuring OracleAS Portal

   Note: When OracleAS Portal is configured using Oracle Enterprise Manager, the Oracle Ultra Search instance is not configured automatically and therefore the Ultra Search Administration link in OracleAS Portal will not work. For instructions on how to create an Oracle Ultra Search instance, see *Oracle Ultra Search Administrator's Guide*.

7.2.2.4 Steps Needed If OracleAS Portal Configured Before OracleAS Web Cache

If you expanded the instance before you configured OracleAS Web Cache, you must take the following steps to correct the situation:

1. Configure OracleAS Web Cache, as described in Section 7.2.1.
2. Edit `iasconfig.xml` and change the port that OracleAS Portal is listening on to match that of OracleAS Web Cache. The file is located in:

   `(UNIX)` *`ORACLE_HOME`*`/portal/conf]`
   `(Windows)` *`ORACLE_HOME`*`\portal\conf`
   `]`
3. Run the following command to make Oracle Application Server aware of the change:

- On UNIX:

  ORACLE_HOME/portal/conf/ptlconfig update ptlconfig -dad portal [-pw *PORTAL_schema_password*]

- On Windows:

  ORACLE_HOME\portal\conf\ptlconfig update ptlconfig -dad portal [-pw *PORTAL_schema_password*]

7.2.3 Configuring OracleAS Wireless After Installation

This section describes how to configure OracleAS Wireless after installation.

7.2.3.1 Configuring OracleAS Wireless

To configure OracleAS Wireless, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the instance in which you want to configure OracleAS Wireless.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Wireless**. Click **Continue.**
4. On the Login page:
   - **User Name:** Enter `cn=orcladmin`, or the distinguished name of a user in the `iASAdmins` group.
   - **Password:** Enter the password for the user.

   The **SSL Only** check box is grayed out because you cannot change this feature in this operation.
5. Click **Finish.**

7.2.3.2 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. Verify that OC4J_Wireless and Wireless are listed in the System Components section. OC4J_Wireless and Wireless will have a status of Down. Select the check boxes next to **OC4J_Wireless** and **Wireless,** and click **Start.**
3. When the operation completes, verify that OC4J_Wireless and Wireless have a status of Up. Click **OC4J_Wireless** and verify that the OC4J_Wireless page is displayed. Click **Wireless** and verify that the Wireless page is displayed.
4. Select the check boxes next to **HTTP_Server** and **Web Cache,** and click **Restart.**
5. Verify that you can access OracleAS Wireless at the following URL:

   `http://`*`hostname.domain`*`:`*`port`*`/webtool/login.uix`

   In the URL, *`hostname.domain`* is the OracleAS Wireless host and *`port`* is the OracleAS Web Cache HTTP listen port number for the instance.

   You can log in as the user `orcladmin` with the `orcladmin` password.

See Also: *Oracle Application Server Wireless Administrator's Guide* for more information on configuring OracleAS Wireless

7.2.4 Configuring OracleBI Discoverer After Installation

This section describes how to configure OracleBI Discoverer after installation.

7.2.4.1 Configuring OracleBI Discoverer

To configure OracleBI Discoverer, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the instance in which you want to configure OracleBI Discoverer.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Discoverer.** Click **Continue.**
4. On the Login page, in the **Administration Password** field, enter the `ias_admin` password. Click **Finish.**

7.2.4.2 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. If you did not have an OC4J_BI_Forms instance before, you will have one now. The OC4J_BI_Forms instance will have a status of Down. You will also have a Discoverer instance with a status of Down. Select the check boxes next to **OC4J_BI_Forms** and **Discoverer,** and click **Start.**
3. When the operation completes, verify that OC4J_BI_Forms and Discoverer have a status of Up. Click **OC4J_BI_Forms** and verify that the OC4J_BI_Forms page is displayed. Click **Discoverer** and verify that the Discoverer page is displayed.
4. Select the check box next to **HTTP_Server**, and click **Restart.**
5. Check that OracleBI Discoverer services are started.

   For all of these URLs, *`hostname.domain`* is the host on which Discoverer is installed and *`port`* is the Web Cache HTTP listen port number.

   - Discoverer Viewer:

     `http://`*`hostname.domain`*`:`*`port`*`/discoverer/viewer`

   - Discoverer Plus:

     `http://`*`hostname.domain`*`:`*`port`*`/discoverer/plus`

   - Discoverer Portlet Provider:

     `http://`*`hostname.domain`*`:`*`port`*`/discoverer/portletprovider`

   See Also: *Oracle Business Intelligence Discoverer Configuration Guide* for additional steps for configuring OracleBI Discoverer, including installing OracleBI Discoverer workbooks and End User Layer (EUL) into each database that contains data to be analyzed

7.2.5 Configuring OracleAS Forms Services After Installation

This section describes how to configure OracleAS Forms Services after installation.

7.2.5.1 Configuring OracleAS Forms Services

To configure OracleAS Forms Services, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the instance in which you want to configure OracleAS Forms Services.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Forms.** Click **Continue.**
4. On the Login page:
   - **User Name:** Enter `cn=orcladmin`, or the distinguished name of a user in the `iASAdmins` group.
   - **Password:** Enter the password for the user.

   The **SSL Only** check box is grayed out because you cannot change this feature in this operation.
5. Click **Finish.**

7.2.5.2 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. If you did not have an OC4J_BI_Forms instance before, you will have one now. The OC4J_BI_Forms instance will have a status of Down. You will also have a Forms instance with a status of Down. Select the check box next to **OC4J_BI_Forms** and click **Start.**
3. When the operation completes, verify that OC4J_BI_Forms and Forms have a status of Up. Click **OC4J_BI_Forms** and verify that the OC4J_BI_Forms page is displayed. Click **Forms** and verify that the Forms page is displayed.
4. Verify that you can access OracleAS Forms Services at the following URL:

   `http://`*`hostname.domain`*`:`*`port`*`/forms/frmservlet`

   In the URL, *`hostname.domain`* is the OracleAS Forms Services host and *`port`* is the OracleAS Web Cache HTTP listen port number.

   If you do not have Oracle JInitiator installed on your system, you are prompted to install and run it. Click **Yes,** then follow the directions in the wizard.

   When the page is displayed, try to access the links on this page to verify that the Forms servlet is available.
5. Refer to the OracleAS Forms Services online help for more information on configuring OracleAS Forms Services. Specifically, note that if you want to manage OracleAS Forms Services runtime processes through the Application Server Control Console, the entry `em_mode` in the default section of the Forms Web Configuration must be set to the value `1` (the default is `0`). Also, to view OracleAS Forms Services trace output, the entry for `allow_debug` in that section should be set to `true`.

See Also: *Oracle Application Server Forms Services Deployment Guide*

7.2.6 Configuring OracleAS Reports Services After Installation

This section describes how to configure OracleAS Reports Services after installation.

7.2.6.1 Things to Know Before You Start

During installation, port numbers were reserved for OracleAS Reports Services. You can find the port numbers in the following file:

```
(UNIX) ORACLE_HOME/install/portlist.ini
(Windows) ORACLE_HOME\install\portlist.ini
```

The port numbers are listed as:

```
Reports Services bridge port = 14011
Reports Services discoveryService port = 14021
```

These port numbers will be used when you configure OracleAS Reports Services. If you want to use different port numbers, you can change them after you configure OracleAS Reports Services.

7.2.6.2 Configuring OracleAS Reports Services

To configure OracleAS Reports Services, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the instance in which you want to configure OracleAS Reports Services.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Reports Server.** Click **Continue.**
4. On the Login page:
   - **User Name:** Enter `cn=orcladmin`, or the distinguished name of a user in the `iASAdmins` group.
   - **Password:** Enter the password for the user.

   The **SSL Only** check box is grayed out because you cannot change this feature in this operation.
5. Click **Finish.**

7.2.6.3 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. If you did not have an OC4J_BI_Forms instance before, you will have one now and it will have a status of Down. You will also have a Reports Server:rep_*server* instance with a status of Down. Select the check boxes next to **OC4J_BI_Forms** and **Reports Server: rep_*server*** and click **Start.**
3. When the operation completes, verify that **OC4J_BI_Forms** and **Reports Server: rep_*server*** have a status of Up. Click **OC4J_BI_Forms** and verify that the OC4J_BI_Forms page is displayed. Click **Reports Server: rep_*server*** and verify that the Reports page is displayed.

4. Specify your outgoing mail server.
   a. Edit the following file:
      – On UNIX systems:

        ORACLE_HOME/reports/conf/rep_*server_name*.conf

      – On Windows systems:

        ORACLE_HOME\reports\conf\rep_*server_name*.conf

   b. Uncomment the `pluginParam name="mailServer"` element and update it with the outgoing mail server name. For example, change the following line:

      ```
      <!--pluginParam name="mailServer">%MAILSERVER_NAME%</pluginParam-->
      ```

      To:

      ```
      <pluginParam name="mailServer">smtpserver.myco.com</pluginParam>
      ```

   c. Save and close the file.
5. Verify that OracleAS Reports Services is started, by using the following URL:

   `http://`*`hostname.domain`*`:`*`port`*`/reports/rwservlet/getserverinfo`

   In the URL, *`hostname.domain`* is the OracleAS Reports Services host and *`port`* is the OracleAS Web Cache HTTP listen port number.

   You can log in as `orcladmin` with the `orcladmin` password.

   See Also: *Oracle Application Server Reports Services Publishing Reports to the Web* for more information on configuring and using Reports

7.2.7 Configuring OracleAS Personalization After Installation

To configure OracleAS Personalization. run the OracleAS Personalization Schema Creation Wizard, which creates the required schemas in the Oracle database.

See Also: *Oracle Application Server Personalization Administrator's Guide*

7.2.8 Configuring OracleAS Single Sign-On After Installation

This section describes how to configure OracleAS Single Sign-On after installation.

7.2.8.1 Configuring OracleAS Single Sign-On

To configure OracleAS Single Sign-On, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the Infrastructure instance in which you want to configure OracleAS Single Sign-On.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Single Sign-On Server.** Click **Continue.**
4. On the Login page:
   - **User Name:** Enter `cn=orcladmin`, or the distinguished name of a user in the `iASAdmins` group.

- **Password:** Enter the password for the user.

The **SSL Only** check box is grayed out because you cannot change this feature in this operation.

5. Click **Finish.**

7.2.8.2 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. If you did not have an OC4J_SECURITY instance before, you will have one now. The OC4J_SECURITY instance will have a status of Down. You will also have a Single Sign-On:orasso instance with a status of Down. Select the check box next to **OC4J_SECURITY** and click **Start**.

   Note: You cannot start the Single Sign-On:orasso instance. This feature is started and stopped when you start and stop HTTP_Server and OC4J_SECURITY.
3. When the operation completes, verify that OC4J_SECURITY has a status of Up.

   Note: The Single Sign-On:orasso status may be displayed as Down. This is normal. The status should be updated approximately five minutes after configuration.
4. Select **HTTP_Server** and click **Restart.**
5. Verify that you can access OracleAS Single Sign-On at the following URL:

   `http://`*`hostname.domain`*`:`*`port`*`/pls/orasso`

   In the URL, *`hostname.domain`* is the host on which OracleAS Single Sign-On is installed and *`port`* is the Infrastructure HTTP Server port.

   You can log in as `orcladmin` with the `orcladmin` password.

   See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for more information on configuring OracleAS Single Sign-On

7.2.9 Configuring Oracle Delegated Administration Services After Installation

This section describes how to configure Oracle Delegated Administration Services after installation.

7.2.9.1 Things to Know Before You Start

When you configure Oracle Delegated Administration Services after installation, you will see the following results:

- The URL for Oracle Delegated Administration Services is set up.
- The appropriate privileges are created.
- Oracle Delegated Administration Services are deployed in the OC4J_SECURITY instance.

7.2.9.2 Configuring mod_osso for Oracle Delegated Administration Services

Before you configure Oracle Delegated Administration Services, you must make sure `mod_osso` is configured, as follows:

1. Check if `mod_osso` is configured in the Oracle home where you want to configure Oracle Delegated Administration Services. Examine the following file:

– On UNIX systems:

```
ORACLE_HOME/Apache/Apache/conf/httpd.conf
```

Look for the following line in the file:

```
include "ORACLE_HOME/Apache/Apache/conf/mod_osso.conf"
```

– On Windows systems:

```
ORACLE_HOME\Apache\Apache\conf\httpd.conf
```

Look for the following line in the file:

```
include "ORACLE_HOME\Apache\Apache\conf\mod_osso.conf"
```

ORACLE_HOME refers to the directory where you want to configure Oracle Delegated Administration Services.

If the line starts with `#`, then it is commented out and `mod_osso` is not configured in this installation. Perform step 2 to configure `mod_osso`.

If the line is not commented out, `mod_osso` is already configured. You can proceed and configure Oracle Delegated Administration Services using Application Server Control, as described in Section 7.2.9.3, "Configuring Delegated Administration Service".

2. To configure `mod_osso` manually, perform these steps:

   a. Set the ORACLE_HOME environment variable to the full path of the directory where you want to configure Oracle Delegated Administration Services.

   b. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you have to set depend on the type of your UNIX operating system.

   c. On Windows systems, set the PATH environment variable to contain `%ORACLE_HOME%\lib`.

   d. Run the following command (all on one line). **Note:** for the `-classpath` parameter, do not type any space characters after the colon (:) and semicolon (;) characters, as indicated by *<no spaces>*.

   On UNIX:

```
$ORACLE_HOME/jdk/bin/java
-classpath .:$ORACLE_HOME/sso/lib/ossoreg.jar:<no spaces>
$ORACLE_HOME/jlib/ojmisc.jar:<no spaces>
$ORACLE_HOME/jlib/repository.jar:<no spaces>
$ORACLE_HOME/j2ee/home/jazn.jar:$ORACLE_HOME/jdk/lib/dt.jar:<no spaces>
$ORACLE_HOME/jdk/lib/tools.jar:$ORACLE_HOME/jlib/infratool.jar
oracle.ias.configtool.UseInfrastructure i
 -f $ORACLE_HOME/config/infratool_mod_osso.properties
 -h OIDhost -p OIDport -u OIDadminName -w OIDclearTextPassword
 -o ORACLE_HOME -m ASinstanceName
 -infra infraGlobalDBName -mh host -sso true
 -sslp sslPort -sslf false
```

   On Windows:

```
%ORACLE_HOME%\jdk\bin\java
-classpath .;%ORACLE_HOME%\sso\lib\ossoreg.jar;<no spaces>
%ORACLE_HOME%\jlib\ojmisc.jar;<no spaces>
```

```
%ORACLE_HOME%\jlib\repository.jar;<no spaces>
%ORACLE_HOME%\j2ee\home\jazn.jar;<no spaces>
%ORACLE_HOME%\jdk\lib\dt.jar;<no spaces>
%ORACLE_HOME%\jdk\lib\tools.jar;%ORACLE_HOME%\jlib\infratool.jar
oracle.ias.configtool.UseInfrastructure i
 -f %ORACLE_HOME%\config\infratool_mod_osso.properties
 -h OIDhost -p OIDport -u OIDadminName -w OIDclearTextPassword
 -o ORACLE_HOME -m ASinstanceName
 -infra infraGlobalDBName -mh host -sso true
 -sslp sslPort -sslf false
```

Table 7–3 describes the parameters that require values.

Table 7–3 Parameters for Configuring mod_osso

Parameter	Description[1]
-h *OIDhost*	Specifies the name of the computer where Oracle Internet Directory is running. You can determine this value from the OIDhost parameter in the *ORACLE_HOME*/config/ias.properties file.
-p *OIDport*	Specifies the port number on which Oracle Internet Directory is listening. You can determine this value from the OIDport parameter in the *ORACLE_HOME*/config/ias.properties file.
-u *OIDadminName*	Specifies the login name for Oracle Internet Directory. Use the superuser: cn=orcladmin.
-w *OIDclearTextPassword*	Specifies the password for the Oracle Internet Directory user.
-o *ORACLE_HOME*	Specifies the full path to the directory where you installed OracleAS Infrastructure 10*g*.
-m *ASinstanceName*	Specifies the name of the OracleAS Infrastructure 10*g* instance where you want to configure mod_osso. You can determine this value from the IASname parameter in the *ORACLE_HOME*/config/ias.properties file.
-infra *infraGlobalDBname*	Specifies the name of the OracleAS Metadata Repository database. You can determine this value from the InfrastructureDBCommonName parameter in the *ORACLE_HOME* /config/ias.properties file.
-mh *host*	Specifies the full hostname (including the domain name) of the computer where you want to configure Oracle Delegated Administration Services.
-sslp *sslPort*	Specifies the SSL port for Oracle Internet Directory. You can determine this value from the OIDsslport parameter in the *ORACLE_HOME*/config/ias.properties file.

[1] Paths are shown in UNIX format; invert the slashes for Windows.

3. If you needed to perform the previous step, restart OC4J and Oracle HTTP Server, using the opmnctl command:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl restartproc ias-component=OC4J
     ORACLE_HOME/opmn/bin/opmnctl restartproc ias-component=HTTP_Server
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl restartproc ias-component=OC4J
     ORACLE_HOME\opmn\bin\opmnctl restartproc ias-component=HTTP_Server
     ```

7.2.9.3 Configuring Delegated Administration Service

To configure Oracle Delegated Administration Services, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the Infrastructure instance in which you want to configure Oracle Delegated Administration Services.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Delegated Administration Service.** Click **Continue.**
4. On the Login page:
   - **User Name:** Enter `cn=orcladmin`.
   - **Password:** Enter the password for the user.

   The **SSL Only** check box is grayed out because you cannot change this feature in this operation.
5. Click **Finish.**

7.2.9.4 Post-Configuration Tasks

When the configuration is finished, take the following steps:

1. In the Application Server Control Console Confirmation page, click **OK.** The Application Server Home page appears.
2. If you did not have an OC4J_SECURITY instance before, you will have one now. The OC4J_SECURITY instance will have a status of Down. Select the check box next to **OC4J_SECURITY** and click **Start.**
3. Select **HTTP_Server** and click **Restart.** Stop and restart all components, using the `opmnctl` command:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ORACLE_HOME\opmn\bin\opmnctl startall
     ```

4. Verify that Oracle Delegated Administration Services is started by navigating to the following URL:

   ```
   http://hostname.domain:port/oiddas
   ```

   In the URL, *`hostname.domain`* is the host on which Oracle Delegated Administration Services is installed and *`port`* is the Infrastructure HTTP Server port.

   See Also: *Oracle Internet Directory Administrator's Guide* for more information on configuring Oracle Delegated Administration Services

7.2.10 Configuring Directory Integration and Provisioning After Installation

To configure Directory Integration and Provisioning after installation, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the Infrastructure instance in which you want to configure Directory Integration and Provisioning.
2. On the Application Server Home page, in the System Components section, click **Configure Component.**
3. On the Select Component page, select **Directory Integration and Provisioning.** Click **Continue.**
4. On the Login page:
   - **User Name:** Enter `cn=orcladmin`.
   - **Password:** Enter the password for the user.

   The **SSL Only** check box is grayed out because you cannot change this feature in this operation.
5. Click **Finish.**
6. When the configuration is finished, click **OK.** The Application Server Home page appears.

See Also: *Oracle Internet Directory Administrator's Guide* for more information on configuring Directory Integration and Provisioning

7.3 Deconfiguring Components

You can configure components at the following times:

- During installation, by selecting the component on the Select Configuration Options screen on Oracle Universal Installer
- After installation, using the Configure Component page of the Application Server Control Console
- When expanding an installation, by selecting the component on the Select Configuration Options screen in Oracle Universal Installer

After you have configured a component, you cannot deconfigure it. An alternative is to disable the component, which prevents it from starting when you start your application server instance. It also removes the component from the System Components list in the Application Server Control Console, and from the `opmnctl status` output. Section 3.4, "Enabling and Disabling Components" describes how to disable a component.

7.4 Deleting OC4J Instances

Guidelines for deleting OC4J instances are as follows:

- You cannot delete OC4J instances that were created by Oracle Application Server during installation.

  These include `home`, `OC4J_BI_FORMS`, `OC4J_Portal`, `OC4J_Wireless`, and `OC4J_SECURITY`. An alternative is to disable an OC4J instance, which prevents it from starting when you start your application server instance. It also removes the component from the System Components list on the Application Server Control Console, and from the `opmnctl status` output.

See Also: Section 3.4, "Enabling and Disabling Components"

- You can delete OC4J instances that were created by a user after installation.

  Deleting these instances removes all applications deployed to the instance. You can delete an OC4J instance using `dcmctl` or the Application Server Control Console.

  To delete an OC4J instance using dcmctl:

  ```
  (UNIX) ORACLE_HOME/dcm/bin/dcmctl removeComponent -co OC4J_instance_name
  (Windows) ORACLE_HOME\dcm\bin\dcmctl removeComponent -co OC4J_instance_name
  ```

  For example, on UNIX:

  ```
  ORACLE_HOME/dcm/bin/dcmctl removeComponent -co OC4J_myapps
  ```

  To delete an OC4J instance using the Application Server Control Console:

  1. Navigate to the Application Server Home page for the instance that contains the OC4J instance.
  2. In the System Components section, select the check box for the OC4J instance and click **Delete OC4J Instance.**

7.5 Configuring J2EE and Web Cache to Use Infrastructure Services

When you install a J2EE and Web Cache instance, you have the option of using the following Infrastructure Services:

- Oracle Identity Management

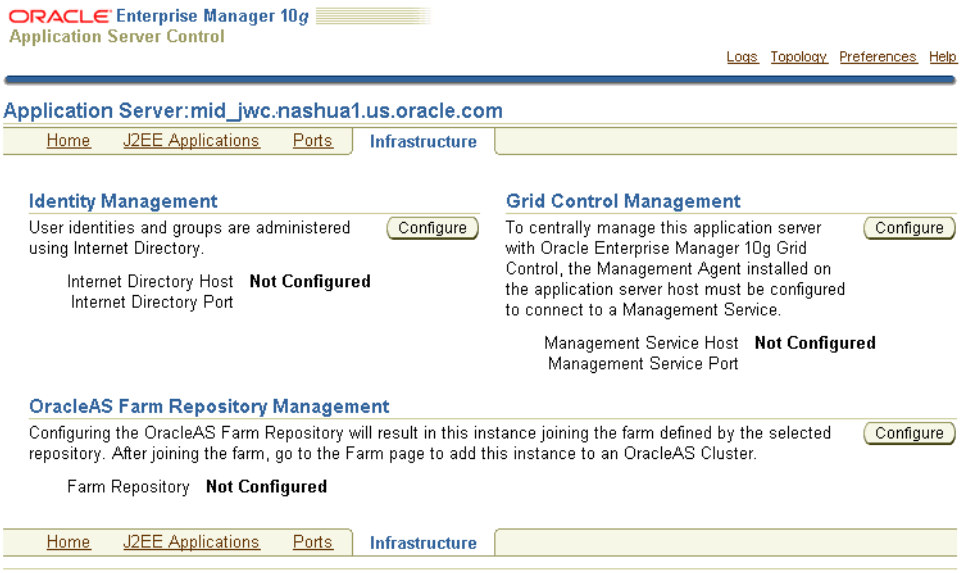
  This enables the J2EE and Web Cache instance to use Single Sign-On services.

- OracleAS Farm Repository Management

  This adds the J2EE and Web Cache instance to the farm of a specified repository, thus enabling it to join an OracleAS Cluster.

If you did not choose these options during installation, you can configure them after installation using the Infrastructure page on the Application Server Control Console, shown in Figure 7–2.

Figure 7–2 Application Server Control Console Infrastructure Page



This section contains the following procedures for configuring a J2EE and Web Cache instance to use Infrastructure services:

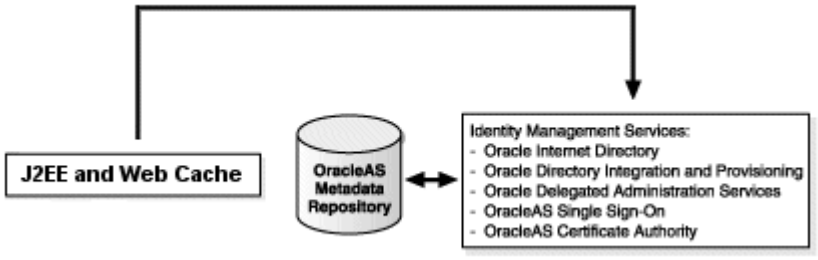
- To configure a J2EE and Web Cache instance to use Oracle Identity Management, refer to Section 7.5.1.
- To configure a J2EE and Web Cache instance to use OracleAS Metadata Repository, refer to Section 7.5.2. Note that the instance must already use Oracle Identity Management.
- To configure a J2EE and Web Cache instance to use an Existing Database (an OracleAS Metadata Repository that is not registered with Oracle Internet Directory used by this instance), refer to Section 7.5.3. Note that the instance may or may not use Oracle Identity Management.
- If you have Oracle Identity Management and OracleAS Metadata Repository, and want to configure a J2EE and Web Cache instance to use OracleAS Metadata Repository only, you can follow the instructions in Section 7.5.4. In this scenario, OracleAS Metadata Repository is registered with Oracle Internet Directory.

  Note that Oracle strongly recommends that you do *not* do this, but instead configure the instance to use Oracle Identity Management and then configure the instance to use the OracleAS Metadata Repository using the instructions in Section 7.5.2.
- To configure a J2EE and Web Cache instance to use an existing file-based repository, refer to Section 7.5.5.
- To configure a J2EE and Web Cache instance to use a new file-based repository, refer to Section 7.5.6.

7.5.1 Configuring Instances to Use Oracle Identity Management

This section describes how to configure a J2EE and Web Cache instance to use Oracle Identity Management, as shown in Figure 7–3.

Figure 7–3 J2EE and Web Cache Using Identity Management



Before you start, make sure that:

- The Oracle Identity Management instance is started (status is Up).
- You know the Oracle Internet Directory host and port numbers.
- You know the password for `cn=orcladmin`, or another user who is a member of the `iASAdmins` group.

Then, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the J2EE and Web Cache instance.

2. Click **Infrastructure.**
3. On the Infrastructure page, in the Identity Management section, click **Configure.**
4. On the Internet Directory page:
   - **Host:** Enter the fully-qualified name of the Oracle Internet Directory host.
   - **Port:** If you do not check **Use only SSL connections with Internet Directory**, enter the non-SSL Oracle Internet Directory port number. Otherwise, enter the SSL Oracle Internet Directory port number.
   - **Use only SSL connections with Internet Directory:** By default, some middle-tier components connect to Oracle Internet Directory using non-SSL connections. If you want components to connect only to Oracle Internet Directory using SSL, check this box and make sure you entered the SSL Oracle Internet Directory port number in the **Port** field.

     Note: If you enter an SSL port number and inadvertently do not select **Use only SSL connections with Internet Directory**, the SSL port number takes precedence and connections to the Oracle Internet Directory are limited to secure connections only.

     If this is not the behavior you intended, you can return to the Identity Management Wizard, enter a non-SSL port in the **Port** field, and make sure that **Use only SSL connections with Internet Directory** is cleared.

   Click **Next.**
5. On the Login page:
   - **User Name:** Enter `cn=orcladmin`, or the distinguished name of a user in the `iASAdmins` group.
   - **Password:** Enter the password for the user.

   Click **Next.**
6. On the Validation page, you receive informational messages regarding the validation of this operation and a warning that all of the components in the instance will be stopped. If you receive any error message, follow the instructions for investigating them. Otherwise, if the operation is valid, click **Finish.**
7. When the operation is finished, you must restart the components in the J2EE and Web Cache instance.
   a. Click **Home** to navigate to the Home page for the middle-tier instance.
   b. Click **Start All.**

Your J2EE and Web Cache instance is now configured to use Oracle Identity Management services.

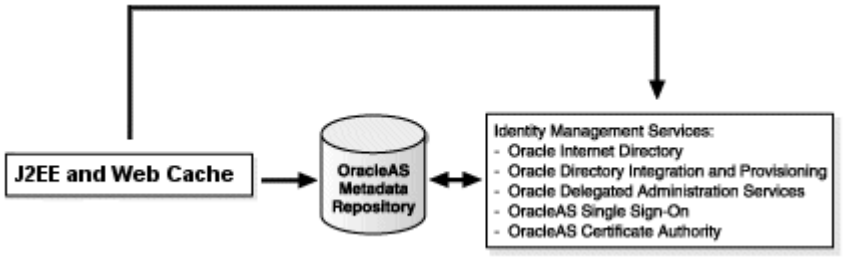
See Also: *Oracle Identity Management Concepts and Deployment Planning Guide*

7.5.2 Configuring Instances with Oracle Identity Management to Use OracleAS Metadata Repository

This section describes how to configure a J2EE and Web Cache instance to use OracleAS Metadata Repository. This procedure requires that the J2EE and Web Cache

instance is already using Oracle Identity Management, and OracleAS Metadata Repository is registered with that Oracle Identity Management, as shown in Figure 7–4.

Figure 7–4 J2EE and Web Cache (with Identity Management) Using OracleAS Metadata Repository



Before you start, make sure that:

- OracleAS Metadata Repository is started (status is Up).
- The Oracle Identity Management instance is started (status is Up).
- You know the password for `cn=orcladmin`, or another user who is a member of the `iASAdmins` group.

Then, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the J2EE and Web Cache instance.
2. Click **Infrastructure.**
3. On the Infrastructure page, in the OracleAS Farm Repository Management section, click **Configure.**
4. On the Source page, choose **OracleAS Metadata Repository.** Then, click **Next.**
5. On the Internet Directory page:
   - **User Name:** Enter `cn=orcladmin` or the distinguished name of a user in the `iASAdmins` group.
   - **Password:** Enter the password for the user.

   Notice that **Use Only SSL connections with Internet Directory** is grayed out. This is because you cannot specify this option in this operation.

   Click **Next.**
6. On the Location page, select the OracleAS Metadata Repository you want to use from the **Repository** list. The Default Schema is always DCM. Then, click **Next.**
7. On the Validation page, you receive informational messages regarding the validation of this operation and a warning that components will be stopped. If you receive any error message, follow the instructions for investigating them. Otherwise, if the repository you specified is valid, click **Finish.**
8. When the operation is finished, you must restart the components in the J2EE and Web Cache instance.
   a. Click **Home** to navigate to the Home page for the middle-tier instance.
   b. Click **Start All.**

Your J2EE and Web Cache instance is now in the OracleAS Metadata Repository's farm and can join an OracleAS Cluster in that farm.

> **See Also:** *Oracle Application Server High Availability Guide* for information on creating and using OracleAS Clusters

7.5.3 Configuring Instances to Use an Existing Database as a Repository

This section describes how to configure a J2EE and Web Cache instance to use an **Existing Database** (an OracleAS Metadata Repository that is not registered with Oracle Internet Directory) as the metadata repository. The J2EE and Web Cache instance may use Oracle Identity Management, as shown in Figure 7–5, or it may not, as shown in Figure 7–6.

Figure 7–5 J2EE and Web Cache (with Identity Management) Using an Existing Database

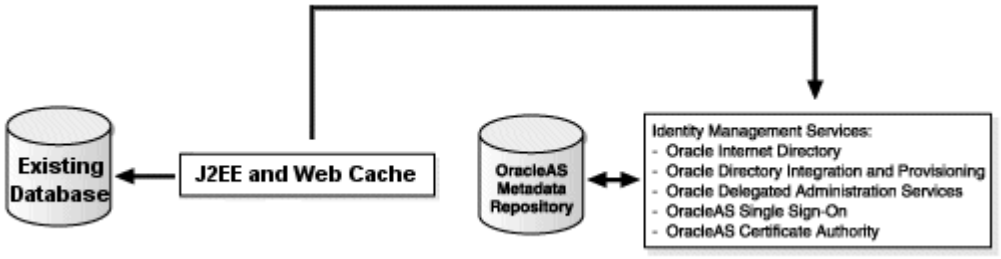


Figure 7–6 shows the J2EE and Web Cache instance using an existing database as a metadata repository, but not using Oracle Identity Management.

Figure 7–6 J2EE and Web Cache (Without Identity Management) Using an Existing Database

Before you start, make sure that:

- The Existing Database is started (status is Up).
- You know the Net listener port and the service name for the Existing Database. These are listed in the entry for the Existing Database in the following file:
  - On UNIX systems:

    ```
    EXISTING_DB_ORACLE_HOME/network/admin/tnsnames.ora
    ```

  - On Windows systems:

    ```
    EXISTING_DB_ORACLE_HOME\network\admin\tnsnames.ora
    ```

- You know the password for the DCM schema in the database.

  If you have just installed the Existing Database and have not used the DCM schema yet, note that the password is generated randomly during installation. To

change the random password to a known value, use the ALTER USER command in SQL*Plus, as shown in the following example (be sure to set the ORACLE_HOME and ORACLE_SID environment variables first):

```
sqlplus "SYS/sys_password as SYSDBA"
SQL> ALTER USER dcm IDENTIFIED BY new_password;
```

Then, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the J2EE and Web Cache instance.
2. Click **Infrastructure.**
3. On the Infrastructure page, in the OracleAS Farm Repository Management section, click **Configure.**
4. On the Source page, choose **Existing Database.** (Note: If the OracleAS Metadata Repository option is grayed out, it is because the J2EE and Web Cache instance is not using Oracle Identity Management). Then, click **Next.**
5. On the Login page, fill in the following fields:
   - **User Name:** `DCM`.
   - **Password:** Enter the DCM schema password.
   - **Hostname** and **Port:** Enter the hostname and Net listener port for the Existing Database. For example: `myhost:1521`.
   - **Service Name:** Enter the service name for the Existing Database. For example, `orcl.myco.com`.

   Click **Next.**
6. On the Validation page, you receive informational messages regarding the validation of this operation. If you receive any error message, follow the instructions for investigating them. Otherwise, if the operation is valid, click **Finish.**
7. When the operation is finished, you must restart the components in the J2EE and Web Cache instance.
   a. Click **Home** to navigate to the Home page for the instance.
   b. Click **Start All.**

Your J2EE and Web Cache instance is now in the Existing Database's farm and can join an OracleAS Cluster in that farm.

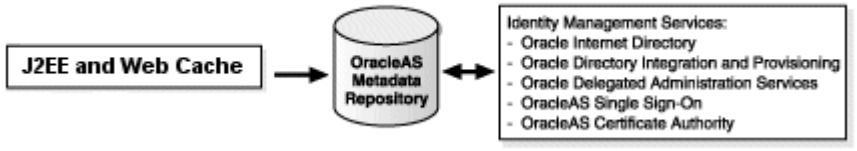
7.5.4 Configuring Instances Without Oracle Identity Management to Use OracleAS Metadata Repository

This section describes how to configure a J2EE and Web Cache instance to use OracleAS Metadata Repository for its metadata. This procedure assumes that OracleAS Metadata Repository is registered with Oracle Internet Directory and that the J2EE and Web Cache instance is not using Oracle Identity Management.

Caution: This configuration is not recommended. Instead, Oracle recommends that you register the J2EE and Web Cache with Oracle Identity Management (see Section 7.5.1) and then configure it to use OracleAS Metadata Repository (see Section 7.5.2).

This configuration is shown in Figure 7–7.

Figure 7–7 J2EE and Web Cache (Without Identity Management) Using OracleAS Metadata Repository



Before you start, make sure that:

- OracleAS Metadata Repository is started (status is Up)
- You know the password for the DCM schema in the database

  If you have just installed the Existing Database and have not used the DCM schema yet, note that the password is generated randomly during installation. To change the random password to a known value, use the ALTER USER command in SQL*Plus, as shown in the following example (be sure to set the ORACLE_HOME and ORACLE_SID environment variables first):

  ```
  sqlplus "SYS/sys_password as SYSDBA"
  SQL> ALTER USER dcm IDENTIFIED BY new_password;
  ```

  If the schema is already in use, use the current password. To find the current password, use the following command:

  ```
  SELECT password FROM dba_users WHERE username='DCM';
  ```

- The Oracle Identity Management instance to which OracleAS Metadata Repository is registered is started (status is Up)
- You know the password for `cn=orcladmin`, or another user who is a member of the `iASAdmins` group

Then, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the J2EE and Web Cache instance.
2. Click **Infrastructure.**
3. On the Infrastructure page, in the OracleAS Farm Repository Management section, click **Configure.**
4. On the Source page, choose **Existing Database.** (Note: The OracleAS Metadata Repository option is grayed out because the J2EE and Web Cache instance is not using Oracle Identity Management). Then, click **Next.**
5. On the Login page, fill in the following fields:
   - **User Name:** Enter `DCM`.
   - **Password:** Enter the DCM schema password.
   - **Hostname** and **Port:** Enter the hostname and Net listener port for the Existing Database. For example: `myhost:1521`.
   - **Service Name:** Enter the service name for OracleAS Metadata Repository. For example, `orcl.myco.com`.

Click **Next.**

6. On the Validation page, you receive informational messages regarding the validation of this operation. If you receive any error message, follow the instructions for investigating them. Otherwise, if the operation is valid, click **Finish.**
7. When the operation is finished, you must restart the components in the J2EE and Web Cache instance.
   a. Click **Home** to navigate to the Home page for the instance.
   b. Click **Start All.**

Your J2EE and Web Cache instance is now in the OracleAS Metadata Repository's farm and can join an OracleAS Cluster in that farm.

7.5.5 Configuring Instances to Use an Existing File-Based Repository

This section describes how to configure a J2EE and Web Cache instance to use an existing file-based repository. The instance does not use Oracle Identity Management or OracleAS Metadata Repository.

Before you start, make sure that:

- The instance that contains the file-based repository is started (status is Up).
- You know the **File-Based Repository ID** for the farm. The ID can be found on the Infrastructure page of an instance already in the farm.

Then, take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the J2EE and Web Cache instance.
2. Click **Infrastructure.**
3. On the Infrastructure page, in the OracleAS Farm Repository Management section, click **Configure.**
4. On the Source page, choose **Existing file-based repository.** Then, click **Next.**
5. In the Location page, enter the **File-Based Repository ID** for the farm. The ID can be found on the Infrastructure page of an instance already in the farm. Then, click **Next.**
6. The Validation page displays informational messages regarding the validation of this operation. If you receive any error message, follow the instructions for investigating them. Otherwise, if the repository you specified is valid, click **Finish.**
7. When the operation is finished, you must restart the components in the J2EE and Web Cache instance.
   a. Click **Home** to navigate to the Home page for the instance.
   b. Click **Start All.**

7.5.6 Configuring Instances to Use a New File-Based Repository

This section describes how to configure a J2EE and Web Cache instance to create and use a new file-based repository. The instance does not use Oracle Identity Management or OracleAS Metadata Repository.

Take the following steps:

1. Using the Application Server Control Console, navigate to the Application Server Home page for the J2EE and Web Cache instance.
2. Click **Infrastructure.**
3. On the Infrastructure page, in the OracleAS Farm Repository Management section, click **Configure.**
4. On the Source page, choose **New file-based repository.** Then, click **Next.**
5. The Validation page displays informational messages regarding the validation of this operation. If you receive any error message, follow the instructions for investigating them. Otherwise, if the repository you specified is valid, click **Finish.**
6. When the operation is finished, you must restart the components in the J2EE and Web Cache instance.
   a. Click **Home** to navigate to the Home page for the instance.
   b. Click **Start All.**

7.6 Disabling and Enabling Anonymous Binds

Beginning with Release 2 (10.1.2.0.2), you can enable and disable anonymous binds (anonymous authentication) in Oracle Internet Directory. By default, anonymous binds are enabled.

Although disabling anonymous binds is useful in many runtime environments, most configuration changes, such as the following, require that anonymous binds are enabled:

- Installing new components with Oracle Universal Installer
- Configuring components with Application Server Control Console
- Changing the host name, domain name, or IP address of a host on which you have installed Oracle Application Server
- Cloning

7.6.1 Disabling Anonymous Binds for RunTime Environments

To disable anonymous binds, take the following steps:

1. Shut down all middle tiers that are connected to the OracleAS Infrastructure, as described in Section 3.2.4, "Stopping a Middle-Tier Instance".
2. Shut down OracleAS Infrastructure, in all Infrastructure Oracle homes:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
   (Windows) ORACLE_HOME\opmn\bin\opmnctl stopall
   ```

3. Start Oracle Internet Directory, because it must be started while you perform the procedure:

   ```
   (UNIX) ORACLE_HOME/bin/oidmon connect=db_connect_string start
   (Windows) ORACLE_HOME\bin\oidmon connect=db_connect_string start
   ```

4. Edit the `ias.properties` file for each middle tier connected to the OracleAS Infrastructure and for the Infrastructure Oracle home that contains OracleAS Single Sign-On and Oracle Delegated Administration Services. The `ias.properties` file is located in the following directory:

   ```
   (UNIX) ORACLE_HOME/config
   ```

(Windows) *ORACLE_HOME*\config

In the `ias.properties` file, add the `OIDAnonymousDisabled` property to the file and set it to `true`:

```
OIDAnonymousDisabled=true
```

5. Edit the `dads.conf` file for each middle tier connected to the OracleAS Infrastructure and for the Infrastructure Oracle home that contains OracleAS Single Sign-On and Oracle Delegated Administration Services. The `dads.conf` file is located in the following directory:

   (UNIX) *ORACLE_HOME*/Apache/modplsql/conf
   (Windows) *ORACLE_HOME*\Apache\modplsql\conf

   By default, the PlsqlDatabaseConnectString parameter contains a value that uses the LDAP name resolution format, for example:

   ```
   PlsqlDatabaseConnectString cn=orcl, cn=oraclecontext NetServiceNameFormat
   ```

   Comment out this line. (Do not delete it because you will need to revert to it if you want to enable anonymous binds in the future.)

   Add the following line, which changes the value of the PlsqlDatabaseConnectString parameter to use the `host:port:service` format instead of LDAP name resolution:

   ```
   PlsqlDatabaseConnectString db_host:db_hostdb_listener_port:db_service_name
   ```

   In the example, *db_host* is the name of the host on which the OracleAS Metadata Repository for OracleAS Single Sign-On is installed, *db_listener_port* is the listener port for that OracleAS Metadata Repository, and *db_service_name* is the service name for the OracleAS Metadata Repository.

6. Use the `ldapmodify` command to disable anonymous binds. Use the command on the Oracle home that contains Oracle Internet Directory.

   Take the following steps:

   a. Create a text file with the following lines:

      ```
      dn:
      changetype: modify
      replace: orclanonymousbindsflag
      orclanonymousbindsflag: 0
      ```

   b. Use the `ldapmodify` command, calling the text file created in the previous step as input. In the following example, the text file is named `anon_off.ldif`:

      (Unix) *ORACLE_HOME*/bin/ldapmodify -h *host* -p *port* -D cn=orcladmin -w *password* -v -f anon_off.ldif
      (Windows) *ORACLE_HOME*\bin\ldapmodify -h *host* -p *port* -D cn=orcladmin -w *password* -v -f anon_off.ldif

7. Stop Oracle Internet Directory:

   (UNIX) *ORACLE_HOME*/bin/oidmon connect=*db_connect_string* stop
   (Windows) *ORACLE_HOME*\bin\oidmon connect=*db_connect_string* stop

8. Start OracleAS Infrastructure, including Oracle Internet Directory, in the Oracle Internet Directory Oracle home, then in any other OracleAS Infrastructure Oracle homes:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME\opmn\bin\opmnctl startall
```

9. Start all middle tiers that are connected to the Infrastructure, as described in Section 3.2.3, "Starting a Middle-Tier Instance".

7.6.2 Enabling Anonymous Binds for Configuration Changes

If you have disabled anonymous binds, you must take the following steps to enable anonymous binds before you can make configuration changes to Oracle Application Server middle tiers or OracleAS Infrastructure:

1. Shut down all middle tiers that are connected to the OracleAS Infrastructure, as described in Section 3.2.4, "Stopping a Middle-Tier Instance".
2. Shut down OracleAS Infrastructure, in all Infrastructure Oracle homes:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
   (Windows) ORACLE_HOME\opmn\bin\opmnctl stopall
   ```

3. Start Oracle Internet Directory, because it must be started while you perform the procedure:

   ```
   (UNIX) ORACLE_HOME/bin/oidmon connect=db_connect_string start
   (Windows) ORACLE_HOME\bin\oidmon connect=db_connect_string start
   ```

4. Edit the `ias.properties` file for each middle tier connected to the OracleAS Infrastructure and for the Infrastructure Oracle home that contains OracleAS Single Sign-On and Oracle Delegated Administration Services. The `ias.properties` file is located in the following directory:

   ```
   (UNIX) ORACLE_HOME/config
   (Windows) ORACLE_HOME\config
   ```

   In the `ias.properties` file, set the `OIDAnonymousDisabled` property to `false`:

   ```
   OIDAnonymousDisabled=false
   ```

   If the property does not exist in the file, or if it is set to `false`, anonymous binds are enabled.

5. Edit the `dads.conf` file for each middle tier connected to the OracleAS Infrastructure and for the Infrastructure Oracle home that contains OracleAS Single Sign-On and Oracle Delegated Administration Services. The `dads.conf` file is located in the following directory:

   ```
   (UNIX) ORACLE_HOME/Apache/modplsql/conf
   (Windows) ORACLE_HOME\Apache\modplsql\conf
   ```

   If you previously commented out the line that contains the PlsqlDatabaseConnectString parameter with a value that uses the LDAP name resolution format, uncomment out that line. If you deleted the line, add a line using the following format:

   ```
   PlsqlDatabaseConnectString cn=orcl, cn=oraclecontext NetServiceNameFormat
   ```

   If you previously added a line similar to the following, which contains the PlsqlDatabaseConnectString parameter with a value that use `host:port:service` format, comment out the line:

   ```
   PlsqlDatabaseConnectString db_host:db_hostdb_listener_port:db_service_name
   ```

6. Use the `ldapmodify` command to enable anonymous binds. Use the command on the Oracle home that contains Oracle Internet Directory.

   Take the following steps:

   a. Create a text file with the following lines:

      ```
      dn:
      changetype: modify
      replace: orclanonymousbindsflag
      orclanonymousbindsflag: 1
      ```

   b. Use the `ldapmodify` command, calling the text file created in the previous step as input. In the following example, the text file is named `anon_on.ldif`:

      ```
      (Unix) ORACLE_HOME/bin/ldapmodify -h host -p port -D cn=orcladmin -w
      password -v -f anon_on.ldif
      (Windows) ORACLE_HOME\bin\ldapmodify -h host -p port -D cn=orcladmin -w
      password -v -f anon_on.ldif
      ```

7. Stop Oracle Internet Directory:

   ```
   (UNIX) ORACLE_HOME/bin/oidmon connect=db_connect_string stop
   (Windows) ORACLE_HOME\bin\oidmon connect=db_connect_string stop
   ```

8. Start OracleAS Infrastructure, including Oracle Internet Directory, in the Oracle Internet Directory Oracle home, then in any other OracleAS Infrastructure Oracle homes:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
   (Windows) ORACLE_HOME\opmn\bin\opmnctl startall
   ```

9. Start all middle tiers that are connected to the Infrastructure, as described in Section 3.2.3, "Starting a Middle-Tier Instance".

8

Changing Network Configurations

This chapter provides procedures for changing the network configuration of an Oracle Application Server host.

It contains the following topics:

8.1 Overview of Procedures for Changing Network Configurations

The following procedures for changing network configurations are presented in this chapter:

- Changing the Hostname, Domain Name, or IP Address

  This section describes how to update Oracle Application Server when changing the hostname, domain name, or IP address of a host.

- Moving Between Off-Network and On-Network

  This section provides procedures for moving an Oracle Application Server host on and off the network. You may use DHCP or a static IP address when on the network. You can use these procedures, for example, if you installed Oracle Application Server on your laptop and want to plug in to different networks to use it.

- Changing Between a Static IP Address and DHCP

  This section provides procedures for changing from a static IP address to DHCP, and from DHCP to a static IP address. You might use these if you install on a static IP address but then decide you want to use DHCP so you can be more mobile, or if you are using DHCP and must plug in to a network using a static IP address.

If you have disabled anonymous binds in Oracle Internet Directory, you must enable them before you make configuration changes. See Section 7.6, "Disabling and Enabling Anonymous Binds" for more information.

8.2 Changing the Hostname, Domain Name, or IP Address

You may want to change the hostname, domain name, or IP address of the host, after you have installed Oracle Application Server. Depending on your installation type, you can perform some or all of these operations.

Many of the procedures in this section use the `chgiphost` command. See Section 8.2.1, "Understanding the chgiphost Command" for more information about the command.

Table 8–1 summarizes the installation types that support hostname, domain name, and IP address changes, and provides pointers to the appropriate procedures.

Table 8–1 Supported Procedures for Hostname, Domain Name, and IP Address Changes

Installation Type	Changing the Hostname or Domain Name	Changing the IP Address
Middle tier	Supported Refer to Section 8.2.2, "Changing the Hostname, Domain Name, or IP Address of a Middle-Tier Installation"	Supported Refer to Section 8.2.2, "Changing the Hostname, Domain Name, or IP Address of a Middle-Tier Installation"
Infrastructure: Identity Management only Identity Management installations with the following components configured: ■ Oracle Internet Directory only ■ OracleAS Single Sign-On, Oracle Delegated Administration Services, and (optionally) Oracle Directory Integration and Provisioning ■ Oracle Internet Directory, OracleAS Single Sign-On, Oracle Delegated Administration Services, and (optionally) Oracle Directory Integration and Provisioning	Supported Refer to Section 8.2.3, "Changing the Hostname, Domain Name, or IP Address of an Identity Management Installation"	Supported Refer to Section 8.2.3, "Changing the Hostname, Domain Name, or IP Address of an Identity Management Installation"
Infrastructure: Identity Management and Metadata Repository	Not supported	Supported Refer to Section 8.2.5, "Changing the IP Address of an Infrastructure Containing a Metadata Repository"
Infrastructure: Metadata Repository only	Not supported	Supported Refer to Section 8.2.5, "Changing the IP Address of an Infrastructure Containing a Metadata Repository"
OracleAS Certificate Authority	Supported Refer to Section 8.2.4, "Changing the Hostname or Domain Name of an OracleAS Certificate Authority Installation"	Supported Simply change the address in your operating system. No updates to Oracle Application Server are required

8.2.1 Understanding the chgiphost Command

The `chgiphost` command-line utility changes the hostname, domain name, or IP address of a middle-tier instance, Infrastructure, or Identity Management installation.

The utility is located at:

- On UNIX systems:

  ORACLE_HOME`/chgip/scripts/chgiphost.sh`

- On Windows systems:

```
ORACLE_HOME\chgip\scripts\chgiphost.bat
```

Table 8–2 shows the options for the command.

Table 8–2 Options for the chgiphost Command

Options	Description
-version	Displays the version of the utility
-infra	Changes the IP address of an Infrastructure instance
-mid	Changes the hostname, domain name, or IP address of a middle-tier instance
-idm	Changes the hostname, domain name or IP address of an Identity Management only instance
-silent	Runs the command in silent mode

Note that if you use `chgiphost` to change the hostname or domain name, it does not update the instance name. For example, assume that the original instance name, with the hostname and domain name appended, is:

```
101202mid.myhost1.mydomain.com
```

If you change the hostname to myhost2, the instance name remains the same.

See Also:

- Section 8.2.6.2, "Setting the Log Level for chgiphost"
- Section 8.2.6.3, "Customizing the chgiphost Command"

8.2.2 Changing the Hostname, Domain Name, or IP Address of a Middle-Tier Installation

This section describes how to change the hostname, domain name, IP address, or any combination of these, of a host that contains any of the following middle-tier installation types:

- J2EE and Web Cache
- Portal and Wireless
- Business Intelligence and Forms

Note: This procedure is not supported for OracleAS Developer Kits.

The following sections describe the procedure:

- Before You Begin
- Task 1: Prepare Your Host
- Task 2: Change the Hostname, Domain Name, or IP Address
- Task 3: Run the chgiphost Command
- Task 4: Restart Your Environment
- Task 5: Update OracleAS Portal, OracleAS Wireless, OracleAS Single Sign-On, and Oracle Ultra Search

- Task 6: Manually Update the Hostname in Files

Before You Begin

Review the following items before you start:

- If any installations contain Oracle Content Management Software Development Kit, you must perform additional steps. Refer to *Oracle Content Management SDK Administrator's Guide* before starting this procedure.
- If the middle-tier instance is registered with Oracle Internet Directory, you must supply the `cn=orcladmin` password during the procedure.
- Consider changing the log level before running the `chgiphost` command so you can view more detailed information. See Section 8.2.6.2, "Setting the Log Level for chgiphost" for more information.
- If your old hostname is a string that is likely to appear in a configuration file, the `chgiphost` command may encounter problems when trying to update the configuration files. Refer to Section 8.2.6.3, "Customizing the chgiphost Command" for information on how to avoid this problem.
- Write down the old hostname and IP address before you begin. You will be prompted for these values.
- Oracle recommends that you perform a backup of your environment before you start this procedure. Refer to Part V, "Backup and Recovery".

Task 1: Prepare Your Host

Prepare your host for the change by removing instances from OracleAS Clusters and stopping all processes:

1. If the host contains a middle-tier instance that is part of an OracleAS Cluster, remove the instance from the OracleAS Cluster. You can add the instance back into the cluster at the end of the procedure.

   See Also: *Oracle Application Server High Availability Guide* for instructions on removing instances from an OracleAS Cluster

2. If the host contains an instance that stores the file-based repository used by an OracleAS File-Based farm, you must remove all instances from that farm, even if they reside on other hosts. This is because the repository ID will change when you change the hostname. As a result, you must remove all instances from the farm, change the hostname (which will change the repository ID), then add the instances back to the farm at the end of this procedure using the new repository ID.

   To remove an instance from an OracleAS File-Based Farm, run the following command in the instance Oracle home:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl leavefarm
   (Windows) ORACLE_HOME\dcm\bin\dcmctl leavefarm
   ```

3. If the host contains a J2EE and Web Cache instance that is part of an OracleAS File-Based Farm that uses a repository on another host or an OracleAS Database-Based Farm, remove the instance from the farm:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl leavefarm
   (Windows) ORACLE_HOME\dcm\bin\dcmctl leavefarm
   ```

   You can add the instance back to the farm at the end of the procedure.

4. If the host contains a middle-tier instance that is part of an OracleAS Web Cache cluster, remove the instance from the cache cluster. You can add the instance back into the cluster at the end of the procedure.

   See Also: *Oracle Application Server Web Cache Administrator's Guide* for instructions on removing caches from a cache cluster

5. Shut down each middle-tier instance on the host by running the following commands in each Oracle home:

   - On UNIX systems:

     ```
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\bin\emctl stop iasconsole
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ```

6. If the middle-tier instance was part of an OracleAS File-Based Farm, make sure the DCM daemon is running in the file-based repository instance. This applies whether the repository instance is on the same host or a different host.

   To verify if the DCM daemon is running, run the following command in the file-based repository Oracle home:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl status
   (Windows) ORACLE_HOME\opmn\bin\opmnctl status
   ```

   To start the DCM daemon:

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=dcm-daemon
   (Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=dcm-daemon
   ```

7. To make sure Oracle Application Server processes will not start automatically after a restart of the host, disable any automated startup scripts you may have set up, such as `/etc/init.d` scripts.
8. Make sure the Oracle Internet Directory that is used by the middle tier is started.

Task 2: Change the Hostname, Domain Name, or IP Address

Update your operating system with the new hostname, domain name, IP address, or any combination of these. Consult your operating system documentation for information on how to perform the following steps.

1. Make the updates to your operating system to properly change the hostname, domain name, or IP address.
2. Restart the host, if necessary for your operating system.
3. Verify that you can ping the host from another host in your network. Be sure to ping using the new hostname to make sure everything is resolving properly.

Task 3: Run the chgiphost Command

Follow these steps for each middle-tier instance on your host. Be sure to complete the steps entirely for one middle-tier instance before you move on to the next.

1. Log in to the host as the user that installed the middle-tier instance.

2. Make sure your ORACLE_HOME environment variable is set to the middle-tier Oracle home. Do not use a trailing slash (UNIX) or backslash (Windows) when specifying the variable.
3. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
4. Run the following commands in the middle-tier Oracle home:
   - On UNIX systems:

     ```
     cd ORACLE_HOME/chgip/scripts
     ./chgiphost.sh -mid
     ```

   - On Windows systems:

     ```
     cd ORACLE_HOME\chgip\scripts
     cmd /c chgiphost.bat -mid
     ```

   The `chgiphost` command prompts for information, as shown in Table 8–3. Note that the prompts may provide values in parentheses. You can enter a different value, or press the return key to accept the suggested value.

Table 8–3 Prompts and Actions for chgiphost -mid

Prompt	Action
`Enter fully qualified hostname (hostname.domainname) of destination`	Enter the new fully-qualified hostname. This may be a new hostname, domain name, or both.
`Enter valid IP Address of destination`	If you changed the IP address of the host, enter the new IP address. Otherwise, enter the current IP address.
`Enter valid IP Address of source`	If you changed the IP address of the host, enter the old IP address. Otherwise, enter the current IP address.
`OIDAdmin Password:`	Enter the `cn=orcladmin` password for the Oracle Internet Directory in which this middle-tier instance is registered.

5. Verify that the tool ran successfully by checking for errors in the files in the following directory:
   - On UNIX systems:

     ```
     ORACLE_HOME/chgip/log
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\chgip\log
     ```

Task 4: Restart Your Environment

Restart the middle-tier instances and restore your configuration to the way it was before you started the procedure:

1. Start each middle-tier instance on your host by running the following commands in each Oracle home:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole
     ```

- On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl startall
ORACLE_HOME\bin\emctl start iasconsole
```

2. If you removed any instances from an OracleAS Web Cache cluster at the beginning of this procedure, add them back to the cache cluster.

   See Also: *Oracle Application Server Web Cache Administrator's Guide* for instructions on adding caches to a cluster

3. If the host contained an instance that stored the file-based repository used by an OracleAS File-Based farm:

   a. Obtain the new repository ID for the new farm by running the following command in the Oracle home of that instance:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl getRepositoryID
   (Windows) ORACLE_HOME\dcm\bin\dcmctl getRepositoryID
   ```

   b. Re-create the OracleAS File-Based Farm by adding that instance to the new farm using the new repository ID obtained in the preceding step. The `repository_ID` is of the form *`hostname:port`*.

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl joinfarm -r repository_ID
   (WINDOWS) ORACLE_HOME\dcm\bin\dcmctl joinfarm -r repository_ID
   ```

   c. Add all instances on other hosts back to the new farm using the command in the preceding step.

4. If you removed any J2EE and Web Cache instances from an OracleAS File-Based Farm (that uses a repository on another host) at the beginning of this procedure, add each one back as follows:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl joinfarm -r repository_ID
   (Windows) ORACLE_HOME\dcm\bin\dcmctl joinfarm -r repository_ID
   ```

   In the preceding command, *`repository_ID`* is the hostname:port value returned by running the following command in the file-based repository Oracle home:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl getRepositoryID
   (Windows) ORACLE_HOME\dcm\bin\dcmctl getRepositoryID
   ```

5. If you removed any J2EE and Web Cache instances from an OracleAS Database-Based Farm at the beginning of this procedure, add each one back as follows:

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl joinfarm
   (Windows) ORACLE_HOME\dcm\bin\dcmctl joinfarm
   ```

6. If you removed any instances from an OracleAS Cluster at the beginning of this procedure, add them back to the cluster.

   See Also: *Oracle Application Server High Availability Guide* for instructions on adding instances to an OracleAS Cluster

7. If you disabled any processes for automatically starting Oracle Application Server at the beginning of this procedure, enable them.

Task 5: Update OracleAS Portal, OracleAS Wireless, OracleAS Single Sign-On, and Oracle Ultra Search

You must update OracleAS Portal, OracleAS Wireless, OracleAS Single Sign-On, and Oracle Ultra Search when you change the hostname.

1. Update OracleAS Portal with the new OracleAS Wireless service URL.

   If you change the hostname, the OracleAS Wireless server URL will also change to use this new hostname. Therefore, you must update OracleAS Portal with the new OracleAS Wireless service URL. For more information, refer to "Updating the OracleAS Wireless Portal Service URL Reference" in *Oracle Application Server Portal Configuration Guide*.

2. Update OracleAS Single Sign-On server with the new OracleAS Wireless SSO Partner URL.

   If you change the hostname, the OracleAS Wireless SSO Partner URL uses the new hostname. Therefore, you must update OracleAS Single Sign-On with the new OracleAS Wireless SSO Partner URL.

   Rather than manually changing the OracleAS Single Sign-On settings, Oracle recommends that you re-register the OracleAS Wireless server with OracleAS Single Sign-On using the following command line tool:

   ```
   (UNIX) ORACLE_HOME/wireless/bin/reRegisterSSO.sh
   (Windows) ORACLE_HOME\wireless\bin\reRegisterSSO.bat
   ```

   This tool, which prompts you through the registration process, not only updates the OracleAS Wireless URL in the OracleAS Single Sign-On server, but it also updates the SSO URL in the OracleAS Wireless server.

3. Re-register OracleAS Portal as an Oracle Ultra Search Content Source.

   If you change the hostname, the OracleAS Portal URL will also change to use this new hostname. Therefore, you must update Oracle Ultra Search with the new OracleAS Portal URL. In Oracle Ultra Search, the OracleAS Portal URL is used to register OracleAS Portal as a crawlable content source. For more information, refer to "Registering OracleAS Portal as a Content Source" in *Oracle Application Server Portal Configuration Guide*.

Task 6: Manually Update the Hostname in Files

If you edited a file and entered the hostname as part of a user-defined parameter such as the Oracle home path, the hostname is not automatically updated by running the `chgiphost` command. To update the hostname in such cases, you must edit the files manually. For example on UNIX, the `plsql.conf` file may contain an NFS path including the hostname, such as: `/net/dsun1/private/...`.

The `chgiphost` command also does not edit the hostname references in the documentation files. You must manually edit these files to update the hostname. Examples of such files are the following files in the *ORACLE_HOME*`/Apache/Apache/htdocs` directory:

- index.html.de
- index.html.es_ES
- index.html.fr
- index.html.it
- index.html.ja

- index.html.ko
- index.html.pt_BR
- index.html.zh_CN
- index.html.zh_TW

8.2.3 Changing the Hostname, Domain Name, or IP Address of an Identity Management Installation

This section describes how to change the hostname, domain name, or IP address on a host that contains an Identity Management installation. This procedure applies to any Identity Management-only installation, including the following:

- Identity Management with only Oracle Internet Directory configured
- Identity Management with OracleAS Single Sign-On and Oracle Delegated Administration Services configured (Oracle Directory Integration and Provisioning is optional)
- Identity Management with Oracle Internet Directory, OracleAS Single Sign-On, and Oracle Delegated Administration Services configured (Oracle Directory Integration and Provisioning is optional)

> **Note:** If your Identity Management installation consists of only OracleAS Certificate Authority, use the procedure described in Section 8.2.4, "Changing the Hostname or Domain Name of an OracleAS Certificate Authority Installation".

The following sections describe the procedure:

Before You Begin

Review the following items before you start the procedure:

- Consider changing the log level before running the `chgiphost` command so you can view more detailed information. See Section 8.2.6.2, "Setting the Log Level for chgiphost" for more information.
- If your old hostname is a string that is likely to appear in a configuration file, the `chgiphost` command may encounter problems when trying to update the configuration files. Refer to Section 8.2.6.3, "Customizing the chgiphost Command" for information on how to avoid this problem.
- Write down the old hostname and IP address before you begin. You will be prompted for these values.

- Oracle recommends that you perform a backup of your environment before you start this procedure. Refer to Part V, "Backup and Recovery" for more information.

Task 1: Shut Down Middle-Tier Instances

For each middle-tier instance that uses Identity Management, stop the Application Server Control Console and the middle-tier instance using the following commands:

- On UNIX systems:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
```

- On Windows systems:

```
ORACLE_HOME\bin\emctl stop iasconsole
ORACLE_HOME\opmn\bin\opmnctl stopall
```

Task 2: Prepare Your Host

Prepare your host for the hostname change by stopping all processes:

1. Set the ORACLE_HOME environment variable.
2. Shut down the Identity Management installation, including the servers, such as Oracle Directory Server, Directory Integration and Provisioning Data server, and Replication Server, and Application Server Control Console. For example, on UNIX, use the following commands:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/bin/oidctl server=odisrv instance=instance_number stop
ORACLE_HOME/bin/oidctl connect=global_db_name server=oidrepld
instance=instance_number stop
ORACLE_HOME/bin/oidctl server=oidldapd instance=instance_number stop
ORACLE_HOME/opmn/bin/opmnctl stopall
```

3. To make sure Oracle Application Server processes will not start automatically after a restart of the host, disable any automated startup scripts you may have set up, such as `/etc/init.d` scripts.

Task 3: Change the Hostname or IP Address

Update your operating system with the new hostname, domain name, or IP address. Consult your operating system documentation for information on how to perform the following steps:

1. Make the updates to your operating system to properly change hostname, domain name, or both.
2. Restart the host, if necessary for your operating system.
3. Verify that you can ping the host from another host in your network. Be sure to ping using the new hostname to make sure everything is resolving properly.

Task 4: Run the chgiphost Command

Perform these steps using the Identity Management Oracle home:

1. Log in to the host as the user that installed Identity Management.
2. Set the ORACLE_HOME environment variable. Do not use a trailing slash (UNIX) or backslash (Windows) when specifying the ORACLE_HOME variable.

3. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
4. Run the following commands in the Identity Management Oracle home:
   - On UNIX systems:

     ```
     cd ORACLE_HOME/chgip/scripts
     ./chgiphost.sh -idm
     ```

   - On Windows systems:

     ```
     cd ORACLE_HOME\chgip\scripts
     cmd /c chgiphost.bat -idm
     ```

   The `chgiphost` command prompts for information, as shown in Table 8–4. Note that the prompts may provide values in parentheses. You can enter a different value, or press the return key to accept the suggested value.

Table 8–4 Prompts and Actions for chgiphost -idm

Prompt	Action
`Enter fully qualified hostname (hostname.domainname) of destination`	If you changed the hostname or domain name on your system, enter the new fully-qualified hostname. Otherwise, enter the current fully-qualified hostname.
`Enter fully qualified hostname (hostname.domainname) of source`	If you changed the hostname or domain name on your system, enter the old fully-qualified hostname. Otherwise, enter the current fully-qualified hostname.
`Enter valid IP Address of destination`	If you changed the IP address of the system, enter the new IP address. Otherwise, enter the current IP address
`Enter valid IP Address of source`	If you changed the IP address of the system, enter the old IP address. Otherwise, enter the current IP address

5. Verify that the tool ran successfully by checking for errors in the files in the following directory:

   ```
   (UNIX) ORACLE_HOME/chgip/log
   (Windows) ORACLE_HOME\chgip\log
   ```

Task 5: Restart Your Environment

Restart the Identity Management installation and any other instances that you stopped during this procedure:

1. Restart the Identity Management instance, using the following commands:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ORACLE_HOME\opmn\bin\opmnctl startall
     ORACLE_HOME\bin\emctl start iasconsole
     ```

2. If you disabled any processes for automatically starting Oracle Application Server at the beginning of this procedure, enable them.

Task 6: Update Your Environment

This task contains the steps to update your environment for the new hostname, domain name, or IP address. The steps you need to take depend on how your environment is configured. If you changed the hostname or IP address of the host containing:

- **Oracle Internet Directory only:** See "Configuration 1: Oracle Internet Directory Only". Oracle Internet Directory is installed on one host and the other Identity Management components are installed on another host and you change the host that contains Oracle Internet Directory. In this case, you must update the other Identity Management components and the middle tiers that use this Identity Management.
- **Identity Management components other than Oracle Internet Directory:** See "Configuration 2: OracleAS Single Sign-On, Oracle Delegated Administration Services, and (optionally) Oracle Directory Integration and Provisioning". Oracle Internet Directory is installed on one host and the other Identity Management components are installed on another host and you change the host that contains the other Identity Management components. In this case, you must update the middle tiers that use this Identity Management.
- **Oracle Internet Directory and other Identity Management components:** See "Configuration 3: Oracle Internet Directory, OracleAS Single Sign-On, Oracle Delegated Administration Services, and (optionally) Oracle Directory Integration and Provisioning". Oracle Internet Directory and the other Identity Management components are installed on the same host. In this case, you must update the middle tiers that use this Identity Management.

If your environment uses LDAP-Based replication of Oracle Internet Directory and Oracle Internet Directory is on a different host than OracleAS Metadata Repository, you can change the hostname, domain name or IP address of the host containing the Master (supplier) or Replica (consumer) Oracle Internet Directory. See Task 7: Update Oracle Internet Directory If LDAP-Based Replication Is Used on page 8-16 for information.

Configuration 1: Oracle Internet Directory Only In this case, Oracle Internet Directory is installed on one host and the other Identity Management components are installed on another host and you changed the host that contains Oracle Internet Directory. Take the following steps:

1. In the OracleAS Single Sign-On installation, stop the Infrastructure processes and the Application Server Control Console:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/bin/emctl stop iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ORACLE_HOME\bin\emctl stop iasconsole
     ```

2. Update the `ias.properties` file in every instance that uses Oracle Internet Directory. This includes other Identity Management instances (OracleAS Single Sign-On, Oracle Delegated Administration Services, and Oracle Directory

Integration and Provisioning) and middle-tier instances (J2EE and Web Cache, Portal and Wireless, and Business Intelligence and Forms).

In each Oracle home, update the following file:

```
(UNIX) ORACLE_HOME/config/ias.properties
(Windows) ORACLE_HOME\config\ias.properties
```

In the file, update the `OIDhost` parameter in with the new hostname:

```
OIDhost=newhost.us.oracle.com
```

3. Update the `ldap.ora` file in every instance that uses Oracle Internet Directory. This includes other Identity Management instances (OracleAS Single Sign-On, Oracle Delegated Administration Services, and Oracle Directory Integration and Provisioning) and middle-tier instances (J2EE and Web Cache, Portal and Wireless, and Business Intelligence and Forms).

   In each Oracle home, edit the following file:

   ```
   (UNIX) ORACLE_HOME/ldap/admin/ldap.ora
   (Windows) ORACLE_HOME\ldap\admin\ldap.ora
   ```

   In the file, update the `DIRECTORY_SERVERS` parameter with the new fully-qualified hostname.

4. In the Oracle homes for the other Identity Management components and the middle-tier instances, restart OPMN and Application Server Control Console:

   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl start
     ORACLE_HOME\bin\emctl start iasconsole
     ```

5. In the Oracle homes for the other Identity Management components and each middle tier, run the Change Identity Management Services wizard and supply the new Oracle Internet Directory information:

   a. Using the Application Server Control Console, navigate to the Application Server Home page for OracleAS Single Sign-On.

   b. Click the **Infrastructure** link.

   c. On the Infrastructure page, in the Identity Management section, click **Change.**

   d. Follow the steps in the wizard for supplying the new Identity Management information (new hostname).

      Note that although you may see the new Internet Directory host and port on the page, you still need to perform this step. The Application Server Control Console displays the virtual hostname only because it read it from the updated `ias.properties` file.

6. When the wizard completes, it asks you to restart the affected components. Run the following commands in each Oracle home:

   – On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ```

– On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl stopall
ORACLE_HOME\opmn\bin\opmnctl startall
```

7. If OracleAS Certificate Authority is installed, take the following steps:

   a. Stop OracleAS Certificate Authority, the OC4J `oca` process, and the Oracle HTTP Server on the host running OracleAS Certificate Authority. For example, on UNIX, execute the following commands:

   ```
   ORACLE_HOME/oca/bin/ocactl stop
   ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=oca
   ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
   ```

   b. Edit the following file and change the name of the host listed in the file:

   ```
   (UNIX) ORACLE_HOME/oca/conf/oca.conf
   (Windows) ORACLE_HOME\oca\conf\oca.conf
   ```

   c. Reassociate with OracleAS Single Sign-On and Oracle Internet Directory. For example, on UNIX:

   ```
   ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port OcaSslPort
   ```

   d. Start Oracle HTTP Server, the OC4J `oca` process, and OracleAS Certificate Authority. For example, on UNIX:

   ```
   ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server
   ORACLE_HOME/opmn/bin/opmnctl startproc process-type=oca
   ORACLE_HOME/oca/bin/ocactl start
   ```

Configuration 2: OracleAS Single Sign-On, Oracle Delegated Administration Services, and (optionally) Oracle Directory Integration and Provisioning In this case, Oracle Internet Directory is installed on one host and the other Identity Management components are installed on another host and you changed the host that contains the other Identity Management components.

In each middle-tier installation (J2EE and Web Cache, Portal and Wireless, or Business Intelligence and Forms), take the following steps:

1. Start the OPMN and the Application Server Control Console:

   - On UNIX systems:

   ```
   ORACLE_HOME/opmn/bin/opmnctl start
   ORACLE_HOME/bin/emctl start iasconsole
   ```

   - On Windows systems:

   ```
   ORACLE_HOME\opmn\bin\opmnctl start
   ORACLE_HOME\bin\emctl start iasconsole
   ```

2. In the Oracle home for each middle tier, run the Change Identity Management Services wizard and supply the new Oracle Internet Directory information:

   a. Using the Application Server Control Console, navigate to the Application Server Home page for OracleAS Single Sign-On.

   b. Click the **Infrastructure** link.

   c. On the Infrastructure page, in the Identity Management section, click **Change.**

d. Follow the steps in the wizard for supplying the new Identity Management information (new hostname).

Note that although you may see the new Internet Directory host and port on the page, you still need to perform this step. The Application Server Control Console displays the virtual hostname only because it read it from the updated `ias.properties` file.

3. Restart the affected components. Run the following commands in each Oracle home:

– On UNIX systems:

```
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
```

– On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl stopall
ORACLE_HOME\opmn\bin\opmnctl startall
```

Configuration 3: Oracle Internet Directory, OracleAS Single Sign-On, Oracle Delegated Administration Services, and (optionally) Oracle Directory Integration and Provisioning In this case, Oracle Internet Directory and the other Identity Management components are installed on the same host and this is the host you changed. Take the following steps:

1. Start the OPMN and the Application Server Control Console:

- On UNIX systems:

```
ORACLE_HOME/opmn/bin/opmnctl start
ORACLE_HOME/bin/emctl start iasconsole
```

- On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl start
ORACLE_HOME\bin\emctl start iasconsole
```

2. Update the `ias.properties` file in every middle-tier instance.

In each Oracle home, update the following file:

```
(UNIX) ORACLE_HOME/config/ias.properties
(Windows) ORACLE_HOME\config\ias.propertie
```

In the file, update the `OIDhost` parameter in with the new hostname:

```
OIDhost=newhost.us.oracle.com
```

3. Update the `ldap.ora` file in every middle-tier instance that uses the Identity Management instance.

In each Oracle home, edit the following file:

```
(UNIX) ORACLE_HOME/ldap/admin/ldap.ora
(Windows) ORACLE_HOME\ldap\admin\ldap.ora
```

In the file, update the `DIRECTORY_SERVERS` parameter with the new fully-qualified hostname.

4. In each middle-tier installation (J2EE and Web Cache, Portal and Wireless, or Business Intelligence and Forms), run the Change Identity Management Services wizard:
   a. Using the Application Server Control Console, navigate to the Application Server Home page for the middle-tier instance.
   b. Click the **Infrastructure** link.
   c. On the Infrastructure page, in the Identity Management section, click **Change.**

      Note that the Infrastructure page may display an error, but the error will be resolved after you complete the steps in the wizard.
   d. Follow the steps in the wizard for supplying the new Identity Management information.
5. Restart the affected components. Run the following commands in each Oracle home:
   – On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ```

   – On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ORACLE_HOME\opmn\bin\opmnctl startall
     ```

Task 7: Update Oracle Internet Directory If LDAP-Based Replication Is Used

If your environment uses LDAP-Based replication of Oracle Internet Directory and Oracle Internet Directory is on a different host than OracleAS Metadata Repository, you can change the hostname, domain name or IP address of the host containing the Master (supplier) or Replica (consumer) Oracle Internet Directory:

- Configuration A: Host with Master Oracle Internet Directory is Changed
- Configuration B: Host with Replica Oracle Internet Directory is Changed

Configuration A: Host with Master Oracle Internet Directory is Changed

If you change the hostname, domain name, or IP address of the host containing the Master Oracle Internet Directory, take the following steps:

1. Obtain the replica ID of the Master Oracle Internet Directory:

   ```
   ldapsearch -p master_port -h master_host -b "" -s base "objectclass=*"
   orclreplicaid
   ```

2. On *both* the Master and the Replica, update either `orclreplicauri` or `orclreplicasecondaryuri` or both, if they exist, in the replica entry of the Master Oracle Internet Directory. Take the following steps:
   a. Create a file named `mod.ldif` and enter the following lines in the file:

      ```
      dn: orclreplicaid=master_replicaID,cn=replication configuration
      changetype:modify
      replace: orclreplicauri
      orclreplicauri: ldap://new_master_host:new_master_port/
      ```

      In the example, *`master_replicaID`* is the ID obtained in Step a, *`new_master_host`* is the new hostname of the Master Oracle Internet

Directory, and *new_master_port* is the port number for the Master Oracle Internet Directory.

b. Run the following command on the Master:

```
ldapmodify -p master_port -h master_host -f mod.ldif
```

c. Run the following command on the Replica:

```
ldapmodify -p replica_port -h replica_host -f mod.ldif
```

3. Restart the Replication server at the Replica:

```
oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
 replica_host -p replica_port -m false" stop
oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
 replica_host -p replica_port -m false" start
```

In the example, *replica_host* is the hostname of the Replica Oracle Internet Directory and *replica_port* is the port of the Replica Oracle Internet Directory.

Configuration B: Host with Replica Oracle Internet Directory is Changed

If you change the hostname, domain name, or IP address of the host containing the Replica Oracle Internet Directory, take the following steps:

1. Obtain the replica ID of the Replica Oracle Internet Directory:

```
ldapsearch -p replica_port -h replica_host -b "" -s base "objectclass=*"
orclreplicaid
```

2. On *both* the Master and the Replica, update either `orclreplicauri` or `orclreplicasecondaryuri` or both, if they exist, in the replica entry of the Replica Oracle Internet Directory. Take the following steps:

a. Create a file named `mod.ldif` and enter the following lines in the file:

```
dn: orclreplicaid=replica_replicaID,cn=replication configuration
changetype:modify
replace: orclreplicauri
orclreplicauri: ldap://new_replica_host:new_replica_port/
```

In the example, *replica_replicaID* is the ID obtained in Step a, *new_replica_host* is the new hostname of the Replica Oracle Internet Directory, and *new_replica_port* is the port number for the Replica Oracle Internet Directory.

b. Run the following command on the Master:

```
ldapmodify -p master_port -h master_host -f mod.ldif
```

c. Run the following command on the Replica:

```
ldapmodify -p replica_port -h replica_host -f mod.ldif
```

3. Restart the Replication server at the Replica:

```
oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
 new_replica_host -p new_replica_port -m false" stop
oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
 new_replica_host -p new_replica_port -m false" start
```

In the example, *new_replica_host* is the new hostname of the Replica Oracle Internet Directory and *new_replica_port* is the port of the Replica Oracle Internet Directory.

8.2.4 Changing the Hostname or Domain Name of an OracleAS Certificate Authority Installation

If you have installed OracleAS Certificate Authority, and you want to change the name of the OracleAS Certificate Authority host, then you must perform these steps:

1. Verify that Oracle Internet Directory and OracleAS Metadata Repository are started.
2. Stop OracleAS Certificate Authority, the OC4J `oca` process, and the Oracle HTTP Server on the host running OracleAS Certificate Authority. For example, on UNIX, execute the following commands:

   ```
   ORACLE_HOME/oca/bin/ocactl stop
   ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=oca
   ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
   ```

3. Change the name of the host where OracleAS Certificate Authority is running.
4. Regenerate the SSL wallet. For example, on UNIX:

   ```
   ORACLE_HOME/oca/bin/ocactl generatewallet -type CASSL
   ```

5. Reassociate with OracleAS Single Sign-On and Oracle Internet Directory. For example, on UNIX:

   ```
   ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port OcaSslPort
   ```

6. Start Oracle HTTP Server, the OC4J `oca` process, and OracleAS Certificate Authority. For example, on UNIX:

   ```
   ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server
   ORACLE_HOME/opmn/bin/opmnctl startproc process-type=oca
   ORACLE_HOME/oca/bin/ocactl start
   ```

8.2.5 Changing the IP Address of an Infrastructure Containing a Metadata Repository

This section describes how to change the IP address of a host that contains either of the following Infrastructure installation types:

- Metadata Repository only
- Identity Management and Metadata Repository

The following sections describe the procedure:

- Before You Begin
- Task 1: Shut Down Middle-Tier Instances
- Task 2: Prepare Your Host
- Task 3: Change the IP Address
- Task 4: Update the Infrastructure
- Task 5: Restart Your Environment

Before You Begin

Review the following items before you start the procedure:

- Write down the old IP address before you begin. You will be prompted for this during the procedure.
- Oracle recommends that you perform a backup of your environment before you start this procedure. Refer to Part V, "Backup and Recovery".

Task 1: Shut Down Middle-Tier Instances

Shut down all middle-tier instances that use the Infrastructure installation, even if they are on other hosts.

Task 2: Prepare Your Host

Prepare your host for the change by stopping all processes:

1. Set the ORACLE_HOME and ORACLE_SID environment variables.
2. Shut down the Infrastructure:
   - On UNIX systems:

     ```
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\bin\emctl stop iasconsole
     ORACLE_HOME\opmn\bin\opmnctl stopall
     ```

   Shut down the listener and database:

   ```
   lsnrctl stop

   sqlplus /nolog
   SQL> connect SYS as SYSDBA
   SQL> shutdown
   SQL> quit
   ```

3. Verify that all Oracle Application Server processes have stopped.
4. To make sure Oracle Application Server processes will not start automatically after a restart of the host, disable any automated startup scripts you may have set up, such as `/etc/init.d` scripts.

Task 3: Change the IP Address

Update your operating system with the new IP address, restart the host, and verify that the host is functioning properly on your network. Consult your operating system documentation for information on how to perform the following steps:

1. Make the updates to your operating system to properly change the IP address.
2. Restart the host, if required by your operating system.
3. Verify that you can ping the host from another host in your network. Be sure to ping using the new IP address to make sure everything is resolving properly.

Task 4: Update the Infrastructure

Update the Infrastructure on your host with the new IP address:

1. Log in to the host as the user that installed the Infrastructure.

2. Set the ORACLE_HOME and ORACLE_SID environment variables. Do not use a trailing slash (UNIX) or backslash (Windows) when specifying the ORACLE_HOME variable.
3. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 1–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
4. Start the database and listener:

```
sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> startup
SQL> quit

lsnrctl start
```

5. Start OPMN:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl start
(Windows) ORACLE_HOME\opmn\bin\opmnctl start
```

6. Start Oracle Internet Directory:

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OID
process-type=OID
(Windows) ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=OID
process-type=OID
```

7. Run the following commands in the Infrastructure Oracle home:
   - On UNIX systems:

     ```
     cd ORACLE_HOME/chgip/scripts
     ./chgiphost.sh -infra
     ```

   - On Windows systems:

     ```
     cd ORACLE_HOME\chgip\scripts
     cmd /c chgiphost.bat -infra
     ```

   The `chgiphost` command prompts for the old and new IP address.
8. Verify that the tool ran successfully by checking for errors in the files in the following directory:

```
(UNIX) ORACLE_HOME/chgip/log
(Windows) ORACLE_HOME\chgip\log
```

Task 5: Restart Your Environment

Start the remaining components of the Infrastructure and start any middle-tier instances that use it:

1. Start the Infrastructure:
   - On UNIX systems:

     ```
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - On Windows systems:

     ```
     ORACLE_HOME\opmn\bin\opmnctl startall
     ```

```
ORACLE_HOME\bin\emctl start iasconsole
```

2. If a middle-tier instance is on the same host as the Infrastructure, then you need to run the `chgiphost` command on the middle-tier instance before restarting the middle-tier processes.
3. If you disabled any processes for automatically starting Oracle Application Server at the beginning of this procedure, enable them.

8.2.6 Special Topics for Changing Your Hostname or Domain Name

This section contains the following special topics that apply to changing the hostname or domain name of an Oracle Application Server host:

- Running SSLConfigTool for SSL Environments
- Setting the Log Level for chgiphost
- Customizing the chgiphost Command
- Changing a Hostname after Upgrading from Windows 2000 to Windows 2003
- Recovering from Errors When Changing Your Hostname

8.2.6.1 Running SSLConfigTool for SSL Environments

After running the `chgiphost` command, you must run the `SSLConfigTool` utility to complete the necessary Oracle Directory Integration and Provisioning server registration and OracleAS Single Sign-On re-association and re-registration.

> **See Also:** Chapter 14 for further information about running the `SSLConfigTool` utility

8.2.6.2 Setting the Log Level for chgiphost

By default, the console log level for the `chgiphost` command is `SEVERE`. This causes only critical information to be printed while running `chgiphost`. To view additional progress information, set the console log level to `CONFIG` as follows:

1. Edit the following file:

   ```
   (UNIX) ORACLE_HOME/chgip/config/chgip.log.properties
   (Windows) ORACLE_HOME\chgip\config\chgip.log.properties
   ```

2. Change the `java.util.logging.ConsoleHandler.level` parameter to `CONFIG`:

   ```
   java.util.logging.ConsoleHandler.level = CONFIG
   ```

8.2.6.3 Customizing the chgiphost Command

By default, the `chgiphost` command updates key configuration files in the Oracle home with the new hostname. If any of the following cases apply to your installation, you may want to consider customizing the behavior of the `chgiphost` command:

- You have created additional configuration files that contain the hostname and want the `chgiphost` command to update those files.

  To update these files, add their full path name to the following file before running `chgiphost`:

  ```
  (UNIX) ORACLE_HOME/chgip/config/hostname.lst
  (Windows) ORACLE_HOME\chgip\config\hostname.lst
  ```

- Your old hostname is very short (one or two letters) or is a string that is likely to appear in a configuration file.

  Before running `chgiphost`, examine each of the files listed in `hostname.lst` to determine if the old hostname exists in any settings in those files. If you find a match, you can correct those settings after you run `chgiphost`.

- Your Oracle home contains the hostname in its full path.

  In this case, the `chgiphost` command may not update your configuration files properly. You can avoid this problem by using a Java utility called FileFixer, which searches for specific text strings in a file by matching regular expressions, and updates them to their new values. Note that FileFixer searches for patterns one line at a time. It cannot match patterns across lines.

  To use FileFixer:

  1. Make a copy of the following file:

     ```
     (UNIX) ORACLE_HOME/chgip/config/hostname_short_sample.lst.xml
     (Windows) ORACLE_HOME\chgip\config\hostname_short_sample.lst.xml
     ```

  2. Edit your copy of the file to specify the regular expression matching required for your old and new hostnames. The file contains an example of how to do this.

  3. Specify the file when running the `chgiphost` command:

     ```
     ./chgiphost option -hostnameShortXml full_path_to_your_xml_file
     ```

     For example, if you named your file `/mydir/my_sample.lst.xml`, and you are updating a middle-tier installation on UNIX, run `chgiphost` as follows:

     ```
     ./chgiphost -mid -hostnameShortXml /mydir/my_sample.lst.xml
     ```

8.2.6.4 Changing a Hostname after Upgrading from Windows 2000 to Windows 2003

When you upgrade from Windows 2000 to Windows 2003, lowercase letters in your hostname may be changed to uppercase letters. For example, if your hostname is `myhost` before the upgrade, it may be changed to `MYHOST`. If this occurs, some Oracle Application Server processes may not function properly.

To resolve this problem, you do not need to run the `chgiphost` command to update Oracle Application Server. You can simply add an entry with the lowercase hostname to the hosts file:

```
OS_path\system32\drivers\etc\hosts
```

For example, if your fully-qualified hostname was `myhost.mydomain` before the upgrade, and your IP address is 1.2.3.4, add the following line:

```
1.2.3.4 myhost.mydomain myhost
```

8.2.6.5 Recovering from Errors When Changing Your Hostname

This section describes how to recover from typical errors you might encounter when using the `chgiphost` command. It contains the following scenarios:

- Scenario 1: You Specified the Wrong Destination Name
- Scenario 2: You Encountered an Error when Running chgiphost

Scenario 1: You Specified the Wrong Destination Name

Suppose you ran the `chgiphost` command but specified the wrong destination name. In this case, you can remedy the error by running `chgiphost` again. Here are the details.

Suppose the current source hostname is loire985, the incorrect destination hostname you specified is mqa985, and the correct destination hostname is sqb985. Initially, you ran `chgiphost` with source = loire985 and destination = mqa985.

To recover from this error:

1. Run `chgiphost` with source = mqa985 and destination = sqb985.
2. Run `chgiphost` again with source = loire985 and destination = sqb985.

Scenario 2: You Encountered an Error when Running chgiphost

For example, you will get an error message if you enter the wrong password for Oracle Internet Directory. In this case, you should run `chgiphost` again, with the same source and destination hostnames as before, and make sure to supply the correct password when prompted.

If you encounter an error when running `chgiphost`, you should fix the error and run `chgiphost` again.

8.3 Moving Between Off-Network and On-Network

This section describes how to move an Oracle Application Server host on and off the network. The following assumptions and restrictions apply:

- The host must contain an Infrastructure and middle-tier instance, or a J2EE and Web Cache instance that does not use an Infrastructure, that is, the entire Oracle Application Server environment must be on the host.
- DHCP must be used in loopback mode. Refer to Oracle Application Server Installation Guide for more information.
- Only IP address change is supported; the hostname must remain unchanged.
- Hosts in DHCP mode should not use the default hostname (`localhost.localdomain`). The hosts should be configured to use a standard hostname and the loopback IP should resolve to that hostname.
- A loopback adapter is required for all off-network installations (DHCP or static IP). Refer to Oracle Application Server Installation Guide for more information.

8.3.1 Moving from Off-Network to On-Network (Static IP Address)

This procedure assumes you have installed Oracle Application Server on a host that is off the network, using a standard hostname (not `localhost`), and would like to move on the network and use a static IP address. The IP address may be the default loopback IP, or any standard IP address.

To move onto the network, you can simply plug the host into the network. No updates to Oracle Application Server are required.

8.3.2 Moving from Off-Network to On-Network (DHCP)

This procedure assumes you have installed on a host that is off the network, using a standard hostname (not `localhost`), and would like to move on the network and use

DHCP. The IP address of the host can be any static IP address or loopback IP address, and should be configured to the hostname.

To move onto the network, connect the host to the network using DHCP and configure the hostname to the loopback IP address only.

8.3.3 Moving from On-Network to Off-Network (Static IP Address)

Follow this procedure if your host is on the network, using a static IP address, and you would like to move it off the network:

1. Configure the `/etc/hosts` file so the IP address and hostname can be resolved locally.
2. Take the host off the network.
3. There is no need to perform any steps to change the hostname or IP address.

8.3.4 Moving from On-Network to Off-Network (DHCP)

Follow this procedure if your host is on the network, using DHCP in loopback mode, and you would like to move it off the network:

1. Configure the /etc/hosts file so the IP address and hostname can be resolved locally.
2. Take the host off the network.
3. There is no need to perform any steps to change the hostname or IP address.

8.4 Changing Between a Static IP Address and DHCP

This section describes how to change between a static IP address and DHCP. The following assumptions and restrictions apply:

- The host must contain an Infrastructure and middle-tier instance, or a J2EE and Web Cache instance that does not use an Infrastructure, that is, the entire Oracle Application Server environment must be on the host.
- DHCP must be used in loopback mode. Refer to *Oracle Application Server Installation Guide* for more information.
- Only IP address change is supported; the hostname must remain unchanged.
- Hosts in DHCP mode should not use the default hostname (`localhost.localdomain`). The hosts should be configured to use a standard hostname and the loopback IP should resolve to that hostname.

8.4.1 Changing from a Static IP Address to DHCP

To change a host from a static IP address to DHCP:

1. Configure the host to have a hostname associated with the loopback IP address before you convert the host to DHCP.
2. Convert the host to DHCP. There is no need to update Oracle Application Server.

8.4.2 Changing from DHCP to a Static IP Address

To change a host from DHCP to a static IP address:

1. Configure the host to use a static IP address.

2. There is no need to update Oracle Application Server.

9

Changing Infrastructure Services

This chapter provides procedures for changing the Infrastructure Services used by a middle-tier instance.

It contains the following topics:

9.1 Overview of Procedures for Changing Infrastructure Services

Most middle-tier instances use Infrastructure Services, such as Identity Management Services and the Metadata Repository. These services are usually assigned during installation.

After installation, you may want to change the Infrastructure Services used by a middle-tier instance. For example, you may want to use an Identity Management Service on a different host. Or, you may want to use a different Metadata Repository.

You can change Infrastructure Services using the Infrastructure page on the Application Server Control Console, shown in Figure 9–1. Notice that you can change the Identity Management or the Metadata Repository used by a middle-tier instance.

Figure 9–1 Application Server Control Console Infrastructure Page

You must change Infrastructure Services when you change any of the following:

- The HTTP OracleAS Single Sign-On port number on an Identity Management installation
- The Oracle Internet Directory non-SSL or SSL port number
- The Oracle Internet Directory Mode (Dual-mode or SSL)
- The host on which Identity Management or the OracleAS Metadata Repository resides

If you have disabled anonymous binds in Oracle Internet Directory, you must enable them before you make configuration changes. See Section 7.6, "Disabling and Enabling Anonymous Binds" for more information.

Note that if you change between a File-Based farm and a Database-based farm, you must restart Application Server Control Console, using the `emctl` command, to see the change reflected in the console.

You cannot simply use the wizard to change from one Infrastructure service to another. You must first perform manual tasks to create and prepare the new Infrastructure service. This chapter describes the following supported procedures for changing Infrastructure Services:

- Changing the Oracle Internet Directory or Oracle HTTP Server Ports on Identity Management

  Use this procedure if you need to change the Oracle Internet Directory listener port or the HTTP listener port for Oracle Internet Directory on an Identity Management installation. In addition to changing the port numbers, you must update middle-tier instances with the new port information, which requires changing Infrastructure Services.

- Changing Oracle Internet Directory from Dual Mode to SSL Mode

  Use this procedure if you want to change the Oracle Internet Directory mode from non-SSL to SSL. In addition to changing the mode, you must update middle-tier instances with the new mode, which requires changing Infrastructure Services.

- Moving Identity Management to a New Host

  Use this procedure if you want to move your Identity Management installation, and its associated Metadata Repository, to a new host. After you perform the move, you must update middle-tier instances with the new host information for Identity Management, which requires changing Infrastructure Services.

- Changing the Metadata Repository Used by a Middle-Tier Instance

  Use this procedure if you want to move the Metadata Repository used for product metadata by middle-tier instances to a new host.

- Changing the Metadata Repository Used by Identity Management

  Use this procedure if you have an Identity Management installation using a Metadata Repository and you want to move the Metadata Repository to a different host.

9.2 Changing the Oracle Internet Directory or Oracle HTTP Server Ports on Identity Management

To change the Oracle Internet Directory non-SSL or SSL port on an Identity Management installation, refer to Section 4.4.2, "Changing Oracle Internet Directory Ports".

To change the Oracle HTTP Server non-SSL or SSL Listen port on an Identity Management installation, which effectively changes the OracleAS Single Sign-On port, refer to Section 4.4.3, "Changing the HTTP Server Port on an Identity Management Installation".

9.3 Changing Oracle Internet Directory from Dual Mode to SSL Mode

When you install Identity Management, you are asked to choose a mode for Oracle Internet Directory. The default mode is dual mode, which allows some components to access Oracle Internet Directory using non-SSL connections. During the installation, you can choose SSL mode, which specifies that all components must use SSL when connecting to the directory.

If you did not choose SSL mode during the installation, and want to change to SSL mode after installation, follow the procedure in this section. It includes changing the mode of the Oracle Internet Directory, and updating middle-tier instances to use the new mode.

9.3.1 Procedure

To change Oracle Internet Directory to SSL mode, perform the following tasks:

- Task 1: Stop Middle-Tier Instances
- Task 2: Change the Oracle Internet Directory Mode
- Task 3: Change Middle-Tier Instances to Use SSL Mode

Task 1: Stop Middle-Tier Instances

Stop all middle-tier instances that use Oracle Internet Directory. Using the Application Server Control Console, navigate to the Home page for each middle-tier instance and click **Stop All.** Be sure to leave Application Server Control running.

Task 2: Change the Oracle Internet Directory Mode

Perform this task on the Infrastructure that contains Oracle Internet Directory.

1. Create a file named `mod.ldif` and enter the following lines in the file:

```
dn:cn=configset0,cn=osdldapd,cn=subconfigsubentry
changetype:modify
replace:orclsslenable
orclsslenable:1
```

2. Run the following command:

```
ldapmodify -D cn=orcladmin -w orcladmin_passwd -p oid_port -v -f mod.ldif
```

   In the example, *oid_port* is the non-SSL Oracle Internet Directory port. This is listed as `OIDport` in *ORACLE_HOME*`/config/ias.properties`.

   Note that if you are using OracleAS Cold Failover Cluster, you must use the following command:

```
ldapmodify -D cn=orcladmin -w orcladmin_passwd -h virtual_hostname
-p oid_port -v -f mod.ldif
```

   In the example, *virtual_hostname* is the virtual hostname of the OracleAS Cold Failover Cluster.

3. Stop the entire instance that contains Oracle Internet Directory:

   - On UNIX systems:

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
```

   - On Windows systems:

```
ORACLE_HOME\bin\emctl stop iasconsole
ORACLE_HOME\opmn\bin\opmnctl stopall
```

4. Edit the following file:

```
(UNIX) ORACLE_HOME/ldap/admin/ldap.ora
(Windows) ORACLE_HOME\ldap\admin\ldap.ora
```

   a. Modify the following line to remove the non-SSL port number:

```
DIRECTORY_SERVERS=(myhost.myco.com::sslport)
```

   b. Save and close the file.

5. If the OracleAS Metadata Repository was created using OracleAS Metadata Repository Creation Assistant, take the following steps:

   a. Copy the `ldap.ora` file from the Identity Management Oracle home to the Oracle home for the OracleAS Metadata Repository. For example, for 10*g* Release 2 (10.1.2), the location is:

```
(UNIX) Oracle_Home/ldap/admin
(Windows) Oracle_Home\ldap\admin
```

   b. Edit the `sqlnet.ora` file that is located in the following location in the Oracle home for the OracleAS Metadata Repository:

```
(UNIX) Oracle_Home/network/admin
(Windows) Oracle_Home\network\admin
```

Add LDAP to the NAMES.DIRECTORY_PATH entry, as shown in the following example:

```
NAMES.DIRECTORY_PATH= (LDAP, TNSNAMES, ONAMES, HOSTNAME)
```

6. Edit the following file:

```
(UNIX) ORACLE_HOME/config/ias.properties
(Windows) ORACLE_HOME\config\ias.properties
```

   a. Change the `SSLOnly` parameter as follows:

```
SSLOnly=true
```

   b. Save and close the file.

7. Start the entire instance that contains Oracle Internet Directory:

   - On UNIX systems:

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

   - On Windows systems:

```
ORACLE_HOME\opmn\bin\opmnctl startall
ORACLE_HOME\bin\emctl start iasconsole
```

Task 3: Change Middle-Tier Instances to Use SSL Mode

In each middle-tier instance, run the Change Identity Management wizard and restart the instance:

1. Using the Application Server Control Console, navigate to the Home page for the middle-tier instance.
2. Click **Infrastructure.**
3. On the Infrastructure Page, in the Identity Management section, click **Change.**
4. On the Internet Directory page:
   - **Host:** Enter the fully-qualified name of the Oracle Internet Directory host.
   - **Port:** Enter the SSL Oracle Internet Directory port number.
   - **Use only SSL connections with Internet Directory:** Check this box.

   Click **Next.**
5. On the Login page:
   - **User Name:** Enter `cn=orcladmin`, or the distinguished name of a user in the `iASAdmins` group.
   - **Password:** Enter the password for the user.

   Click **Next.**
6. On the Validation page, you will receive informational messages regarding the validation of this operation. If you receive any error message, follow the instructions for investigating them. Otherwise, if the operation is valid, click **Finish.**
7. When the operation is finished, start the components in the middle-tier instance:
   a. Click **Home** to navigate to the Home page for the instance.

b. Click **Start All.**

Note: Now that you have disabled the non-SSL Oracle Internet Directory port, you must provide the "-U 1" option when using LDAP command-line utilities (such as `ldapsearch`, `ldapmodify`, and `ldapaddmt`) to connect to the SSL port.

9.4 Moving Identity Management to a New Host

This section provides a procedure for moving Identity Management to a new host. This procedure involves creating a replica (or copy) of the original Identity Management on a different host, along with its own new Metadata Repository, and then changing the middle-tier instance to use the new Identity Management.

9.4.1 Sample Uses for This Procedure

The following are sample uses for this procedure:

- You have an existing Identity Management and associated Metadata Repository that is used by one or more middle-tier instances. Your organization intends to replace the current Identity Management host with a new system. You can use this procedure to create a replica of the Identity Management, along with its own Metadata Repository, and change the middle-tier instances to use the new Identity Management. You can then retire the original host.
- You want to create a failover environment for your Identity Management. You can use this procedure to create a replica of the current Identity Management, along with its own Metadata Repository. You can keep the replica running so it stays synchronized with the original Identity Management. You can perform regular exports of data in the original Metadata Repository and save them. In the event that you lose the original Identity Management, you can import the data to the new Metadata Repository, and change the middle-tier instances to use the new Identity Management. Refer to Section 9.4.4, "Strategy for Performing Failover with This Procedure" for more information.

9.4.2 Assumptions and Restrictions

- For both the original and new installations, the Identity Management and Metadata Repository can exist in the same Oracle home, or in separate Oracle homes (same or different host). If they are in separate Oracle homes, perform the operations on each in their own Oracle home.
- For both the original and new installations, the Identity Management components (OracleAS Single Sign-On, Oracle Internet Directory, Delegated Administration Services, Directory Integration and Provisioning) may exist in the same Oracle home, or may exist in separate Oracle homes (same or different host). If they exist in separate Oracle homes, perform the operations on each in their own Oracle home.
- The Metadata Repository used by middle-tier instances for product metadata is not affected by this procedure.
  - If the middle-tier instances use product metadata in the same Metadata Repository that the original Identity Management uses, they will continue to use that Metadata Repository after you have changed them to the new Identity Management. If you want, you can change them to use a different Metadata Repository after you have finished moving Identity Management. Refer to

Section 9.5, "Changing the Metadata Repository Used by a Middle-Tier Instance".

  - If the middle-tier instances use a separate Metadata Repository for product metadata, they will continue to use that Metadata Repository after you have changed them to the new Identity Management.

- This procedure does not take OracleAS Certificate Authority into consideration.

  See Also: *Oracle Application Server Certificate Authority Administrator's Guide* for information on updating OracleAS Certificate Authority when changing Identity Management services

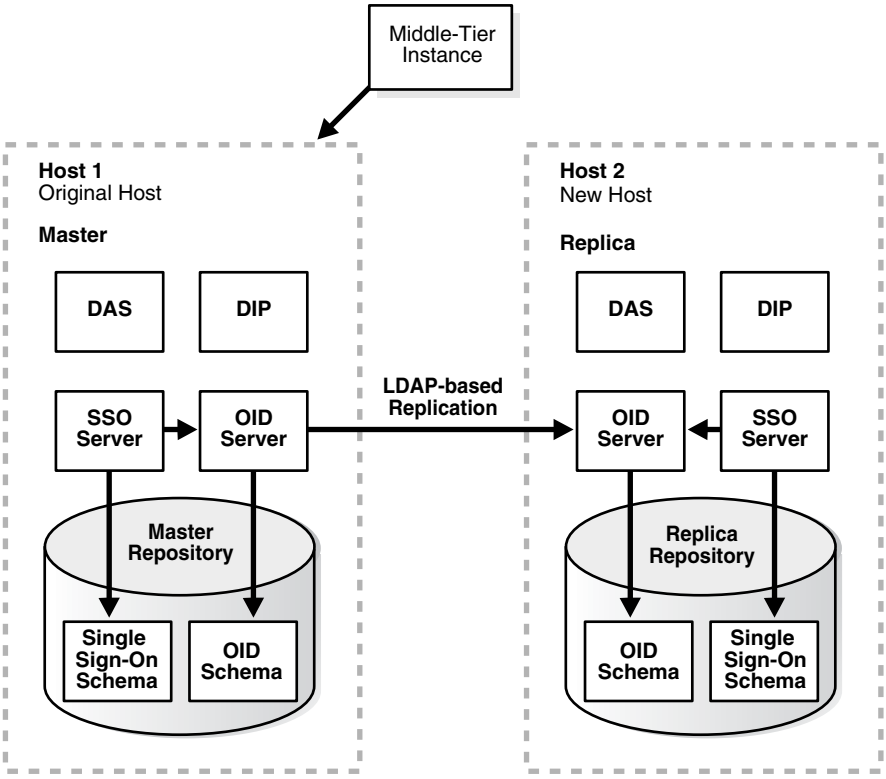
9.4.3 Procedure

An overview of the procedure is as follows:

1. You have an original Identity Management (also called the Master) used by one or more middle-tier instances. The Identity Management has a Metadata Repository. You install and set up a new Identity Management (also called the Replica). This Identity Management has its own Metadata Repository. The Oracle Internet Directory in the new Identity Management is an LDAP-based replica of the original Oracle Internet Directory. Replication takes place constantly from the original Oracle Internet Directory to the new Oracle Internet Directory.

   Figure 9–2 shows a sample of this setup.

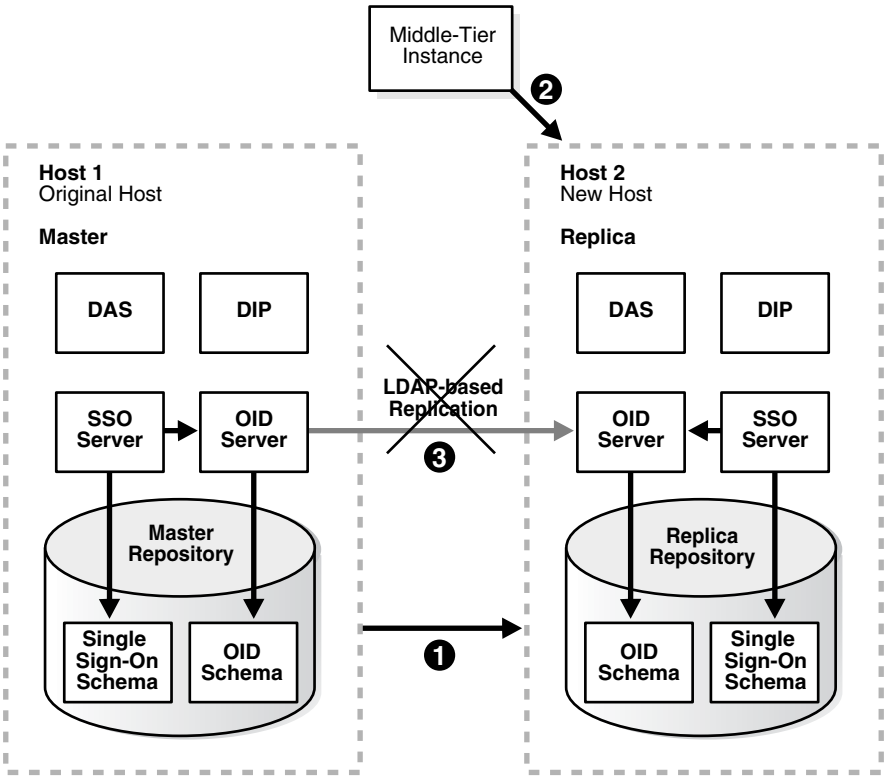
Figure 9–2 Original Host (Master) and New Host (Replica)



2. You perform the following tasks to change to the new Identity Management. The tasks are shown in Figure 9–3.

- **1**: Migrate OracleAS Single Sign-On and Directory Integration and Provisioning data from the original Metadata Repository (Master) to the new Metadata Repository (Replica).
- **2**: Change the middle-tier instances to use the new Metadata Repository.
- **3**: Stop the LDAP-based replication.

Figure 9–3 Changing from Original to New Identity Management



To change to the new Oracle Internet Directory, perform the following tasks:

- Task 1: Install and Set Up the New Identity Management and Metadata Repository
- Task 2: Migrate OracleAS Single Sign-On and Directory Integration and Provisioning Data
- Task 3: Change Middle-Tier Instances to the New Identity Management
- Task 4: Stop Replication

Task 1: Install and Set Up the New Identity Management and Metadata Repository

In this task, you install and set up the new Identity Management and its associated Metadata Repository. The new Identity Management is an LDAP-based replica of the original Identity Management.

1. Read Section H.1, "About LDAP-Based Replicas" to learn about LDAP-based replicas and how they are used for this procedure.
2. Follow the procedure in Section H.2, "Installing and Setting Up an LDAP-Based Replica" to install and set up the new Identity Management and Metadata Repository.

Task 2: Migrate OracleAS Single Sign-On and Directory Integration and Provisioning Data

In this task, you migrate the OracleAS Single Sign-On and Directory Integration and Provisioning Data from the original Metadata Repository to the new Metadata Repository. The source for the migration is the original Metadata Repository (Master) and the target for the migration is the new Metadata Repository (Replica).

This task contains the following subtasks:

- Migrate the OracleAS Single Sign-On Data
- Migrate the Directory Integration and Provisioning Data

> **Note:** Make sure the ORACLE_HOME and ORACLE_SID environment variables are set before you begin. This applies to all platforms.

Migrate the OracleAS Single Sign-On Data

To migrate the OracleAS Single Sign-On data:

1. Obtain the ORASSO schema password on the master:

```
MASTER_HOME/bin/ldapsearch -p master_oid_port -h
master_host -D "cn=orcladmin"
 -w master_orcladmin_passwd -b
"orclresourcename=orasso,  orclreferencename=master_global_db_name,
cn=ias infrastructure databases,
cn=ias, cn=products, cn=oraclecontext" -s base "objectclass=*"
orclpasswordattribute
```

   This command prints the ORASSO password in a line like the following:

```
orclpasswordattribute=LAetjdQ5
```

2. Export the OracleAS Single Sign-On data from the master, ensuring that the ORACLE_HOME environment variable is set before you run this command:

```
MASTER_HOME/sso/bin/ssomig -export -s orasso -p
 master_orasso_passwd -c master_db_name -log_d $MASTER_HOME/sso/log
```

   In the example, *master_orasso_passwd* is the ORASSO password obtained in the previous step.

3. Copy the `ssomig.dmp` and `ssoconf.log` files from the master to the replica, preserving the exact full path for each file:

```
UNIX:
cp MASTER_HOME/sso/log/ssomig.dmp REPLICA_HOME/sso/log/ssomig.dmp
cp MASTER_HOME/sso/log/ssoconf.log REPLICA_HOME/sso/log/ssoconf.log

Windows:
copy MASTER_HOME\sso\log\ssomig.dmp REPLICA_HOME\sso\log\ssomig.dmp
copy MASTER_HOME\sso\log\ssoconf.log REPLICA_HOME\sso\log\ssoconf.log
```

4. Obtain the ORASSO schema password on the replica:

```
REPLICA_HOME/bin/ldapsearch -p replica_oid_port -h replica_host -D
"cn=orcladmin" -w replica_orcladmin_password -b "orclresourcename=orasso,
orclreferencename=replica_global_db_name, cn=ias infrastructure databases,
cn=ias, cn=products, cn=oraclecontext" -s base "objectclass=*"
```

```
orclpasswordattribute
```

5. Import the OracleAS Single Sign-On data to the replica:

```
REPLICA_HOME/sso/bin/ssomig -import -overwrite -s
orasso -p replica_orasso_passwd -c replica_db_name -log_d
$REPLICA_HOME/sso/log -discoforce
```

In the example, *replica_orasso_passwd* is the ORASSO password obtained in the previous step.

6. Verify that the export and import of OracleAS Single Sign-On succeeded.

Verify that the OracleAS Single Sign-On migration tool reported success. You can also check the following log files for errors:

```
MASTER_HOME/sso/log/ssomig.log
REPLICA_HOME/sso/log/ssomig.log
```

> **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide* for information on interpreting messages in the log files

Migrate the Directory Integration and Provisioning Data

To migrate your Directory Integration and Provisioning Data:

> **See Also:** Directory Integration and Provisioning Data documentation in the *Oracle Internet Directory Administrator's Guide* for information about running the following commands using the HTTPS port in environments in which the Oracle Internet Directory HTTP port is disabled

1. Stop the Directory Integration and Provisioning Data server on the master:

```
MASTER_HOME/bin/oidctl server=odisrv instance=1 stop
```

2. Migrate the Directory Integration and Provisioning Data:

```
MASTER_HOME/bin/dipassistant reassociate -src_ldap_host
master_host -src_ldap_port
master_oid_port -dst_ldap_host
replica_host -dst_ldap_port replica_oid_port -src_ldap_passwd
master_orcladmin_passwd -dst_ldap_passwd replica_orcladmin_passwd
```

This command prints log messages to:

```
MASTER_HOME/ldap/odi/log/reassociate.log
```

3. Stop the Directory Integration and Provisioning Data server on the replica:

```
REPLICA_HOME/bin/oidctl server=odisrv instance=1 stop
```

4. Register the Directory Integration and Provisioning Data server on the replica:

```
REPLICA_HOME/bin/odisrvreg -D "cn=orcladmin" -w
replica_orcladmin_passwd -h replica_host -p replica_oid_port
```

5. Start the Directory Integration and Provisioning Data server on the replica:

```
REPLICA_HOME/bin/oidctl server=odisrv instance=1 flags="port=replica_oid_port"
start
```

Task 3: Change Middle-Tier Instances to the New Identity Management

In each middle-tier instance, run the Change Identity Management wizard and restart the instance:

1. Using the Application Server Control Console, navigate to the Home page for the middle-tier instance.
2. Click **Infrastructure.**
3. On the Infrastructure Page, in the Identity Management section, click **Change.**
4. Follow the steps in the wizard for supplying the new Identity Management information.
5. When the wizard is finished, navigate to the Home page for the instance and start your instance by clicking **Start All.**

If you have a problem changing the middle-tier instances to the new host, check to make sure replication is running and try again.

Task 4: Stop Replication

Stop the replication between the original Identity Management and the new Identity Management (Replica) by running the following command in the new Identity Management Oracle home:

```
oidctl connect=global_db_name server=oidrepld instance=1 flags="-p oid_port" stop
```

In the example, *global_db_name* is the global database name of the new Identity Management. (This is referred to as *replica_db_name* in Section H.2, "Installing and Setting Up an LDAP-Based Replica".)

In the example, *oid_port* is the non-SSL Oracle Internet Directory port in the new Identity Management.

9.4.4 Strategy for Performing Failover with This Procedure

As mentioned in Section 9.4.1, you can modify this procedure to perform failover for Identity Management. This enables you to move your middle-tier instances to the new Identity Management in case the original is lost.

To perform failover:

1. Install and set up the new Identity Management as described in "Task 1: Install and Set Up the New Identity Management and Metadata Repository".
2. Export Oracle Application Server Single Sign-On and Directory Integration and Provisioning Data on a regular basis from the original Metadata Repository. You do not need to import the data into the new Metadata Repository. You only need to export the data and copy the files to the new Metadata Repository Host. Refer to "Task 2: Migrate OracleAS Single Sign-On and Directory Integration and Provisioning Data".
3. If you lose the original Identity Management:
   a. Stop replication. Refer to "Task 4: Stop Replication".
   b. Import your most recent copy of the Oracle Application Server Single Sign-On and Directory Integration and Provisioning Data into the new Identity Management repository. Refer to "Task 2: Migrate OracleAS Single Sign-On and Directory Integration and Provisioning Data".

c. Change the middle-tier instances to use the new Identity Management. Refer to "Task 3: Change Middle-Tier Instances to the New Identity Management".

9.5 Changing the Metadata Repository Used by a Middle-Tier Instance

This section provides a procedure for changing the Metadata Repository used by a middle-tier instance. This procedure involves making a copy of the original Metadata Repository on a different host, and then changing the middle-tier instance to use the new Metadata Repository.

9.5.1 Sample Uses for This Procedure

The following are sample uses for this procedure:

- You have an existing Metadata Repository that is used by one or more middle-tier instances. Your organization intends to replace the current Metadata Repository host with a new system. You can use this procedure to copy the Metadata Repository to the new host and change your middle-tier instances to use the new Metadata Repository. You can then retire the original host.
- You want to move a Metadata Repository from a host in your test environment, to a host in your Production Environment. You can use this procedure to copy the Metadata Repository from the test-to-production host, and change your test middle-tier instances to use the new Metadata Repository.

9.5.2 Assumptions and Restrictions

In this scenario:

- The middle-tier instances use Identity Management.
- The Identity Management installation does not use the original Metadata Repository for its Identity Management schemas; it uses a separate Metadata Repository.
- The original Metadata Repository:
  - Is used for product metadata and DCM management only (it is not used by Identity Management)
  - Must be registered with Oracle Internet Directory
- The new Metadata Repository:
  - Must not be registered with Oracle Internet Directory initially. During the procedure, you will register it with the same Oracle Internet Directory as the original Metadata Repository.
  - Must be created with the same Oracle home, datafile location, SID, and global database name as the original Metadata Repository. You will eventually change the global database name to a unique name.
- OracleAS Certificate Authority is not supported by this procedure and must not be configured in your environment.
- If the Metadata Repository is used for OracleAS Clusters, the cluster members will not be accessible until all members of the cluster have been changed to the new Metadata Repository.

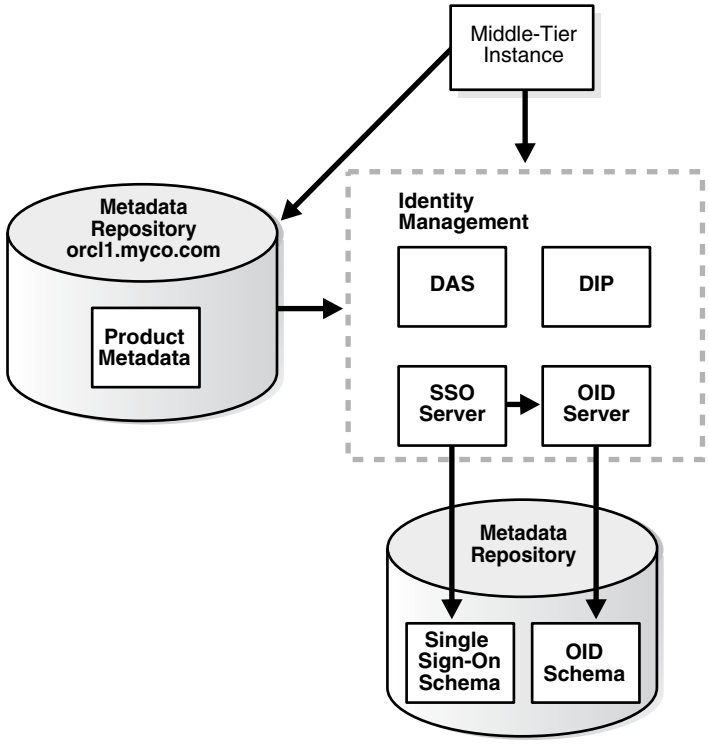
9.5.3 Overview

An overview of the procedure is as follows:

1. You have an original Metadata Repository. It is used by one or more middle-tier instances for product metadata. The middle-tier instances use Identity Management, and the Metadata Repository is registered with Oracle Internet Directory in that Identity Management.

   Figure 9–4 shows a sample original Metadata Repository (`orcl1.myco.com`).

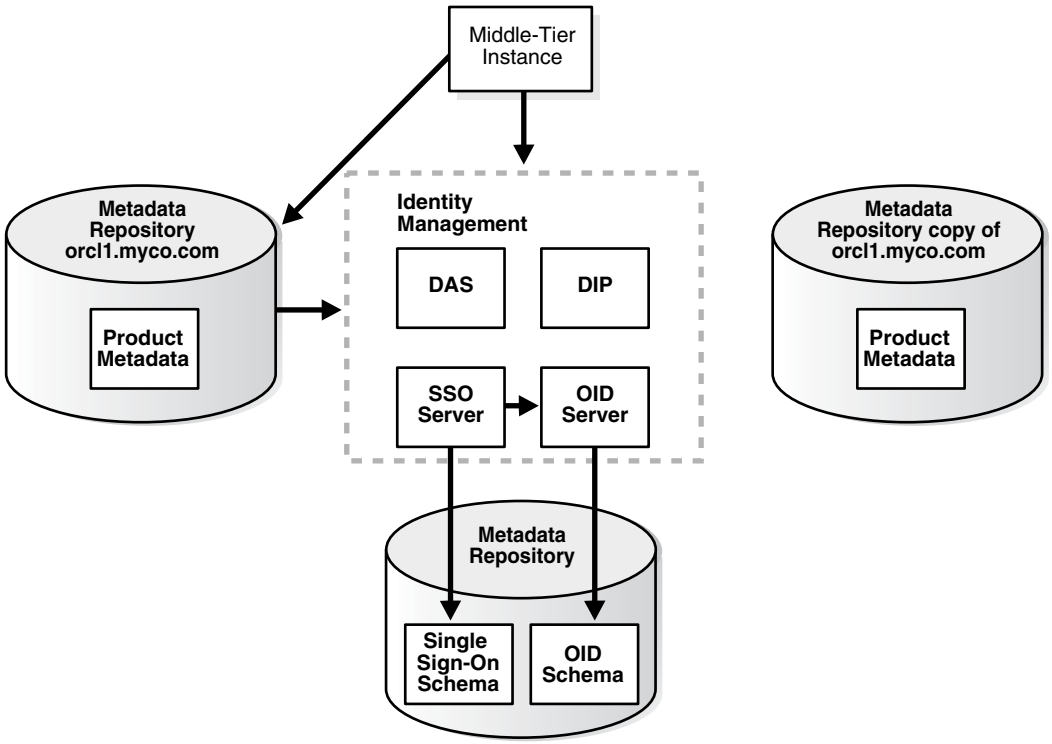
Figure 9–4 Original Metadata Repository



2. You create a copy of the original Metadata Repository by installing a new Metadata Repository, backing up the original Metadata Repository, and restoring to the new Metadata Repository.

   Figure 9–5 shows sample original and new Metadata Repositories.

Figure 9–5 Original Metadata Repository and New Metadata Repository

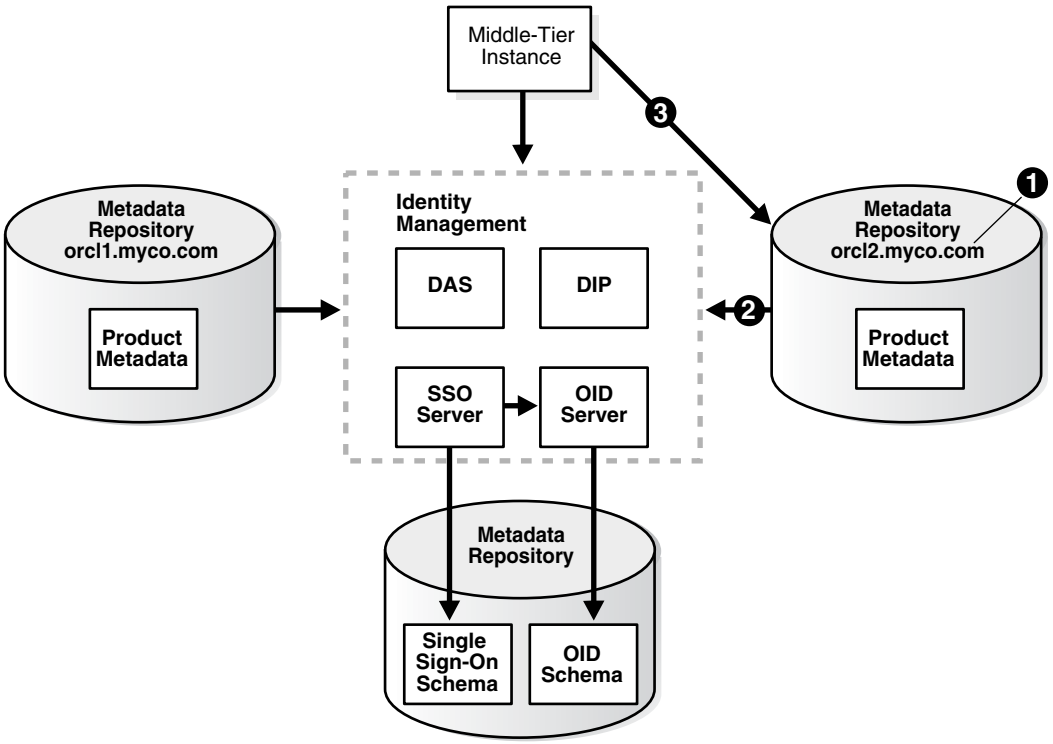


The following table shows sample attributes for the original and new Metadata Repositories after you installed the new Metadata Repository:

Attribute	Original Metadata Repository	New Metadata Repository
Oracle home	`/private/oraHome`	`/private/oraHome`
Datafile location	`/private/oraHome/oradata`	`/private/oraHome/oradata`
SID	`orcl1`	`orcl1`
Global database name	`orcl1.myco.com`	`orcl1.myco.com`
Registered with Oracle Internet Directory?	Yes	**No**

3. You perform the following tasks to change to the new Metadata Repository. The tasks are shown in Figure 9–6.
   - **1**: Change the global database name of the new Metadata Repository to a unique name (in this sample, `orcl2.myco.com`).
   - **2**: Register the new Metadata Repository with the same Oracle Internet Directory as the old Metadata Repository.
   - **3**: Change the middle-tier instances to use the new Metadata Repository.

Figure 9–6 Changing from the Original to the New Metadata Repository



The following table shows the sample attributes for the original Metadata Repository and the attributes for the new Metadata Repository after you perform this step.

Attribute	Original Metadata Repository	New Metadata Repository
Oracle home	`/private/oraHome`	`/private/oraHome`
Datafile location	`/private/oraHome/oradata`	`/private/oraHome/oradata`
SID	`orcl1`	`orcl1`
Global database name	`orcl1.myco.com`	**`orcl2.myco.com`**
Registered with Oracle Internet Directory?	Yes	**Yes**

4. If you are using the scenario where you no longer require the original Metadata Repository, you can discard the original Metadata Repository.

9.5.4 Procedure

To change the Metadata Repository, perform the following tasks:

- Task 1: Install the New Metadata Repository
- Task 2: Back Up the Original Metadata Repository
- Task 3: Restore the Backup to the New Metadata Repository
- Task 4: Configure Oracle Ultra Search Metadata in the New Metadata Repository
- Task 5: Change the Global Database Name for the New Metadata Repository
- Task 6: Register the New Metadata Repository with Oracle Internet Directory

- Task 7: Change Middle-Tier Instances to the New Metadata Repository
- Task 8: Update the Farm Name

Before You Begin

If your middle-tier instances use OracleAS Portal and Oracle Ultra Search, you will need to supply the WKSYS schema password later in this procedure. You should obtain this password now from the old Metadata Repository.

> **Note:** For information on how to obtain the WKSYS password, see Section 6.3, "Viewing OracleAS Metadata Repository Schema Passwords".

Task 1: Install the New Metadata Repository

Install the new Metadata Repository as follows:

1. Make sure you install the Metadata Repository into an Oracle home that has the same path as the old Metadata Repository Oracle home.
2. Use Oracle Universal Installer to install the Metadata Repository.
3. Choose to install an Infrastructure.
4. Choose to install a Metadata Repository only.
5. Do not register the Metadata Repository with Oracle Internet Directory.
6. Specify the same SID and global database name as the old Metadata Repository.
7. Specify the same datafile location as the old Metadata Repository.

Task 2: Back Up the Original Metadata Repository

In this task, you create a backup of the original Metadata Repository. This section provides the steps for creating the backup using Oracle Recovery Manager; however, if you are an experienced DBA, you can back up the Metadata Repository according to your standard practices.

Perform all of the steps in this task on the original Metadata Repository host:

1. Create directories to store backup files and log files. For example:

   ```
   mkdir -p BACKUP_DIR/log_files
   mkdir -p BACKUP_DIR/db_files
   ```

2. Make sure the original Metadata Repository is started.
3. Make sure you have set the ORACLE_HOME and ORACLE_SID environment variables.
4. Connect to the database as a user with SYSDBA privileges.
5. Obtain the DBID of the original Metadata Repository using SQL*Plus:

   ```
   SQL> SELECT DBID FROM v$database;
   ```

   Make note of this value; you will use it later in the procedure.

6. Create a file named *BACKUP_DIR*`/cold_backup.rcv`. Enter the following lines in the file:

   ```
   shutdown immediate;
   startup mount;
   ```

```
configure controlfile autobackup on;
configure controlfile autobackup format for device type disk to 'BACKUP_DIR/db_
files/%F';

run {
allocate channel dev1 device type disk format
'BACKUP_DIR/db_files/%U';
backup database plus archivelog;
release channel dev1;
}
```

In the file, substitute the full path for *BACKUP_DIR*.

7. Run Oracle Recovery Manager to back up the Metadata Repository.

   - You can run Oracle Recovery Manager on the Metadata Repository host as follows (the following is a single command; type it all on one line):

     ```
     ORACLE_HOME/bin/rman target /
     cmdfile=BACKUP_DIR/cold_backup.rcv > BACKUP_DIR/log_files/backup.log
     ```

     Note that the preceding command contains a forward slash "/" character.

   - You can run Oracle Recovery Manager from another host on the network as follows (the following is a single command; type it all on one line):

     ```
     ORACLE_HOME/bin/rman target SYS/oracle@trgt
     cmdfile=BACKUP_DIR/cold_backup.rcv > BACKUP_DIR/log_files/backup.log
     ```

8. Copy the backup directories to the new host.

Note: If you copy the files to a different location on the new host, you must use the CATALOG command to update the RMAN repository with the new filenames and use the CHANGE ... UNCATALOG command to uncatalog the old filenames.

See *Oracle Database Backup and Recovery Advanced User's Guide* in the Oracle Database documentation for more information about the CATALOG command.

Task 3: Restore the Backup to the New Metadata Repository

In this task, you restore the backup to the new Metadata Repository.

Perform all of the steps in this task on the new Metadata Repository host.

1. Make sure the new Metadata Repository is shut down:

   ```
   sqlplus "SYS/sys_password as SYSDBA"
   SQL> SHUTDOWN IMMEDIATE;
   ```

2. Regenerate the password file:

   - On UNIX:

     ```
     mv ORACLE_HOME/dbs/orapwORACLE_SID ORACLE_HOME/dbs/orapwORACLE_SID.old

     ORACLE_HOME/bin/orapwd file=ORACLE_HOME/dbs/orapwORACLE_SID password=new_
     password
     ```

   - On Windows:

     ```
     move ORACLE_HOME\database\PWDORACLE_SID.ora
     ```

```
ORACLE_HOME\database\PWDORACLE_SID.ora.old

ORACLE_HOME\bin\orapwd file=ORACLE_HOME\database\PWDORACLE_SID.ora
password=new_password
```

In the examples, *new_password* is the new SYS password. You can use the old SYS password, or set it to a new password.

3. Start the new Metadata Repository but do not mount it:

```
SQL> STARTUP NOMOUNT;
```

4. Create a file named *BACKUP_DIR*/restore.rcv that contains the following lines. In the file, substitute the full path for *BACKUP_DIR* and the *DBID* value you obtained in the previous task for *dbid*.

```
set dbid=DBID;
connect target /;
set controlfile autobackup format for device type disk to 'BACKUP_DIR/db_
files/%F';
restore controlfile from autobackup;
startup mount force;

run {
allocate channel dev1 device type disk format
'BACKUP_DIR/db_files/%U';
restore database;
release channel dev1;
alter database open resetlogs;
}
```

5. Run Oracle Recovery Manager to restore the Metadata Repository.
   - If you are logged into the host machine for the Metadata Repository, run the following command:

     ```
     ORACLE_HOME/bin/rman cmdfile=BACKUP_DIR/restore.rcv >
     BACKUP_DIR/log_files/restore.log
     ```

   - If you are accessing the host machine for the Metadata Repository from another machine on the network, edit the following file:

     ```
     BACKUP_DIR/restore.rcv
     ```

     Make sure the file contains the following line:

     ```
     TARGET SYS/oracle@trgt
     ```

     Run the following command:

     ```
     ORACLE_HOME/bin/rman cmdfile=BACKUP_DIR/restore.rcv >
     BACKUP_DIR/log_files/restore.log
     ```

6. After you restore using Oracle Recovery Manager, determine if the TEMP tablespace has a datafile by connecting to the database as a user with SYSDBA privileges and running the following command in SQL*Plus:

```
SQL> select file_name from dba_temp_files where tablespace_name like 'TEMP';
```

If the preceding command does not return any files, take the following steps:

a. Check if the following file exists on your system:

```
ORADATA_DIRECTORY/db_name/temp01.dbf
```

b. If the file exists, enter the following command:

```
SQL> alter tablespace "TEMP" add tempfile 'ORADATA_DIRECTORY/
db_name/temp01.dbf'
size 5120K reuse autoextend on next 8k maxsize unlimited;
```

If the file does not exist, enter the following command:

```
SQL> alter tablespace "TEMP" add tempfile 'ORADATA_DIRECTORY/
db_name/temp01.dbf' size 5120K autoextend on next 8k maxsize unlimited;
```

Note that you must type the preceding commands on one line, and *db_name* is the first portion of the new global database name.

Task 4: Configure Oracle Ultra Search Metadata in the New Metadata Repository

Perform this task on the new Metadata Repository:

1. Make sure the ORACLE_HOME and ORACLE_SID environment variables are set.
2. Run the following commands:

```
(UNIX)    cd ORACLE_HOME/ultrasearch/admin
(Windows) cd ORACLE_HOME\ultrasearch\admin

sqlplus "SYS/sys_password as SYSDBA"
SQL> @wk0config.sql WksysPw JDBC_ConnStr LAUNCH_ANYWHERE ""
```

(Note the two double quotes at the end of the preceding command.)

In the example:

- *WksysPw* is the password of the WKSYS schema that you obtained at the beginning of this procedure.
- *JDBC_ConnStr* is the JDBC connection string *host*:*port*:*SID*, for example: `myhost:1521:testdb`.
- *LAUNCH_ANYWHERE* is `TRUE` if the Metadata Repository is in Real Application Clusters mode, otherwise `FALSE`. For this procedure, set it to `FALSE`.

Task 5: Change the Global Database Name for the New Metadata Repository

In this task, you change the global database name of the new Metadata Repository to a new, unique name so you can register it with Oracle Internet Directory.

Perform all of the steps in this task on the new Metadata Repository host:

1. Run the following commands to set up the database:

```
sqlplus "SYS/sys_password as SYSDBA"
SQL> alter system switch logfile;
SQL> alter database backup controlfile to trace resetlogs;
```

2. Check the spfile using SQL*Plus:

```
SQL> select value from v$parameter where name='spfile';
```

3. If the previous command returns no rows, skip this step.

   Check if the previous command returns output like the following:

```
VALUE
---------------------------------
?/dbs/spfile@.ora
```

If it does, run the following command to create a pfile from the spfile:

- On Unix:

```
SQL> create pfile='ORACLE_HOME/dbs/initORACLE_SID.ora' from spfile;
```

- On Windows:

```
SQL> create pfile='%ORACLE_HOME%\database\initORACLE_SID.ora' from spfile;
```

In the example, *ORACLE_SID* is the SID of the original and new Metadata Repository.

4. Shut down the new Metadata Repository:

```
SQL> SHUTDOWN IMMEDIATE;
```

The database must be shut down with SHUTDOWN NORMAL or SHUTDOWN IMMEDIATE. Do not use SHUTDOWN ABORT.

5. Rename the spfile so the pfile will be used when the database instance is restarted:

- On Unix:

```
cd ORACLE_HOME/dbs
mv spfileORACLE_SID.ora spfileORACLE_SID.ora.save
```

- On Windows:

```
cd ORACLE_HOME\database
rename spfileORACLE_SID.ora spfileORACLE_SID.ora.save
```

6. Edit the following file:

```
(UNIX) ORACLE_HOME/dbs/initORACLE_SID.ora
(Windows) ORACLE_HOME\database\initORACLE_SID.ora
```

Update the db_name to the new db_name (the first portion of the new global database name). For example, if the new global database name is `orcl1.myco.com`, the value of db_name should be `orcl1`. Note that this is not necessarily (nor likely) the same value as the SID on the new Metadata Repository.

Also, update all other instances of the old db_name to the new db_name. Specifically, you should update directory paths that contain the old db_name. If the directory paths are not updated, when you run the `ccf.sql` script in step 16, the script will fail.

7. Rename the following directory with the new *db_name*:

```
(UNIX)  ORADATA_DIRECTORY/db_name
(Windows) ORADATA_DIRECTORY\db_name
```

8. Rename the control files so they do not exist later when the new ones are created:

```
(UNIX)   cd ORADATA_DIRECTORY/db_name
(Windows) cd ORADATA_DIRECTORY\db_name

mv control01.ctl control01.ctl.old
mv control02.ctl control02.ctl.old
mv control03.ctl control03.ctl.old
```

9. Rename the following directory with the new *db_name*:

```
(UNIX)   ORACLE_HOME/admin/db_name
```

```
(Windows) ORACLE_HOME\..\admin\db_name
```

Note that on Windows, the `admin` directory is in the same directory as the Oracle home.

10. Edit the following file:

```
(UNIX)    ORACLE_HOME/admin/db_name/pfile/init.ora.NNNNNNNNNNNN
(Windows) ORACLE_HOME\..\admin\db_name\pfile\init.ora.NNNNNNNNNNNN
```

Note that the filename includes a random number at the end.

Change all instances of the old db_name to the new db_name; do not update the SID. To do this, change the old db_name in all directory paths and the `db_name` parameter.

11. Change to the trace file directory:

```
(UNIX)    cd ORACLE_HOME/admin/db_name/udump
(Windows) cd ORACLE_HOME\..\admin\db_name\udump
```

Note that the preceding is the default location for the trace file directory. This location can be overridden by the `user_dump_dest` parameter in `init`*`ORACLE_SID`*`.ora` or `spfile`*`ORACLE_SID`*`.ora`.

12. Locate the trace file; it has a name of the form *`ORACLE_SID`*`_ora_`*`NNNNN`*`.trc`, where *`NNNNN`* is a number. Choose the trace file with the most recent modification date.

13. Copy the contents of the trace file, starting from the line with `STARTUP NOMOUNT` to the end of the file, into a new file named *`BACKUP_DIR`*`/ccf.sql`.

Do not copy any the following lines, if they exist:

```
*** TIMESTAMP kcrr.c
ARCH: Archival disabled due to shutdown: 1089
```

14. Edit *`BACKUP_DIR`*`/ccf.sql` as follows. (Example 9–1 shows an example of the `ccf.sql` file after performing the edits in this step.)

    a. Update the following line with the new global database name and change `REUSE` to `SET`:

    Before modification:

    ```
    CREATE CONTROLFILE REUSE DATABASE "Old_Global_DB_Name" RESETLOGS ...
    ```

    After modification:

    ```
    CREATE CONTROLFILE SET DATABASE "New_Global_DB_Name" RESETLOGS ...
    ```

    b. Remove the line that appears in one of the following two forms:

    ```
    # STANDBY LOGFILE
    -- STANDBY LOGFILE
    ```

    c. Comment out the following lines, if they exist, with "REM", as shown:

    ```
    REM RECOVER DATABASE USING BACKUP CONTROLFILE

    REM VARIABLE RECNO NUMBER;

    REM EXECUTE :RECNO := SYS.DBMS_BACKUP_RESTORE.SETCONFIG('CONTROLFILE
    AUTOBACKUP','ON');
    ```

```
REM VARIABLE RECNO NUMBER;

REM EXECUTE :RECNO := SYS.DBMS_BACKUP_
RESTORE.SETCONFIG('CONTROLFILEAUTOBACKUP FORMAT FOR DEVICE TYPE','DISK TO
 BACKUP_DIR/db_files/%F');

REM ALTER TABLESPACE TEMP ADD TEMPFILE
'ORACLE_HOME/TEMP01.DBF' SIZE 5242880 AUTOEXTEND ON MAXSIZE 4294950912
REUSE;
```

d. Change all comment symbols to `REM`. Depending on your platform, the comment symbol may be # or `--`.

Example 9–1 Example ccf.sql File after Edits

```
STARTUP NOMOUNT
CREATE CONTROLFILE set DATABASE "NEW DATABASE" RESETLOGS ARCHIVELOG
   MAXLOGFILES 50
   MAXLOGMEMBERS 5
   MAXDATAFILES 100
   MAXINSTANCES 1
   MAXLOGHISTORY 226
LOGFILE
 GROUP 1 '/private1/inst/oradata/orcl/redo01.log'  SIZE 50M,
 GROUP 2 '/private1/inst/oradata/orcl/redo02.log'  SIZE 50M,
 GROUP 3 '/private1/inst/oradata/orcl/redo03.log'  SIZE 50M
DATAFILE
 '/private1/inst/oradata/orcl/system01.dbf',
 '/private1/inst/oradata/orcl/undotbs01.dbf',
 '/private1/inst/oradata/orcl/drsys01.dbf',
 '/private1/inst/oradata/orcl/dcm.dbf',
 '/private1/inst/oradata/orcl/portal.dbf',
 '/private1/inst/oradata/orcl/ptldoc.dbf',
 '/private1/inst/oradata/orcl/ptlidx.dbf',
 '/private1/inst/oradata/orcl/ptllog.dbf',
 '/private1/inst/oradata/orcl/oca.dbf',
 '/private1/inst/oradata/orcl/discopltc1.dbf',
 '/private1/inst/oradata/orcl/discopltm1.dbf',
 '/private1/inst/oradata/orcl/oss_sys01.dbf',
 '/private1/inst/oradata/orcl/wcrsys01.dbf',
 '/private1/inst/oradata/orcl/uddisys01.dbf',
 '/private1/inst/oradata/orcl/ip_dt.dbf',
 '/private1/inst/oradata/orcl/ip_rt.dbf',
 '/private1/inst/oradata/orcl/ip_idx.dbf',
 '/private1/inst/oradata/orcl/ip_lob.dbf',
 '/private1/inst/oradata/orcl/attrs1_oid.dbf',
 '/private1/inst/oradata/orcl/battrs1_oid.dbf',
 '/private1/inst/oradata/orcl/gcats1_oid.dbf',
 '/private1/inst/oradata/orcl/gdefault1_oid.dbf',
 '/private1/inst/oradata/orcl/svrmg1_oid.dbf',
 '/private1/inst/oradata/orcl/ias_meta01.dbf'
CHARACTER SET WE8MSWIN1252
;
REM Configure RMAN configuration record 1
REM VARIABLE RECNO NUMBER;
REM EXECUTE :RECNO := SYS.DBMS_BACKUP_RESTORE.SETCONFIG('CONTROLFILE
AUTOBACKUP','ON');
REM Configure RMAN configuration record 2
REM VARIABLE RECNO NUMBER;
REM EXECUTE :RECNO := SYS.DBMS_BACKUP_RESTORE.SETCONFIG('CONTROLFILE AUTOBACKUP
```

```
FORMAT FOR DEVICE TYPE',’DISK TO /private1/inst/backup_dir/db_files/%F');
REM Recovery is required if any of the datafiles are restored backups,
REM or if the last shutdown was not normal or immediate.
REM RECOVER DATABASE USING BACKUP CONTROLFILE
REM Database can now be opened zeroing the online logs.
ALTER DATABASE OPEN RESETLOGS;
REM No tempfile entries found to add.
```

15. Edit the following script:

    BACKUP_DIR/ccf.sql

    Replace the old global database name with the new global database name throughout the script.

16. Run the `ccf.sql` script:

    ```
    SQL> @BACKUP_DIR/ccf.sql
    ```

17. Change the global database name in the database, specifying the *New_Global_DB_Name*:

    ```
    SQL> alter database rename global_name to New_Global_DB_Name;
    ```

18. Determine if the TEMP tablespace has a datafile by connecting to the database as a user with SYSDBA privileges and running the following command in SQL*Plus:

    ```
    SQL> select file_name from dba_temp_files where tablespace_name like 'TEMP';
    ```

    If the preceding command does not return any files, take the following steps:

    a. Check if the following file exists on your system:

       ORADATA_DIRECTORY/*db_name*/temp01.dbf

    b. If the file exists, enter the following command:

       ```
       SQL> alter tablespace "TEMP" add tempfile 'ORADATA_DIRECTORY/
       db_name/temp01.dbf'
       size 5120K reuse autoextend on next 8k maxsize unlimited;
       ```

       If the file does not exist, enter the following command:

       ```
       SQL> alter tablespace "TEMP" add tempfile 'ORADATA_DIRECTORY/
       db_name/temp01.dbf' size 5120K autoextend on next 8k maxsize unlimited;
       ```

       Note that you must type the preceding commands on one line, and *db_name* is the first portion of the new global database name.

19. Update the service name and the global database name to the new global database name in the following file:

    (UNIX) *ORACLE_HOME*/network/admin/tnsnames.ora
    (Windows) *ORACLE_HOME*\network\admin\tnsnames.ora

    Note that you should not change the SID.

20. Edit the following file:

    (UNIX) *ORACLE_HOME*/config/ias.properties
    (Windows) *ORACLE_HOME*\config\ias.properties

    Change the `InfrastructureDBCommonName` parameter to the new global database name.

Task 6: Register the New Metadata Repository with Oracle Internet Directory

In this task, you register the new Metadata Repository with the same Oracle Internet Directory used by the original Metadata Repository. To do this, you run Oracle Application Server Metadata Repository Creation Assistant (OracleAS Metadata Repository Creation Assistant), a wizard that guides you through the registration.

> **Note:** OracleAS Metadata Repository Creation Assistant is available on the "OracleAS Metadata Repository Creation Assistant" CD-ROM.

1. Using SQL*Plus, log in to the new Metadata Repository as SYS with SYSDBA privileges.
   a. Run the following SQL commands:

```
SQL> execute dbms_ias_version.set_component_loading(component_id=>'MRC',
component_name=>'Oracle Application Server Metadata Repository Version',
schema_name=>'SYS');

SQL> execute dbms_ias_version.set_component_valid(component_id=>'MRC');
```

   b. Verify that the following command returns the following value:

```
SQL> select comp_name, version, status from app_registry where comp_
id='MRC';

COMP_NAME                                              VERSION     STATUS
-------------------------------------------------------   ---------   -------
Oracle Application Server Metadata Repository Version-R  10.1.2.0.1  VALID
```

2. Register the new Metadata Repository with Oracle Internet Directory:

   See Also: *Oracle Application Server Metadata Repository Creation Assistant User's Guide* for more information on how to perform the following steps

   a. Install OracleAS Metadata Repository Creation Assistant into its own Oracle home on the host where the new Metadata Repository resides. In the Select a Product to Install screen, choose Oracle Application Server Repository Creation Assistant.

   b. Run OracleAS Metadata Repository Creation Assistant as follows:

```
(UNIX)  ORACLE_HOME_RepCA/runRepca
(Windows) ORACLE_HOME_RepCA\runRepca.bat
```

   The wizard guides you through the process.

   c. When the process is finished, an `ldap.ora` file is created in the OracleAS Metadata Repository Creation Assistant Oracle home. Copy this file to the new Metadata Repository Oracle home.

   Copy the file from:

```
(UNIX)  ORACLE_HOME_RepCA/network/admin/ldap.ora
(Windows) ORACLE_HOME_RepCA\network\admin\ldap.ora
```

   To:

```
(UNIX)  ORACLE_HOME_NEW_METADATA_REPOSITORY/network/admin/ldap.ora
(Windows) ORACLE_HOME_NEW_METADATA_REPOSITORY\network\admin\ldap.ora
```

Task 7: Change Middle-Tier Instances to the New Metadata Repository

On each middle-tier instance you want to change to the new Metadata Repository, run the Change Metadata Repository wizard and restart the instance:

1. Using the Application Server Control Console, navigate to the Home page for the middle-tier instance.
2. Make sure all components except Management are down. If not, click **Stop All** to stop them. Note that this will not stop Management.
3. Click **Infrastructure.**
4. On the Infrastructure Page, in the Metadata Repository section, click **Change.**
5. Follow the steps in the wizard for supplying the new Metadata Repository information.
6. When the wizard is finished, navigate to the Home page for the instance and start your instance by clicking **Start All.**

Task 8: Update the Farm Name

Run the following command in the Oracle home of one of the middle-tier instances that you changed to use the new Metadata Repository in the previous task:

```
(UNIX)   ORACLE_HOME/dcm/bin/dcmctl resetFarmName new_farm_name
(Windows) ORACLE_HOME\dcm\bin\dcmctl resetFarmName new_farm_name
```

In the example, *new_farm_name* is the global database name of the new Metadata Repository.

> **Note:** You only need to run the command in one middle-tier instance. The command updates all other instances.

9.6 Changing the Metadata Repository Used by Identity Management

This section provides a procedure for changing the Metadata Repository used by Identity Management. This procedure applies if the Metadata Repository is also used by a middle-tier instance that is connected to the Identity Management instance.

The procedure involves making a copy of the original Metadata Repository on a different host, and then changing the Identity Management and middle-tier components to use the new Metadata Repository.

9.6.1 Sample Uses for This Procedure

The following are sample uses for this procedure:

- You have an Identity Management installation using a Metadata Repository, and a middle-tier instance connected to the Identity Management instance may be using the same Metadata Repository. You want to move the Metadata Repository to a different host so you can retire the original Metadata Repository.
- Your current Metadata Repository host is overloaded and you want to move the Metadata Repository to a host that can handle a heavier load.

9.6.2 Assumptions and Restrictions

In this scenario:

- The Identity Management installation can reside in one Oracle home, or its components can be distributed across several Oracle homes.
- A middle-tier instance that is connected to Identity Management can use the same Metadata Repository.
- The new Metadata Repository:
  - Must reside on a different host than the original Metadata Repository. That host must use the same operating system platform as the original.
  - Must use the same Oracle home, datafile location, SID, and global database name as the original Metadata Repository.
  - Can use a different database listener port than the original.

9.6.3 Procedure

To change the Metadata Repository, perform the following tasks:

- Task 1: Install the New Metadata Repository
- Task 2: Shut Down Your Environment
- Task 3: Back Up the Original Metadata Repository
- Task 4: Restore the Backup to the New Metadata Repository
- Task 5: Configure Oracle Ultra Search Metadata in the New Metadata Repository
- Task 6: Update Oracle Internet Directory
- Task 7: Shut Down the Original Metadata Repository
- Task 8: Start Oracle Internet Directory Using Special Commands
- Task 9: Update the Oracle Internet Directory Database Registration
- Task 10: Stop Oracle Internet Directory Using Special Commands
- Task 11: Start Your Environment
- Task 12: Update OracleAS Certificate Authority

Before You Begin

If your middle-tier instances use OracleAS Portal and Oracle Ultra Search, you will need to supply the WKSYS schema password later in this procedure. You should obtain this password now from the old Metadata Repository.

> **Note:** For information on how to obtain the WKSYS password, see Section 6.3, "Viewing OracleAS Metadata Repository Schema Passwords".

Task 1: Install the New Metadata Repository

Install the new Metadata Repository on a different host, as follows:

1. Make sure you install the Metadata Repository into an Oracle home that has the same path as the old Metadata Repository Oracle home.
2. Use Oracle Universal Installer to install the Metadata Repository.
3. Choose to install an Infrastructure.
4. Choose to install a Metadata Repository only.

5. Do not register the Metadata Repository with Oracle Internet Directory.
6. Specify the same SID and global database name as the old Metadata Repository.
7. Specify the same datafile location as the old Metadata Repository.

Task 2: Shut Down Your Environment

Shut down your environment:

1. Shut down all middle-tier instances that use this instance of Identity Management.

   ```
   ORACLE_HOME/opmn/bin/opmnctl stopall
   ORACLE_HOME\opmn\bin\opmnctl stopall
   ```

2. Run the following command in the Identity Management Oracle home:

   ```
   ORACLE_HOME/opmn/bin/opmnctl stopall
   ORACLE_HOME\opmn\bin\opmnctl stopall
   ```

   If the Identity Management installation is distributed across several Oracle homes, also run the command in the Oracle Internet Directory Oracle home.

3. If you use OracleAS Certificate Authority, stop it as follows:

   ```
   (UNIX)  ORACLE_HOME/oca/bin/ocactl stop
   (Windows)  ORACLE_HOME\oca\bin\ocactl stop
   ```

Task 3: Back Up the Original Metadata Repository

In this task, you create a backup of the original Metadata Repository. This section provides the steps for doing this using Oracle Recovery Manager; however, if you are an experienced DBA, you can back up the Metadata Repository according to your standard practices.

Perform all of the steps in this task on the original Metadata Repository host:

1. Create directories to store backup files and log files. For example:

   ```
   mkdir -p BACKUP_DIR/log_files
   mkdir -p BACKUP_DIR/db_files
   ```

2. Make sure the original Metadata Repository is started.
3. Make sure you have set the ORACLE_HOME and ORACLE_SID environment variables before you run the SQL*Plus command.
4. Obtain the DBID of the original Metadata Repository using SQL*Plus:

   ```
   sqlplus "SYS/sys_password as SYSDBA"
   SQL> SELECT DBID FROM v$database;
   ```

   Make note of this value; you will use it later in the procedure.

5. Create a file named *BACKUP_DIR*`/cold_backup.rcv`. Enter the following lines in the file:

   ```
   shutdown immediate;
   startup mount;
   configure controlfile autobackup on;
   configure controlfile autobackup format for device type disk to
   'BACKUP_DIR/db_files/%F';

   run {
   allocate channel dev1 device type disk format
   'BACKUP_DIR/db_files/%U';
   ```

```
backup database plus archivelog;
release channel dev1;
}
```

In the file, substitute the full path for *BACKUP_DIR*.

6. Run Oracle Recovery Manager to back up the Metadata Repository.

   - You can run Oracle Recovery Manager on the Metadata Repository host as follows (the following is a single command; type it all on one line):

     ```
     ORACLE_HOME/bin/rman target /
     cmdfile=BACKUP_DIR/cold_backup.rcv > BACKUP_DIR/log_files/backup.log
     ```

     Note that the preceding command contains a forward slash "/" character.

   - You can run Oracle Recovery Manager from another host on the network as follows (the following is a single command; type it all on one line):

     ```
     ORACLE_HOME/bin/rman target SYS/oracle@trgt
     cmdfile=BACKUP_DIR/cold_backup.rcv > BACKUP_DIR/log_files/backup.log
     ```

7. Copy the backup directories to the new host.

Note: If you copy the files to a different location on the new host, you must use the CATALOG command to update the RMAN repository with the new filenames and use the CHANGE ... UNCATALOG command to uncatalog the old filenames.

See *Oracle Database Backup and Recovery Advanced User's Guide* in the Oracle Database documentation for more information about the CATALOG command.

Task 4: Restore the Backup to the New Metadata Repository

In this task, you restore the backup to the new Metadata Repository.

Perform all of the steps in this task on the new Metadata Repository host:

1. Make sure the new Metadata Repository is shut down:

   ```
   sqlplus "SYS/sys_password as SYSDBA"
   SQL> SHUTDOWN IMMEDIATE;
   ```

2. Regenerate the password file:

   - On UNIX:

     ```
     mv ORACLE_HOME/dbs/orapwORACLE_SID ORACLE_HOME/dbs/orapwORACLE_SID.old
     ORACLE_HOME/bin/orapwd file=ORACLE_HOME/dbs/orapwORACLE_SID password=new_
     password
     ```

   - On Windows:

     ```
     mv ORACLE_HOME\database\PWDORACLE_SID.ora ORACLE_HOME\database\PWDORACLE_
     SID.ora.old
     ORACLE_HOME\bin\orapwd file=ORACLE_HOME\database\PWDORACLE_SID.ora
     password=new_password
     ```

   In the example, *new_password* is the new SYS password. You can use the old SYS password, or set it to a new password.

3. Start the new Metadata Repository, but do not mount it:

```
SQL> STARTUP NOMOUNT;
```

4. Create a file named *BACKUP_DIR*/restore.rcv that contains the following lines. In the file, substitute the full path for *BACKUP_DIR* and the *DBID* obtained in the previous task.

```
set dbid=DBID;
connect target /;
set controlfile autobackup format for device type disk to
 'BACKUP_DIR/db_files/%F';
restore controlfile from autobackup;
startup mount force;

run {
allocate channel dev1 device type disk format
'BACKUP_DIR/db_files/%U';
restore database;
release channel dev1;
alter database open resetlogs;
}
```

5. Run Oracle Recovery Manager to restore the Metadata Repository:

```
ORACLE_HOME/bin/rman cmdfile=BACKUP_DIR/restore.rcv >
BACKUP_DIR/log_files/restore.log
```

6. After you restore using Oracle Recovery Manager, determine if the TEMP tablespace has a datafile by connecting to the database as a user with SYSDBA privileges and running the following command in SQL*Plus:

```
SQL> select file_name from dba_temp_files where tablespace_name like 'TEMP';
```

   If the preceding command does not return any files, add a datafile:

```
SQL> alter tablespace "TEMP" add tempfile 'ORADATA_DIRECTORY/ \
db_name/temp01.dbf' size 5120K autoextend on next 8k maxsize unlimited;
```

   In the example, *db_name* is the first portion of the new global database name.

   Note that the preceding command creates a file called temp01.dbf and adds it to the TEMP tablespace. If the temp01.dbf file already exists in the directory, add a REUSE clause to the command:

```
SQL> alter tablespace "TEMP" add tempfile 'ORADATA_DIRECTORY/ \
db_name/temp01.dbf' size 5120K reuse autoextend on next 8k maxsize unlimited;
```

Task 5: Configure Oracle Ultra Search Metadata in the New Metadata Repository

If a middle-tier instance uses this Metadata Repository, perform this task on the new Metadata Repository:

1. Make sure the ORACLE_HOME and ORACLE_SID environment variables are set.
2. Run the following commands:

```
(UNIX)  cd ORACLE_HOME/ultrasearch/admin
(Windows) cd ORACLE_HOME\ultrasearch\admin

sqlplus "SYS/sys_password as SYSDBA"
SQL> @wk0config.sql WksysPw JDBC_ConnStr LAUNCH_ANYWHERE ""
```

   (Note the two double quotes at the end of the preceding command.)

   In the example:

- *WksysPw* is the password of the WKSYS schema that you obtained at the beginning of this procedure.
- *JDBC_ConnStr* is the JDBC connection string *host*:*port*:*SID*, for example: `myhost:1521:testdb`.
- *LAUNCH_ANYWHERE* is `TRUE` if the Metadata Repository is in Real Application Clusters mode, otherwise `FALSE`. For this procedure, set it to `FALSE`.

Task 6: Update Oracle Internet Directory

In the Oracle Internet Directory home, update the following file with the new Metadata Repository hostname (and, optionally, the new port number):

```
(UNIX) ORACLE_HOME/network/admin/tnsnames.ora
(Windows) ORACLE_HOME\network\admin\tnsnames.ora
```

After you modify the file, use the tnsping command to make sure you can reach the new Metadata Repository:

```
(UNIX)  ORACLE_HOME/bin/tnsping net_service_name
(Windows) ORACLE_HOME\bin\tnsping net_service_name
```

For example, on UNIX:

```
ORACLE_HOME/bin/tnsping orcl
```

Task 7: Shut Down the Original Metadata Repository

Shut down the original Metadata Repository using your usual procedure.

Task 8: Start Oracle Internet Directory Using Special Commands

Start Oracle Internet Directory by taking the following steps (do not use `opmnctl`):

1. Change directory to the Oracle Internet Directory home.
2. Set the ORACLE_SID environment variable to the new Metadata Repository SID (the default is `orcl`).
3. Start the Oracle Internet Directory monitor:

   ```
   (UNIX) ORACLE_HOME/bin/oidmon connect=connectstring start
   (Windows) ORACLE_HOME\bin\oidmon connect=connectstring start
   ```

   For example, on Windows:

   ```
   OraHome1_AS\bin\oidmon connect=orcl start
   ```

Task 9: Update the Oracle Internet Directory Database Registration

Update Oracle Internet Directory with the new Metadata Repository hostname and, optionally, new port number:

1. Start Oracle Directory Manager:
   - On UNIX, use the following command:

     ```
     ORACLE_HOME/bin/oidadmin
     ```

   - On Windows, navigate to Oracle Directory Manager (**Start** > **Programs** > ***Oracle Application Server Oracle_Home*** > **Integrated Management Tools** > **Oracle Directory Manager.**)
2. Log in to Oracle Directory Manager.

3. In the System Objects frame:
   a. Expand **Entry Management.**
   b. Expand **cn=Oracle Context.**
   c. Select the DBName for the OracleAS Metadata Repository. For example, if the DBName is the default, `orcl`, select **cn=ORCL.**
4. On the Properties tab, update the `HOST` parameter in the `orclnetdescstring` field with the new hostname. Update the `PORT` parameter if you have changed the port number.
5. Click **Apply.**

Task 10: Stop Oracle Internet Directory Using Special Commands

Stop Oracle Internet Directory by taking the following steps (do not use `opmnctl`):

1. Change directory to the Oracle Internet Directory Oracle home.
2. Set the ORACLE_SID environment variable to the new Metadata Repository SID (the default is `orcl`).
3. Stop the Oracle Internet Directory monitor:

```
(UNIX) ORACLE_HOME/bin/oidmon connect=connectstring stop
(Windows) ORACLE_HOME\bin\oidmon connect=connectstring stop
```

   For example, on Windows:

```
OraHome1_AS\bin\oidmon connect=orcl stop
```

Task 11: Start Your Environment

Start your environment as follows:

1. Start the Identity Management installation by running the following command in the Identity Management Oracle home:

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME\opmn\bin\opmnctl startall
```

   If the Identity Management installation is distributed across several Oracle homes, run the command in the Oracle Internet Directory home.

2. Start all middle-tier instances that use Identity Management by running the following command in the middle tier Oracle home:

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME\opmn\bin\opmnctl startall
```

Task 12: Update OracleAS Certificate Authority

If you use OracleAS Certificate Authority, update it as follows:

1. Associate it with the new Metadata Repository:

```
(UNIX)  ORACLE_HOME/oca/bin/ocactl updateconnection
(Windows) ORACLE_HOME\oca\bin\ocactl updateconnection
```

2. Start OracleAS Certificate Authority:

```
(UNIX) ORACLE_HOME/oca/bin/ocactl start
(Windows) ORACLE_HOME\oca\bin\ocactl start
```

10

Cloning Application Server Middle-Tier Instances

This chapter provides information on cloning an installation of Oracle Application Server middle-tier instance.

If you need to copy an entire production environment that includes a middle-tier instance, Identity Management with a Metadata Repository, and a product Metadata Repository for OracleAS Portal and OracleBI Discoverer metadata, perform the procedures in Chapter 11 instead.

It contains the following topics:

10.1 Introduction to Cloning

Cloning is the process of copying an existing installation to a different location while preserving its configuration. Some situations in which cloning an installation of Oracle Application Server is useful are:

- Creating an installation that is a copy of a production, test, or development installation. Cloning enables you to create a new installation with all patches applied to it in a single step. This is in contrast to separately installing, configuring and applying any patches to Oracle Application Server.
- Rapidly deploying an instance and the applications it hosts.
- Preparing a "gold" image of a patched home and deploying it to many hosts.

The cloned installation behaves the same as the source installation. For example, the cloned instance can be deinstalled or patched using Oracle Universal Installer. It can also be used as the source for another cloning operation.

You can create a cloned copy of a test, development, or production installation by using the command-line cloning scripts.

You can quickly create a cloned copy of a test, development, or production installation by using either the command-line cloning scripts or Oracle Enterprise Manager 10*g* Grid Control Console.

The default cloning procedure is adequate for most usage cases. However, you can also customize various aspects of the cloning process, for example, to specify custom port assignments, or preserve custom settings.

Figure 10–1 shows cloning a J2EE and Web Cache middle tier that is not connected to OracleAS Infrastructure components.

Figure 10–1 Cloning a J2EE and Web Cache Middle Tier

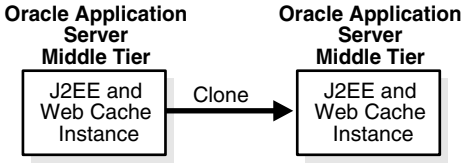
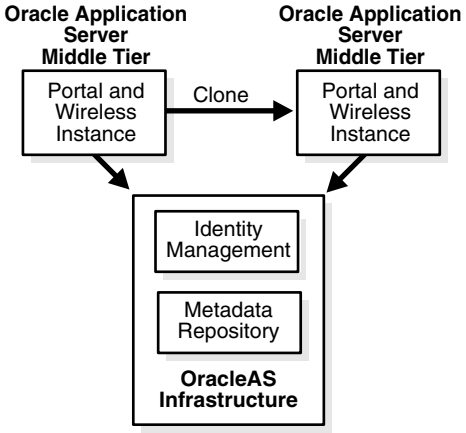


Figure 10–2 shows cloning a Portal and Wireless middle tier that is connected to OracleAS Infrastructure. The Portal and Wireless middle tier is cloned, but the OracleAS Infrastructure is not, because you cannot clone the OracleAS Infrastructure itself. Both instances use the same OracleAS Infrastructure.

Figure 10–2 Cloning a Portal and Wireless Middle Tier



The cloning process works by copying all files from the source Oracle home to the destination Oracle home. Hence, any files used by the source instance that are located outside the source Oracle home's directory structure are not copied to the destination location.

After the files are copied, a set of scripts is used to update the information in key configuration files. For example, all references to the host name and the Oracle home in `httpd.conf` and `webcache.xml` are updated to their new values.

Any applications deployed in the source instance are also copied to the cloned instance and automatically deployed, provided they are located in the source Oracle home's directory structure.

10.2 What Installation Types Can You Clone?

In this release, you can clone the following types of middle-tier installations:

- J2EE and Web Cache middle tier

  See Section 10.5, "Considerations and Limitations for Cloning", especially the subsections Section 10.5.2, Section 10.5.3, and Section 10.5.4, for details of considerations affecting specific components in the cloned Oracle home.

- Portal and Wireless middle tier

  See Section 10.5, "Considerations and Limitations for Cloning", especially the subsections Section 10.5.6 and Section 10.5.7, for details of considerations affecting specific components in the cloned Oracle home.

- Business Intelligence and Forms middle tier.

  - Business Intelligence and Forms
  - Oracle Application Server Forms and Reports Services. For this release, this component is shipped in a separate CD-ROM with Oracle Application Server.
  - Business Intelligence Services. For this release, this component is shipped in a separate CD-ROM with Oracle Application Server.

  See Section 10.5, "Considerations and Limitations for Cloning", especially the subsections Section 10.5.9 and Section 10.5.10 for details of considerations affecting specific components in the cloned Oracle home.

Note the following:

- You cannot clone the OracleAS Infrastructure itself. However, you can clone a middle tier that is connected to OracleAS Infrastructure, including OracleAS Metadata Repository and Oracle Identity Management.
- You cannot clone OracleAS Integration B2B, Oracle BPEL Process Analytics, or Oracle BPEL Process Manager.
- The cloned instance must have a different instance name than the source instance. You specify the instance name when you clone the instance, as described in Section 10.4.3, "Cloning the Instance".
- You can clone a middle-tier instance that is a member of an OracleAS Cluster or farm. However, you must remove the instance from an OracleAS Cluster. See Section 10.4.5, "Cloning Instances That Are Members of a Farm or OracleAS Cluster" for more information.

10.3 Understanding the Cloning Process

The cloning process makes use of the cloning functionality in Oracle Universal Installer. The operation is driven by a set of scripts that are included in the Oracle Application Server installation. The following sections describe the processes involved in cloning an instance:

1. Source Preparation Phase
2. Cloning Phases

10.3.1 Source Preparation Phase

At the source, you run the script called `prepare_clone.pl`. This is a Perl script that prepares the source for cloning. It takes a snapshot of the information required for cloning.

During this phase, `prepare_clone.pl` parses files in the source Oracle home to extract and store required values. It also creates an archive using Distributed

Configuration Management (DCM), backs up required files, and runs the OracleAS Wireless clone assistant, if OracleAS Wireless is installed.

Then, you tar the Oracle home directories.

See Section 10.4.2, "Preparing the Source" for specific instructions for preparing the source instance.

10.3.2 Cloning Phases

At the destination, you extract the Oracle home from the tar file. Then, you run the script called `clone.pl`. This is a Perl script that performs all parts of the cloning operation automatically, calling various other utilities and Oracle Universal Installer, as needed. This script makes use of the cloning functionality in Oracle Universal Installer. When you invoke the `clone.pl` script, it goes through the following three phases:

1. Pre-cloning phase

   During this phase, the `clone.pl` script lays the groundwork necessary to ensure that cloning can be done.

2. Cloning phase

   During this phase, the `clone.pl` script invokes Oracle Universal Installer in clone mode with the necessary arguments to perform the Oracle Universal Installer home cloning. This re-instantiates all files (after creating backups of the existing instantiated files), sets environment variables, updates links, and so on. In other words, it repeats all actions that were performed at installation time, with the exception of the file copying.

3. Post-cloning phase

   The postinstallation configuration assistants are not designed to be run again at clone time. Consequently, some of the instance-specific configuration files that should be updated by the configuration assistants are not updated at the end of the Oracle Universal Installer cloning session. Instead, Oracle has created a set of post-cloning scripts that update those files to bring the cloned home to a working state.

   The post-cloning steps performed by the script are:

   a. Sets the new Oracle home in DCM.

   b. Updates configuration files. In this step, many configuration files that have been re-instantiated by Oracle Universal Installer during the cloning phase are restored from their backups. Those files are then updated with the new values that reflect the new environment, if needed. For example, if a file has a reference to the source Oracle home, that reference is updated to reflect the destination Oracle home.

   c. Calls the home's `chgiphost` command to change the host name and IP number information in the cloned home. Before calling `chgiphost`, the script must collect the following required information to invoke `chgiphost` in silent mode:

      – Source host name

      – Source IP address

      – Destination host name

      – Destination IP address

Note that when `chgiphost` is run as part of cloning, it does not run all of the configuration tools that are run when `chgiphost` is run standalone (as when you change hostname or domain name).

d. If the source instance is connected to Oracle Internet Directory, adds information about the clone to Oracle Internet Directory.

e. After all the operations for cloning are completed, starts services, as well as the Application Server Control Console, to verify the success of the cloning operation.

Note that you do not need to perform each of these phases manually, because the `clone.pl` script takes care of all three phases automatically. The information is provided only for conceptual understanding.

See Section 10.4.3, "Cloning the Instance" for specific instructions on the tasks you do at the destination.

Files Updated During the Post-Cloning Phase

During the post-cloning phase, a set of important configuration files are restored from their backup versions and updated. Typical changes made to the files include updating environment-specific variables such as host name, Oracle home, and port numbers to their new values.

The following list shows some of the key files that are updated. Note that this is not an exhaustive list of the files being updated.

- *Oracle_Home*`/config/ias.properties`
- *Oracle_Home*`/sysman/emd/targets.xml`
- *Oracle_Home*`/sysman/j2ee/application-deployments/em/emd/orion-web.xml`
- *Oracle_Home*`/diagnostics/config/dcmPlugins.xml`
- *Oracle_Home*`/Apache/Apache/conf/httpd.conf`
- *Oracle_Home*`/Apache/Apache/conf/mod_oc4j.conf`
- *Oracle_Home*`/Apache/Apache/conf/oracle_apache.conf`
- *Oracle_Home*`/Apache/modplsql/conf/dads.conf`
- *Oracle_Home*`/Apache/modplsql/conf/plsql.conf`
- *Oracle_Home*`/Apache/modplsql/conf/cache.conf`
- *Oracle_Home*`/Apache/oradav/conf/moddav.conf`
- *Oracle_Home*`/opmn/conf/opmn.xml`

The format of the paths are shown in UNIX format. For Windows, invert the slashes.

10.4 Cloning Oracle Application Server Instances

To clone an Oracle Application Server instance, you first prepare the source Oracle home. Then, you clone the destination.

10.4.1 Prerequisites for Cloning

For cloning, Perl 5.6.1 or higher must be installed on your system. Before running the cloning Perl scripts, you must set the PERL5LIB environment variable to the path of

the Perl directory in the Oracle home. This path must be the first one listed in the variable definition. For example:

- On UNIX

```
setenv PERL5LIB $Oracle_Home/perl/lib/5.6.1:$Oracle_Home/perl/lib/site_
perl/5.6.1:$PERL5LIB
```

- On Windows:

```
set PERL5LIB=%Oracle_Home%\perl\5.6.1\lib;%OH%\perl\
5.6.1\lib\MsWin32-x86;%Oracle_Home%\perl\site\5.6.1\lib;%PERL5LIB%
```

10.4.2 Preparing the Source

To prepare the source Oracle home to be cloned, take the following steps at the source instance:

1. Change to the following directory:

```
(UNIX) ORACLE_HOME/clone/bin
(Windows) ORACLE_HOME\clone\bin
```

2. Run the script `prepare_clone.pl`. This script prepares the source to be cloned.

   The command line for the script has the following format:

```
perl prepare_clone.pl [ORACLE_HOME=OH_dir]
                      [-oid_password OIDPassword]
                      [-oid_admin OIDadmin]
                      [-silent]
                      [-debug]
                      [-help]
```

   Table 10–1 describes the parameters and options for the `prepare_clone.pl` script.

Table 10–1 Parameters and Options for the prepare_clone.pl Script

Parameter or Option	Description
ORACLE_HOME	The complete directory specification for the source Oracle home. If you do not supply this parameter, the script uses the ORACLE_HOME environment variable, if it exists. If the environment variable does not exist, the script assumes that ORACLE_HOME is the directory from which the script is being run. Do not use a trailing slash (UNIX) or backslash (Windows) when specifying the Oracle home. Use the value that was provided during installation; do not use symbolic links. If ORACLE_HOME is invalid, the script exits and logs an error to standard output (STDOUT).
-oid_password	The Oracle Internet Directory password. If the original installation required that the user specify an Oracle Internet Directory password (such as an instance connected to Oracle Identity Management), this option is required. If you do not supply the option, but one is needed, the script prompts the user for the password.
-oid_admin	The Oracle Internet Directory admin value. If you do not specify this option, the script uses a default value of `cn=orcladmin`.
-silent	Runs the script in silent mode. If the command line does not contain the required password-related options, the script exits.

Table 10–1 (Cont.) Parameters and Options for the prepare_clone.pl Script

Parameter or Option	Description
-debug	Runs the script in debug mode.
-help	Prints the usage for the script.

3. Archive and compress the source Oracle home, using your preferred tool for archiving. For example, you can use WinZip on Windows and tar and gzip on UNIX. Make sure that the tool you are using preserves the permissions and timestamps of the files. The following example shows how to archive and compress the source on UNIX:

```
cd Source_Oracle_Home
tar cf - * | gzip > oracleas.tar.gz
```

The tar utility may issue warnings if the sticky bit is set on some files. You can safely ignore these warnings.

Note that you should not use the `jar` utility to archive and compress the Oracle home.

10.4.3 Cloning the Instance

At the destination, to clone the source instance, take the following steps:

1. Copy the compressed Oracle home from the source machine to the destination machine.
2. Extract the compressed Oracle home into a directory, which will become the new Oracle home at the destination location. Use your preferred tool to extract the compressed files. For example, you can use WinZip on Windows and tar and gunzip on UNIX. Make sure that the tool you are using preserves the permissions and timestamps of the files. The following example shows how to extract the files on UNIX:

```
mkdir -p Destination_Oracle_Home
gunzip  <  Dir_Containing_Tar/oracleas.tar.gz | tar xf -
```

> **Note:** Make sure that the tar and gzip/gunzip versions on the source and destination machines are compatible. You may encounter problems unzipping the archive if these versions differ.

3. Change to the following directory:

```
(UNIX) ORACLE_HOME/clone/bin
(Windows) ORACLE_HOME\clone\bin
```

4. Run the `clone.pl` script. You must have write permission to the directory containing the Oracle inventory file. (See Section 10.4.4, "Locating and Viewing Log Files" for information about the location of the Oracle inventory directory.)

The command line for the script has the following format:

```
perl clone.pl ORACLE_HOME=OH_dir
              ORACLE_HOME_NAME=OH_Name
              -instance Instance_Name
              [-ias_admin_old_pwd Old_Ias_Admin_Password]
              [-ias_admin_new_pwd New_Ias_Admin_Password]
```

```
[-oid_password OIDPassword]
[-dcm_schema_pwd DCMPassword]
[-lbr {true|false}]
[-Ostring]
[-silent]
[-debug]
[-help]
```

Table 10–2 describes the parameters and options for the `clone.pl` script.

Table 10–2 Parameters and Options for the clone.pl Script

Parameter or Option	Description
ORACLE_HOME	**Required.** The complete directory specification for the destination Oracle home. This parameter is required. If you do not supply this parameter, or if the value is invalid, the script exits. Do not use a trailing slash (UNIX) or backslash (Windows) when specifying the Oracle home.
ORACLE_HOME_NAME	**Required.** The name for the destination Oracle home (the Oracle home of the clone.)
-instance	**Required.** The instance name for the clone. The instance name must be different from the source instance and any other instances that use the same OracleAS Infrastructure or that are part of the same OracleAS Cluster or farm.
-ias_admin_old_pwd	**Required.** The administrator password for Oracle Application Server for the source instance. If you do not supply this option and the script is not running in silent mode, the script prompts the user for the password.
-ias_admin_new_pwd	**Required.** A new password for administrator for Oracle Application Server for the cloned instance. If you do not supply this option and the script is not running in silent mode, the script prompts the user for the password.
-oid_password	The Oracle Internet Directory password. If the original installation required that the user specify an Oracle Internet Directory password (such as an instance connected to Oracle Identity Management), this option is required. If you do not supply this option, but one is needed, the script prompts the user for the password.
-dcm_schema_pwd	The Distributed Configuration Management (DCM) schema password. If the original installation required that the user specify a DCM schema password (such as a J2EE instance connected to a OracleAS Metadata Repository, but not to Oracle Identity Management), this option is required. If you do not supply this option, but one is needed, the script prompts the user for the password. To find the current encrypted password, use the following command: `SELECT password FROM dba_users WHERE username='DCM';`
-lbr	Whether or not a Load Balancing Router is used. The default is `true`. If a Load Balancing Router is not used, you must specify `false`. If you are cloning a Portal and Wireless instance and you specify `false`, the cloning process overwrites the configuration entries for the source instance, which are stored in the OracleAS Metadata Repository. See Section 10.5.6 and Section 10.7.2 for more information.

Table 10–2 (Cont.) Parameters and Options for the clone.pl Script

Parameter or Option	Description
-O	Specifies that any text following the option is passed to the Oracle Universal Installer command line. For example, you can use this option to pass the location of the `oraparam.ini` file to be used by Oracle Universal Installer, by using the following code: `'-O-paramFile C:\OraHome_1\oui\oraparam.ini'` Note that if the text you want to pass contains spaces or other delimiting characters, you must enclose the option in double quotation marks ("). To pass multiple parameters to Oracle Universal Installer using this option, you can either pass all parameters with a single -O option or pass individual parameters using multiple -O options.
-silent	Runs the script in silent mode. If the command line does not contain the required password-related options, the script exits.
-debug	Runs the script in debug mode.
-help	Prints the usage for the script.

For example:

```
perl clone.pl ORACLE_HOME=/opt/oracle/Ora_1012_B
              ORACLE_HOME_NAME=OH_1012_B
              -instance Portal_B
              -ias_admin_old_pwd my_old_ias_pass
              -ias_admin_new_pwd my_new_ias_pass
              -oid_password my_oidpwd
              -dcm_schema_pwd my_DCM_pass
              -lbr true
              '-O-paramFile /var/opt/oracle/oui/oraparam.ini'
              -silent
```

5. On UNIX, run the `root.sh` script in the Oracle home so that the cloned instance works properly. You must be logged in as the root user to run the script. The script is located in the cloned instance's Oracle home directory.

   For example:

   ```
   $ORACLE_HOME/root.sh
   ```

Now, the cloned instance's configuration is identical to that of the source instance. Application Server Control Console and OPMN are able to start and stop all processes in the cloned instance, including any OC4J custom instances. All applications deployed should be visible and able to run as expected.

10.4.4 Locating and Viewing Log Files

The cloning script invokes multiple tools, each of which generates its own log files. However, the following log files, which are generated by Oracle Universal Installer and the cloning scripts, are the key ones of interest for diagnostic purposes:

- *Oracle_inventory*`/logs/cloneActions`*timestamp*`.log`: Contains a detailed log of the actions that occur during the Oracle Universal Installer part of the cloning.
- *Oracle_inventory*`/logs/oraInstall`*timestamp*`.err`: Contains information about errors that occur when Oracle Universal Installer is running.

- *Oracle_inventory*/logs/oraInstall*timestamp*.out: Contains other miscellaneous messages generated by Oracle Universal Installer.
- *Oracle_Home*/clone/logs/clone*timestamp*.log: Contains a detailed log of the actions that occur during the precloning and cloning operations.
- *Oracle_Home*/clone/logs/error*timestamp*.log: Contains information about errors that occur during the precloning and cloning operations. In addition, it contains all messages written to standard error (STDERR) by the multiple tools that are invoked by the cloning script. Depending upon the tool, some of these messages may be informational messages or error messages.

The format of the path is shown in UNIX format. For Windows, invert the slashes.

Note: To find the location of the Oracle Inventory directory:

- On UNIX systems, look in /var/opt/oracle/oraInst.loc or /etc/oraInst.loc
- On Windows systems, the location can be obtained from the registry: HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\INST_LOC

After the clone.pl script finishes running, consult these log files to get more information about the cloning process. To view the log files from Application Server Control Console, take the following steps:

1. Select **Logs** from the Home page.
2. In the View Logs page, select **ASClone** from the **Available Components** box. Click **Move** to move the selection to the **Selected Components** box.
3. Click **Search.**

   The log files are displayed in the **Results** table.
4. To view the log, click the log name in the **Log File** column.

10.4.5 Cloning Instances That Are Members of a Farm or OracleAS Cluster

You can clone a middle-tier instance that is a member of a farm or an OracleAS Cluster. However, you must remove the instance from the OracleAS Cluster before beginning the cloning operation.

The following information describes how cloning works when the source instance is a member of a farm or an OracleAS Cluster:

- File-based repository

  If the source instance is connected to a file-based repository in a separate instance, the cloned instance will be a member of the same farm as the source instance. If the source instance is connected to a file-based repository in the same instance (that is, it is the host of the file-based repository), the cloned instance will be the host of a new file-based repository.
- File-based repository and Oracle Identity Management

  If the source instance is connected to a file-based repository and Oracle Identity Management in a separate instance, the cloned instance will be a member of the same farm as the source instance and will be connected to the same Oracle Identity Management as the source instance. If the source instance is connected to a

file-based repository and Oracle Identity Management in the same instance (that is, it is the host of the file-based repository), the cloned instance will be the host of a new file-based repository and will it be connected to the same Oracle Identity Management.

- OracleAS Metadata Repository

  If the source instance is connected to OracleAS Metadata Repository, the cloned instance will be connected to the same OracleAS Metadata Repository. The cloned instance will be a member of the same farm as the source instance.

- OracleAS Metadata Repository and Oracle Identity Management

  If the source instance is connected to OracleAS Metadata Repository and Oracle Identity Management, the cloned instance will be connected to the same OracleAS Metadata Repository and Oracle Identity Management. The cloned instance will be a member of the same farm as the source instance.

  One example of this scenario is a Portal and Wireless installation, which requires OracleAS Metadata Repository and Oracle Identity Management.

- No repository, but Oracle Identity Management

  If the source instance is not connected to a repository, but is connected to Oracle Identity Management in a separate instance, the cloned instance will be connected to that same Oracle Identity Management.

If you remove an instance from an OracleAS Cluster, you can add it, and the cloned instance, to the OracleAS Cluster after you complete the cloning operation.

10.5 Considerations and Limitations for Cloning

The following sections provide details of considerations and limitations affecting cloning in general and specific components in the cloned Oracle home:

- General Considerations and Limitations for Cloning
- Considerations for Cloning Oracle HTTP Server
- Considerations for Cloning Oracle Application Server Containers for J2EE (OC4J)
- Considerations for Cloning OracleAS Web Cache
- Considerations for Cloning Application Server Control
- Considerations for Cloning OracleAS Portal
- Considerations for Cloning OracleAS Wireless
- Considerations for Cloning OracleBI Discoverer
- Considerations for Cloning OracleAS Forms Services
- Considerations for Cloning OracleAS Reports Services
- Considerations for Cloning OracleAS Forms and Reports Services

10.5.1 General Considerations and Limitations for Cloning

For this release, you cannot clone the following:

- OracleAS Infrastructure components (Oracle Identity Management and OracleAS Metadata Repository)
- Developer Kits, including Oracle Content Management SDK (Oracle CM SDK)

- Installations that include Oracle Application Server Integration B2B, Oracle Application Server Integration InterConnect, Oracle BPEL Process Analytics
- Adapters, including Oracle Application Server Adapters and Oracle Application Server Integration InterConnect Adapters
- Installations that include Oracle Workflow

Note the following important additional considerations about cloning:

- The user may need to update the security certificates to match the new host name and to set up the certificates. See Chapter 15 for information about managing wallets and certificates.
- If an instance is part of an OracleAS Cluster, you must remove the instance from the cluster before performing the cloning operation.

  See Section 10.4.5, "Cloning Instances That Are Members of a Farm or OracleAS Cluster" for more information.
- Cloning an Oracle Application Server middle tier that is connected to an OracleAS Infrastructure will result in a new Oracle Application Server middle tier connected to the same OracleAS Infrastructure as the source instance. To join a different OracleAS Infrastructure, you can reassociate the middle tier with a different OracleAS Infrastructure, as described in Section 9.5, "Changing the Metadata Repository Used by a Middle-Tier Instance".
- If you have changed the default file permissions for configuration files, those file permissions are not preserved by cloning.
- User customizations for the following components are not preserved. The status of these components are reset to the default:
  - Log Loader
  - Oracle Application Development Framework
  - Port tunneling
  - UIX
  - XDK
- Cloning does not carry over all the dependencies of the source Oracle home, such as loadable modules or application-specific libraries to the cloned home, because cloning proceeds by copying the entire source Oracle home to the destination Oracle home. Any files outside the source Oracle home are not automatically copied. Hence, any applications that refer to files outside the source Oracle home may not work properly in the cloned home.

  You may need to copy the files manually to the destination host after extracting the archived source Oracle home, but before running the `clone.pl` script.
- If you created symbolic links to files or applications outside the source Oracle home (for example, to Oracle Wallet files that are not stored in the default location), you must re-create the link manually in the cloned home for your applications to work properly.
- The cloning operation generates default ports for the cloned instance. To specify other ports, you can use the `staticports.ini` file as described in Section 10.6.2, "Assigning Custom Ports". If you specify ports less than 1024 on UNIX, the cloned instance will not start during the cloning operation. After the cloning process is completed, you must run the `root.sh` script with root privileges, then start the processes.

- The cloning process does not configure a Load Balancing Router to recognize the cloned instance. If you use a Load Balancing Router in your environment, you must manually configure the Load Balancing Router, including any invalidation port.
- If a cloning operation fails, but it results in the Oracle home being registered with Oracle Inventory, you cannot use the same Oracle home in subsequent cloning operations. Either use another directory and name for the Oracle home in subsequent cloning operations or deinstall the Oracle home before attempting another cloning operation.

10.5.2 Considerations for Cloning Oracle HTTP Server

The following describes important information about cloning Oracle HTTP Server:

- All configuration information in the following files is updated:
  - *Oracle_Home*/Apache/Apache/conf/httpd.conf
  - *Oracle_Home*/Apache/Apache/conf/oracle_apache.conf
  - *Oracle_Home*/Apache/Apache/conf/mod_oc4j.conf
  - *Oracle_Home*/Apache/modplsql/conf/dads.conf
  - *Oracle_Home*/Apache/modplsql/conf/plsql.conf
  - *Oracle_Home*/Apache/modplsql/conf/cache.conf
  - *Oracle_Home*/Apache/oradav/conf/moddav.conf

  The format of the paths are shown in UNIX format. For Windows, invert the slashes.

  The cloning script preserves the source settings and updates these files with new environment parameters.

  Note that cloning only updates the files it knows about, that is, files that are part of the original installation. In particular, cloning does not update configuration files that the user added to the "include" list in files such as `httpd.conf`, `oracle_apache.conf`, `dads.conf`, `plsql.conf`, `olap.conf`, or `moddav.conf`. You can, however, explicitly add the "include" files to the list of files that cloning will update. See Section 10.6.3, "Updating Custom Data" for details on how to update custom settings.
- Cloning preserves all VirtualHost directives in `httpd.conf`. It replaces any references to the source home inside these directives. However, cloning does not change the IP numbers or port numbers that these virtual hosts listen to.

  If these values are not valid for the destination environment, then you must do one of the following:
  - Register these changes with the clone script to be updated during cloning. See Section 10.6.3, "Updating Custom Data" for more information.
  - Update them manually in `httpd.conf` after cloning.
- If you changed the port number in `httpd.conf` to use the Load Balancing Router port rather than the local Oracle HTTP Server or OracleAS Web Cache port, that change is lost after cloning. You must edit the `httpd.conf` file in the cloned home to change the port number to the Load Balancing Router port.

- Cloning is not supported if you are using Oracle HTTP Server based on Apache 1.3 or Apache 2.0. (These are not installed by default, but are included in the companion CD-ROM.)

10.5.3 Considerations for Cloning Oracle Application Server Containers for J2EE (OC4J)

This following lists considerations in cloning OC4J and describes which OC4J components are preserved:

- On the source Oracle home during the prepare phase of the cloning process, do not attempt to undeploy OC4J applications or perform other administrative work for OC4J applications while the `prepare_clone.pl` script is running.
- You must manually register files that contain environment-specific information in custom OC4J instances so that those files are updated during cloning. An example of such a file is `oc4j.properties`. See Section 10.3, "Understanding the Cloning Process" for details.

The following describes which OC4J components are preserved:

- All default OC4J instances are preserved.
- Custom OC4J instances that you created, as well as applications deployed in them, are preserved. However, external dependencies for these applications that are not in the Oracle home are not copied to the cloned home and will be lost.
- If the OC4J instance uses an Oracle HTTP Server instance that is not part of the source Oracle home, the cloning procedure does not update the `mod_oc4j.conf` file for the Oracle HTTP Server. You must manually add the instance to the `mod_oc4j.conf` file.
- OPMN can manage all default and custom cloned OC4J instances.
- Grid Control Console on the cloned home can manage default and custom OC4J instances.
- Data source information in `data-sources.xml` is preserved.
- User configurations in `jms.xml`, `java2.policy`, `jazn.xml`, `jazn-data.xml`, `global-web-application.xml`, and `application.xml` are preserved.

10.5.4 Considerations for Cloning OracleAS Web Cache

The following describes important information about cloning OracleAS Web Cache:

- Settings in `webcache.xml` and `internal.xml` are preserved. Only the Oracle home location, host name, instance name, and port numbers are updated to reflect the new environment.
- If the source OracleAS Web Cache instance is a cache cluster member, you must reconfigure the cache cluster after cloning. The cloned home contains the reference to the other nodes in the cache cluster whereas the other nodes do not know about the new instance. After cloning, you need to either add the cloned instance to the cache cluster or remove the cloned cache from the cluster. See the *Oracle Application Server Web Cache Administrator's Guide* for information about adding caches to a cache cluster.
- If the source OracleAS Web Cache instance is configured to forward requests to more than one origin server (application Web server or proxy server) on the same host as the source OracleAS Web Cache instance, the cloning process will fail.

To work around this limitation, remove all but one of the origin servers from the source OracleAS Web Cache configuration, using OracleAS Web Cache administration tools. See *Oracle Application Server Web Cache Administrator's Guide* for information about removing the origin server.

Note that because the cloning process does not modify definitions of origin servers on remote hosts, you do not need to remove references to those origin servers from the OracleAS Web Cache configuration.

10.5.5 Considerations for Cloning Application Server Control

The Application Server Control Console can manage the same instances that were managed at the source.

The SSL settings of the source instance are preserved, by preserving the `emd.properties` and `emd-web-site.xml` files. In other words, if the source Application Server Control Console was configured for HTTPS, then the cloned Application Server Control Console will be as well.

10.5.6 Considerations for Cloning OracleAS Portal

The following describes important considerations about cloning OracleAS Portal:

- If the cache for OracleAS Portal is located outside the Oracle home, you must manually specify the cache location in the `cache.conf` file. (Any files used by the source instance that are located outside the source Oracle home's directory structure are not moved to the destination location.) The `cache.conf` file is located in the following directory:

  ```
  (UNIX) ORACLE_HOME/Apache/modplsql/conf
  (Windows) ORACLE_HOME\Apache\modplsql\conf
  ```

  In this file, make sure that the location for the PlsqlCacheDirectory directive is valid.

- The settings in the following file are not preserved during the cloning operation:

  ```
  (UNIX) Oracle_Home/portal/pdkjava/providerGroups/iasProviders.xml
  (Windows) Oracle_Home\portal\pdkjava\providerGroups\iasProviders.xml
  ```

  If you added providerGroup entries to the file in the source instance, you must add those entries to the file in the cloned instance.

- OracleAS Portal does not support two instances connected to the same infrastructure unless they are front-ended by a Load Balancing Router.

  Because OracleAS Portal stores only a single set of configuration information, the cloning process overwrites the configuration entries stored in the repository if you do not specify that there is a Load Balancing Router. If you specify that there is a Load Balancing Router, the cloning process does not perform any configuration steps, and therefore it does not overwrite the configuration settings stored in the Portal schemas in the OracleAS Metadata Repository. See Section 10.7.2, "Cloning a Portal and Wireless Instance Front-Ended by a Load Balancing Router" for information about cloning a Portal and Wireless instance that is front-ended by a Load Balancing Router.

  Note that you can reconfigure OracleAS Portal by using the `ptlconfig` command, as shown in the following example:

  ```
  $ORACLE_HOME/portal/config/ptlconfig -dad portal
  ```

- When you are cloning a Portal and Wireless instance in an environment where the instances are front-ended by a Load Balancing Router, you *must* make sure that the cloned instance uses the same ports as the other existing instances. To guarantee that the cloning process will assign the same ports for the cloned instance, you can use the `staticports.ini` file, as described in Section 10.6.2, "Assigning Custom Ports".
- OracleAS Portal creates Oracle Internet Directory users only once, during the first installation of OracleAS Portal. If you clone a Portal and Wireless installation, you use the original password of the `ias_admin` user of the first OracleAS Portal instance associated with the OracleAS Metadata Repository when you log in to OracleAS Portal on the cloned instance. This is true even if you changed the password of the `ias_admin` user of the cloned instance. The password used when Portal users and groups are created in Oracle Internet Directory remains the same even if multiple instances are later connected to this infrastructure.
- Any external applications are stored in the OracleAS Single Sign-On server. If OracleAS Portal is not front-ended by a Load Balancing Router and if there is any change in the URL of an external application, you must update it manually in the external application. To do this, log in directly to the OracleAS Single Sign-On server as a user who has single sign-on Administrator privileges:

  `http://`*`SSO_host:SSO_port`*`/pls/orasso/`

  Then, select the **Administer External Applications** link, select the application, and update the **Login URL.** Alternatively, you can use the **Administer External Applications** link in the SSO Administration portlet in the portal. This portlet requires the same privileges as the OracleAS Single Sign-On Server.

  For more information, see the chapter "Configuring and Administering External Applications" in the *Oracle Application Server Single Sign-On Administrator's Guide*.
- If OracleAS Portal is not front-ended by a Load Balancing Router, you must manually update the registration URL of custom-built Web providers and provider groups that are local to the source instance. A custom-built provider or provider group is any Web provider or provider group that is not seeded by the OracleAS Portal installation. To update the registration, take the following steps:

  For a Web provider:

  1. Click **Edit Registration** for the provider on the Providers tab of the Navigator, under Registered Providers.
  2. Select the **Connection** tab, and change the host and port part of the provider registration URL from the source host and port to the cloned instance host and port. For example:

     `http://`*`clone_host:clone_port`*

  3. Verify that the custom-built provider works properly on the cloned instance, by using the test pages at the URL:

     `http://`*`clone_host:clone_port`*`/webApp/providers/`*`providerName`*

  For a provider group:

  1. Click **Edit Registration** for the provider group on the Providers tab of the Navigator, under Providers Groups.
  2. Change the host and port part of the URL from the source host and port to the cloned instance host and port. For example:

```
http://clone_host:clone_port
```

10.5.7 Considerations for Cloning OracleAS Wireless

The following describes important information about cloning OracleAS Wireless:

- If the Portal and Wireless middle tier is not front-ended by a Load Balancing Router and you do not specify the `-lbr` option when cloning, you must reconfigure the OracleAS Portal and OracleAS Wireless integration as described in "Manually Reconfiguring the Mobile Setup" section of the *Oracle Application Server Portal Configuration Guide*.

10.5.8 Considerations for Cloning OracleBI Discoverer

The following describes important information about cloning OracleBI Discoverer:

- Values set by users after installation in `configuration.xml` are not preserved in the cloned Oracle home. The file is located at:

  ```
  (UNIX) Oracle_Home/discoverer/config/configuration.xml
  (Windows) Oracle_Home\discoverer\config\configuration.xml
  ```

  To work around this problem, reset the overwritten values by copying them from the `configuration.xml` file in the source Oracle home to the `configuration.xml` file in the cloned Oracle home.

10.5.9 Considerations for Cloning OracleAS Forms Services

The following describes important information about cloning OracleAS Forms Services:

- Cloning preserves the users configuration information in the OracleAS Forms Services default configuration files and makes the necessary modifications (such as changing the host name, re-deploying the Forms EAR file and registering the OracleAS Forms Services in the new instance with Oracle Internet Directory) to reflect the new environment.
- Cloning preserves the user settings in the following files:
  - *Oracle_Home*/forms/forms.conf
  - *Oracle_Home*/forms/server/default.env
  - *Oracle_Home*/forms/server/formsweb.cfg
  - *Oracle_Home*/forms/converter.properties
  - *Oracle_Home*/bin/frmcmp.sh
  - *Oracle_Home*/bin/frmcmp_batch.sh
  - *Oracle_Home*/bin/graphicsrun.sh
  - *Oracle_Home*/bin/frmplsqlconv.sh (UNIX)
  - *Oracle_Home*\bin\frmplsqlconv.bat (Windows)

  Unless otherwise specified, the format of the paths are shown in UNIX format. For Windows, invert the slashes.
- Cloning does not preserve the user settings in the following file because the Forms EAR file (`formsapp.ear`) is redeployed in the cloned instance:

  ORACLE_HOME/j2ee/OC4J_BI_Forms/applications/formsapp/formsweb/WEB-INF/web.xml

- If the OracleAS Forms Services executable files (.fmx) are residing on the local drive on the source machine but not in the Oracle home, then you must copy these files to the machine hosting the cloned instance. If the .fmx files are residing on a shared network drive or residing inside the Oracle home, then no action is required.
- If you have created customized configuration files to replace the default Forms configuration files (`formsweb.cfg`, `default.env`, and template HTML files) and if these files are in the Oracle home, then these files will be present in the cloned Oracle home instance. However, any references to the source Oracle home will not be automatically replaced with references to the cloned Oracle home settings. You should manually make the changes in the cloned Oracle home.
- If you have redeployed the Forms EAR file (`formsapp.ear`) using Enterprise Manager, you must repeat this action in the cloned Oracle home.

10.5.10 Considerations for Cloning OracleAS Reports Services

The following describes important information about cloning OracleAS Reports Services:

- Cloning preserves the settings in the following files: `rwservlet.properties`, `rwserver.template`, `rwbridge.template`, `rwnetwork.template`, *`network-files`*`.conf`, *`server-name`*`.conf`, `cgicmd.dat`, `proxyinfo.xml`, `xmlpds.conf`, `jdbcpds.conf`, and `textpds.conf`.
- Cloning preserves the port settings in the following files: `rwbridge.template`, `rwnetwork.template`, *`network-files`*`.conf`, *`bridge-network-files`*`.conf`, *`server-name`*`.conf`, `tsnnames.ora`.
- Cloning preserves OracleAS Reports Services server settings in `opmn.xml` and `targets.xml`.
- The following files are not cloned:
  - All files in the `cache` subdirectory
  - The data (.dat) files in the `server` subdirectory
- The name of the in-process server in the cloned instance will not be the same as the name of the source instance. The name takes the following form:

  `rep_hostname_OracleHomeName`

  If you have registered the in-process server in OracleAS Portal, you must manually change the name of the in-process server in OracleAS Portal and associate the new name with reports that use the in-process server.
- The user name and password in the identifier element in the server configuration files in the cloned instance are randomly generated; they are not the same as the source instance. You do not need to change them because they are used only by Enterprise Manager.

  However, if you want to change the user name and password, take the following steps:

  1. In the OracleAS Reports Services configuration file, *`servername`*`.conf`, modify the <identifier> element to specify the user name and password, and set the encrypted attribute to `no`. For example:

     ```
     <identifier confidential="yes" encrypted="no"> scott/tiger</identifier>
     ```

2. Restart the OracleAS Reports Services server.
3. Edit the *ORACLE_HOME*/sysman/emd/targets.xml.
4. Search for the target with TYPE="oracle_repserv" and DISPLAY_NAME="Reports Server" *server_name*.
5. In the entry, set the Username and Password properties to the same values as you used in the <identifier> element of the OracleAS Reports Services configuration file and set the ENCRYPTED attribute to FALSE.
6. Restart Enterprise Manager.

   When Enterprise Manager restarts, it sets the ENCRYPTED attribute to TRUE, encrypting the password.

10.5.11 Considerations for Cloning OracleAS Forms and Reports Services

The following describes important information about cloning OracleAS Forms and Reports Services (installed from a separate CD):

- If OracleAS Forms and Reports Services is associated with a Metadata Repository, you must update the ias.properties file in the source Oracle home, before you run the prepare_clone.pl script. Change the value of DatabaseManagedClusterSupport from false to true, as shown in the following example:

  ```
  DatabaseManagedClusterSupport=true
  ```

  If OracleAS Forms and Reports Services is not associated with a Metadata Repository, the value of DatabaseManagedClusterSupport must be false.
- On Windows, before you run the prepare_clone.pl script, you must update the ias.properties file in the source Oracle home, changing the value of installType from FRServices to Business, as shown in the following example:

  ```
  installType=Business
  ```

For information about other considerations for cloning OracleAS Forms Services and OracleAS Reports Services, see the following sections:

- Section 10.5.9, "Considerations for Cloning OracleAS Forms Services"
- Section 10.5.10, "Considerations for Cloning OracleAS Reports Services"

10.6 Customizing the Cloning Process

The default cloning process is adequate for most cases. However, you can customize some aspects of the cloning process by performing manual configuration steps, as described in the following sections:

- Specifying Oracle Universal Installer Parameters
- Assigning Custom Ports
- Updating Custom Data

10.6.1 Specifying Oracle Universal Installer Parameters

When cloning an instance, you do not directly invoke Oracle Universal Installer. However, you can still pass information to Oracle Universal Installer indirectly, by specifying any Oracle Universal Installer parameters that you normally specify on the

command line in the configuration file `cs.properties`. This file is located in the following directory:

```
(UNIX) ORACLE_HOME/clone/ias/config
(Windows) ORACLE_HOME\clone\ias\config
```

For example, to specify a nondefault location for the Oracle inventory file on UNIX, you can add the following line to the `cs.properties` file:

```
clone_command_line= -invptrloc /private/oracle/oraInst.loc
```

To specify multiple arguments, separate each argument with a space.

In addition, you can specify information to be passed to the Oracle Universal Installer command line by using the `-O`*`string`* option. For example, you can use this option to pass the location of the `oraparam.ini` file to be used by Oracle Universal Installer, by using the following code:

```
'-O-paramFile C:\OraHome_1\oui\oraparam.ini'
```

10.6.2 Assigning Custom Ports

By default, the cloning script automatically assigns free ports and lists them in the file `portlist.ini`. The algorithm for assigning default ports when cloning is the same as that used when installing Oracle Application Server.

When installing a new Oracle Application Server instance, you can specify the ports to be used by listing them in a `staticports.ini` file. Then, this file is passed as the value of a parameter when calling Oracle Universal Installer. For more information on how ports are assigned and on using the `staticports.ini` file, please refer to the *Oracle Application Server Installation Guide* for your platform.

When cloning an instance, you do not directly invoke Oracle Universal Installer. Hence, you cannot assign custom ports by specifying a `staticports.ini` file on the command line. However, you can still pass port information to Oracle Universal Installer indirectly, by specifying the location of the `staticports.ini` file in the configuration file `cs.properties`.

For example, if you want to use ports less than 1024, you can specify them in the `staticports.ini` file and specify the location of the `staticports.ini` file in the `cs.properties` file.

To assign custom ports during cloning:

1. List the port numbers in the `staticports.ini` file, as explained in the *Oracle Application Server Installation Guide* for your platform.
2. Specify the location of the `staticports.ini` file by adding the following line to the `cs.properties` file:

   ```
   clone_command_line =
   oracle.iappserver.iapptop:szl_PortListSelect="{\"YES\",
   \"/tmp/staticports.ini\"}"
   ```

The ports listed in the `staticports.ini` file are read during cloning, and Oracle Universal Installer assigns the port numbers accordingly.

If you specify ports less than 1024 on UNIX, the cloned instance will not start during the cloning operation. After the cloning process is completed, you must run the `root.sh` script with root privileges, then start the processes.

Note: By default, Oracle Universal Installer saves all user input at installation and uses it to automate the actions when cloning. As a result, if you used a `staticports.ini` file to install the source instance, Oracle Universal Installer will, by default, use the same `staticports.ini` file. This happens even if you do not specify a `staticports.ini` file when you clone the instance. To override this behavior and let Oracle Universal Installer generate new ports, use the following line to the `cs.properties` file:

```
oracle.iappserver.iapptop:szl_PortListSelect="{\"NO\", \"\"}"
```

10.6.3 Updating Custom Data

By default, the cloning scripts update key configuration files in the Oracle home so they contain the correct information for the destination environment. Section 10.4.3, "Cloning the Instance" contains a partial listing of the files that are updated.

You can modify the default cloning process to update custom data that is not updated by default. Information about which files to update during cloning and which entries to update in those files is contained in another set of files, which are read by the cloning scripts. By editing these files, you can:

- Preserve changes you have made to files present in the source Oracle home that are not updated by default during cloning
- Preserve changes you have made to files that are updated by default during cloning, but which are not normally preserved by the cloning process

These changes are made by a Java utility called FileFixer, which searches for specific text strings in a file by matching regular expressions, and updates them to their new values. Note that FileFixer searches for patterns one line at a time. It cannot match patterns across lines.

The changes that you can make include the following:

- Change the host name in a file.

  To do this, for the file in which the host name needs to be changed, add the path name for the file, relative to the Oracle home, to the following file:

  ```
  (UNIX) ORACLE_HOME/chgip/config/hostname.lst
  (Windows) ORACLE_HOME\chgip\config\hostname.lst
  ```

- Update all occurrences of the Oracle home in a file from the old to the new value.

  To do this, add a `replace` tag in the XML configuration file, `fixup_script.xml.tmpl`. This file is located in the following directory:

  ```
  (UNIX) ORACLE_HOME/clone/ias/config
  (Windows) ORACLE_HOME\clone\ias\config
  ```

  The value of the file_name attribute specifies the name and location of the file in which the replacement should occur. For example, the following tag updates the Oracle home value in the file `dcmPlugins.xml`.

  ```
  <cfw:operation>
     <replace file_name="%NEW_HOME%/sysman/config/dcmPlugins.xml">
           <cfw:replaceCommand>
              <cfw:pattern>(%OLD_HOME%)</cfw:pattern>
              <cfw:value_ref>1</cfw:value_ref>
              <cfw:new_value>%NEW_HOME%</cfw:new_value>
  ```

```
        </cfw:replaceCommand>
    </replace>
</cfw:operation>
```

- Extract a value from file1 and use it to replace a value in file2.

  To do this, add an `alter` tag in `fixup_script.xml.tmpl`. For example, adding the following tag extracts the new HTTP port number from the `portlist.ini` file in the cloned home and uses it to replace the old port number in `targets.xml`, as shown in the following example:

```
<cfw:operation>
    <alter cluster="false" alter_file_name="%NEW_HOME%/sysman/emd/targets.xml"
reference_file_name="%NEW_HOME%/install/portlist.ini">
       <cfw:alterCommand>
          <cfw:pattern>(Oracle HTTP Server port)([ ]*)(=)([ ]*)([0-9]*)
          </cfw:pattern>
          <cfw:value_ref>5</cfw:value_ref>
          <cfw:subst>(Property NAME=&amp;quot;HTTPPort&amp;quot;
                     VALUE=&amp;quot;)([0-9]*)(&amp;quot;)</cfw:subst>
          <cfw:subst_ref>2</cfw:subst_ref>
       </cfw:alterCommand>
    </alter>
</cfw:operation>
```

10.7 Examples of Cloning Application Server Instances

The following sections provide step-by-step instructions for cloning different installation types:

- Using Cloning to Expand an OracleAS Cluster
- Cloning a Portal and Wireless Instance Front-Ended by a Load Balancing Router
- Cloning a Business Intelligence Instance

10.7.1 Using Cloning to Expand an OracleAS Cluster

A common use of cloning is to expand the size of an OracleAS Cluster. Consider a OracleAS Cluster consisting of multiple J2EE and Web Cache middle tiers, with identical configuration and application deployment. To expand the OracleAS Cluster, you want to create a new middle-tier instance that has the same configuration as the other instances and add it to the cluster. This example assumes that the source instance is connected to a file-based repository and is a member of a farm and an OracleAS Cluster.

The strategy for expanding an OracleAS Cluster consists of the following steps:

1. Remove the source instance from the OracleAS Cluster.

   To do this, issue the following commands at the source instance:

   - On UNIX:

     ```
     cd $ORACLE_HOME/dcm/bin
     ./dcmctl leaveCluster
     ```

   - On Windows:

     ```
     cd ORACLE_HOME\dcm\bin
     dcmctl leaveCluster
     ```

2. Prepare the source instance for cloning, using the steps described in Section 10.4.2:
   a. Perform Step 1, as described, in Section 10.4.2.
   b. Perform Step 2 in Section 10.4.2.

      In that step, use a command similar to the following:

      ```
      perl prepare_clone.pl ORACLE_HOME=/opt/oracle/Ora_1012
      ```

   c. Perform Step 3 as described, in Section 10.4.2.
3. Clone the source instance to create a new instance, using the steps described in Section 10.4.3:
   a. Perform Steps 1, 2, and 3 as described, in Section 10.4.3.
   b. Perform Step 4 in Section 10.4.3.

      In that step, you must change the name of the cloned instance, by specifying it on the command line, as shown in the following example:

      ```
      perl clone.pl ORACLE_HOME=/opt/oracle/Ora_1012_B
                    ORACLE_HOME_NAME=OH_1012_B
                    -instance J2EE_B
                    -ias_admin_old_pwd my_old_ias_pass
                    -ias_admin_new_pwd my_new_ias_pass
      ```

      In this example, the instance name for the cloned instance is J2EE_B.
4. Add the source instance to the OracleAS Cluster, using the following command:
   - On UNIX:

     ```
     cd $ORACLE_HOME/dcm/bin
     ./dcmctl joinCluster -cl cluster_name
     ```

   - On Windows:

     ```
     cd ORACLE_HOME\dcm\bin
     dcmctl joinCluster -cl cluster_name
     ```

5. Add the cloned instance to the OracleAS Cluster, using the following command:
   - On UNIX:

     ```
     cd $ORACLE_HOME/dcm/bin
     ./dcmctl joinCluster -cl cluster_name
     ```

   - On Windows:

     ```
     cd ORACLE_HOME\dcm\bin
     dcmctl joinCluster -cl cluster_name
     ```

The cloned instance will be a member of the same farm as the source instance.

Note: Instead of using the command line, you can add the instances to the OracleAS Cluster using Application Server Control Console. You can remove an instance from the OracleAS Cluster using Application Server Control Console.

10.7.2 Cloning a Portal and Wireless Instance Front-Ended by a Load Balancing Router

Consider an environment where you have configured a Portal and Wireless middle-tier instance to be front-ended by a Load Balancing Router. Now, you want to add more instances with identical configuration and application deployment. The source instance is connected to OracleAS Infrastructure (Oracle Identity Management and OracleAS Metadata Repository). You want the new instances to be connected to the same OracleAS Infrastructure as the source instance and the same farm as the source instance.

Before you start the cloning, note the following:

- Make sure that the source instance is configured to work with a Load Balancing Router. See "Configuring Multiple Middle-Tiers with a Load Balancing Router" in the *Oracle Application Server Portal Configuration Guide* for more information.
- Review the limitations and considerations listed in Section 10.5, "Considerations and Limitations for Cloning", especially those in Section 10.5.2, Section 10.5.4, and Section 10.5.6.
- You *must* use the same port numbers for the cloned instance as for the source instance. To review the port numbers used by the source instance, navigate to the Application Server Control Console Home page for the source instance and click **Ports.** To guarantee that the cloning process will assign the same ports for the cloned instance, you can use the `staticports.ini` file, as described in Section 10.6.2.

The strategy for cloning a Portal and Wireless middle tier connected to OracleAS Infrastructure and configured with a Load Balancing Router consists of the following steps:

1. Prepare the source instance for cloning, using the steps described in Section 10.4.2:
   a. Perform Step 1 as described, in Section 10.4.2.
   b. Perform Step 2 in Section 10.4.2.

      In this scenario, you must specify the Oracle Internet Directory password on the command line, as shown in the following example:

      ```
      perl prepare_clone.pl ORACLE_HOME=/opt/oracle/Ora_1012
                            -oid_password my_oidpwd
      ```

   c. Perform Step 3, as described, in Section 10.4.2.
2. To use the same ports as the source instance, use the `staticports.ini` file to specify the port numbers for the cloned instance. See Section D.1.2 for a list of the default ports used by a Portal and Wireless instance.

   See Section 10.6.2 for details on using the `staticports.ini` file.
3. Clone the source instance to create a new instance using the steps described in Section 10.4.3:
   a. Perform Steps 1, 2, and 3, as described, in Section 10.4.3.
   b. Perform Step 4 in Section 10.4.3. However, note the following:
      – You must specify the Oracle Application Server administration password and the Oracle Internet Directory password on the command line.
      – Because OracleAS Portal stores only a single set of configuration information, the cloning process overwrites the configuration entries stored in the repository if you do not specify that there is a Load Balancing

Router. If you specify that there is a Load Balancing Router, the cloning process does not perform any configuration steps, and therefore it does not overwrite the configuration settings stored in the Portal schemas in the OracleAS Metadata Repository. To specify that there is a Load Balancing Router, use the `-lbr` option with the value of `true` in the `clone.pl` command line.

The following example clones a Portal and Wireless instance and names the cloned instance as Portal_B:

```
perl clone.pl ORACLE_HOME=/opt/oracle/Ora_1012_B
              ORACLE_HOME_NAME=OH_1012_B
              -instance Portal_B
              -ias_admin_old_pwd my_old_ias_pass
              -ias_admin_new_pwd my_new_ias_pass
              -oid_password my_oidpwd
              -lbr true
```

When the cloning process is finished, both the source instance and the cloned instance are part of the farm.

Note that when you log in to OracleAS Portal on the cloned instance, you use the original `ias_admin` password of the first OracleAS Portal instance associated with the OracleAS Metadata Repository. See Section 10.5.6 for more information.

4. The cloning process does not configure the Load Balancing Router to recognize the cloned instance. You must manually configure the Load Balancing Router. To configure OracleAS Portal with a Load Balancing Router, see the section, "Configuring Multiple Middle-Tiers with a Load Balancing Router" in *Oracle Application Server Portal Configuration Guide*.

5. From the source instance, using either Application Server Control Console or OracleAS Web Cache Manager, take the following steps:

   a. Add the cloned instance to the OracleAS Web Cache cluster.

   b. Add the cloned origin server as an origin server for the cluster.

   c. Save the OracleAS Web Cache changes and propagate the changes to the cluster.

   See the *Oracle Application Server Web Cache Administrator's Guide* for information configuring OracleAS Web Cache, particularly Chapter 10 about adding caches to a cache cluster and Chapter 8 for information about specifying origin servers. See the section "Configuring Multiple Middle-Tiers with a Load Balancing Router" in the *Oracle Application Server Portal Configuration Guide* for an example of these tasks in an OracleAS Portal environment with a Load Balancing Router.

6. Configure the Load Balancing Router to accept invalidation requests from OracleAS Metadata Repository on a separate port, so that requests are forwarded to OracleAS Web Cache. See the section, "Configuring Multiple Middle-Tiers with a Load Balancing Router" in the *Oracle Application Server Portal Configuration Guide* for more information.

7. Configure the Load Balancing Router to balance the load between the OracleAS Web Cache cluster members.

8. Re-register mod_osso for the source instance with the Load Balancing Router. See the section, "Configuring Multiple Middle-Tiers with a Load Balancing Router" in the *Oracle Application Server Portal Configuration Guide*.

9. Edit the `httpd.conf` file to change the port number to use the Load Balancing Router port rather than the local Oracle HTTP Server port.

   See also: *Oracle Application Server Portal Configuration Guide* for more information about configuring OracleAS Portal

10.7.3 Cloning a Business Intelligence Instance

Consider an environment where you have one Business Intelligence middle-tier instance and you want to add more instances with identical configuration and application deployment. The instance is connected to OracleAS Infrastructure.

Review the limitations and considerations listed in Section 10.5, "Considerations and Limitations for Cloning", especially those in Section 10.5.9 and Section 10.5.10.

The strategy for cloning a Business Intelligence instance consists of the following steps:

1. Prepare the source instance for cloning, using the steps described in Section 10.4.2:
   a. Perform Step 1, as described, in Section 10.4.2.
   b. Perform Step 2 in Section 10.4.2.

      In this scenario, you must specify the Oracle Internet Directory password on the command line, as shown in the following example:

      ```
      perl prepare_clone.pl ORACLE_HOME=/opt/oracle/Ora_1012_BI
                            -oid_password my_oidpwd
      ```

   c. Perform Step 3, as described, in Section 10.4.2.
2. Clone the source instance to create a new instance using the steps described in Section 10.4.3:
   a. Perform Steps 1, 2, and 3, as described, in Section 10.4.3.
   b. Perform Step 4 in Section 10.4.3.

      In that step, you must change the name of the cloned instance, by specifying it on the command line, and you must specify the Oracle Application Server administration password and the Oracle Internet Directory password on the command line, as shown in the following example:

      ```
      perl clone.pl ORACLE_HOME=/opt/oracle/Ora_1012_BI2
                    ORACLE_HOME_NAME=OH_1012_BI2
                    -instance BI2
                    -ias_admin_old_pwd my_old_ias_pass
                    -ias_admin_new_pwd my_new_ias_pass
                    -oid_password my_oidpwd
      ```

      In this example, the instance name for the cloned instance is BI2.
3. During cloning, the data (.dat) files in the OracleAS Reports Services `server` subdirectory are not cloned. As a result, the past, current, and scheduled jobs information is not available. If you want to preserve the scheduled jobs information, take the following steps:
   a. Copy the .dat file from the source instance to the cloned instance. The files are located in:

      (UNIX) *ORACLE_HOME*/reports/server/*server_name*.dat
      (Windows) *ORACLE_HOME*\reports\server*server_name*.dat

For the in-process server, the .dat file name is `rep_sourceHostName_sourceOracleHomeName`, unless you have changed the in-process server name in the `rwservlet.properties` file.

b. Ensure that any external dependencies for running reports in the source Oracle home is made the same in the cloned Oracle home. Some of the dependencies include:

- The location of OracleAS Reports Services on the host
- The REPORTS_PATH setting in the registry on Windows
- The location of the destination
- Distribution details

c. Enter the .dat file name in the server configuration file:

- For the in-process server in the cloned instance, the configuration file name is `rep_cloneHostName_cloneOracleHomeName`, unless you have changed the in-process server name in the `rwservlet.properties` file. You can add the .dat file name to the configuration file in the persistFile element, as shown in the following example:

  ```
  <persistFile filename="source_server_name.dat"/>
  ```

  In the example, *source_server_name* is `rep_sourceHostName_sourceOracleHomeName`, unless you have changed the in-process server name in the `rwservlet.properties` file.

- For the Standalone server, if you want to run the server with the same name in the cloned instance, the server will use the copied .dat file with the name `server_name.dat`.

  However, if you want to run a server with a different name in the cloned instance, and you want to use the job information stored in the .dat file in the source home, you must add the .dat file name to the server configuration file in the persistFile element, as shown in the following example:

  ```
  <persistFile filename="source_server_name.dat"/>
  ```

Note: OracleAS Reports Services server cache files are not copied as part of the cloning process. OracleAS Reports Services server's past jobs information stored as part of persistFile is not available for use in the cloned instance.

11

Staging a Test Environment from a Production Environment

This chapter describes copying a 9.0.4.*x* or 10.1.2.0.*x* production environment that includes a middle-tier instance, Identity Management with a Metadata Repository, and a product Metadata Repository for OracleAS Portal and OracleBI Discoverer metadata to a test environment. The Identity Management and the product metadata can share the same Metadata Repository, or they each use a dedicated Metadata Repository. You use this procedure for the purpose of:

- Periodically taking a snapshot of the production environment
- Testing an upgrade to 10.1.2.0.2
- Testing to apply a patchset

If you need to only create additional middle-tier nodes, perform the procedures in Chapter 10 instead.

This chapter contains the following topics:

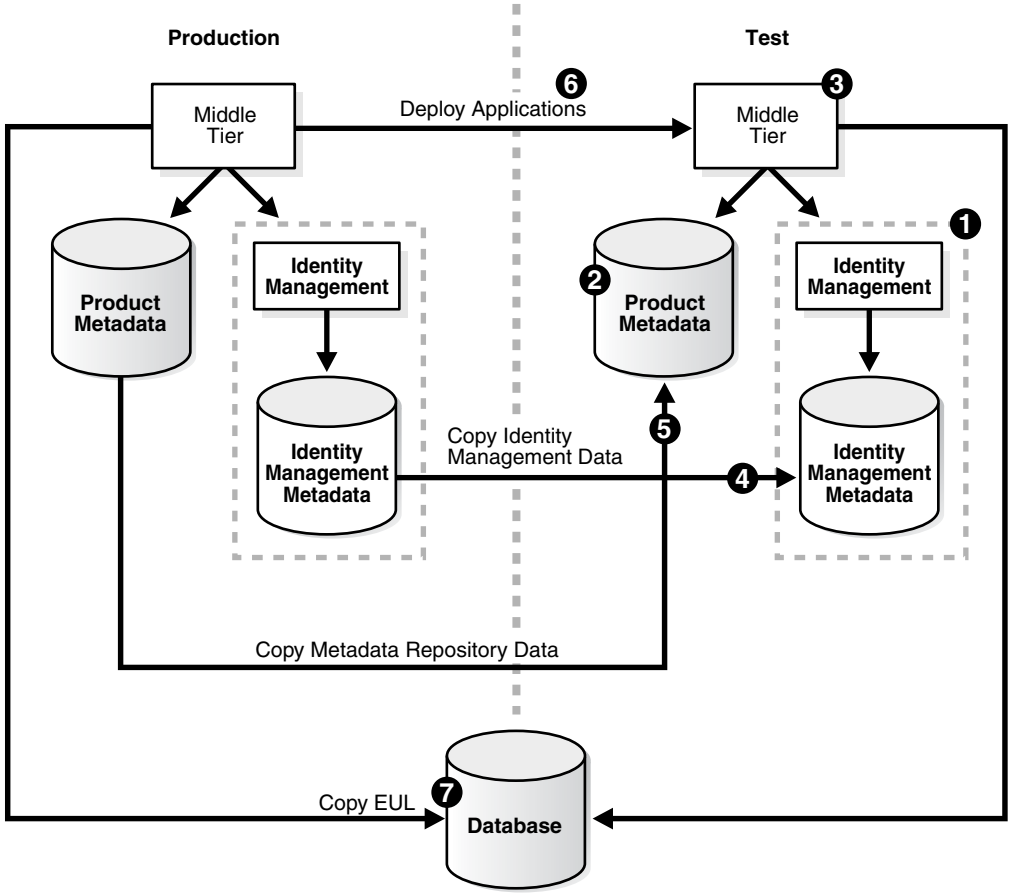
11.1 Creating a Test Environment from a Production Environment and Copying Metadata

In this configuration, you have an existing production environment that includes a middle-tier instance, an Identity Management installation with a Metadata Repository, a product Metadata Repository for OracleAS Portal and OracleBI Discoverer metadata, and customer database. You would like to create a copy of this production environment on the test environment.

For this configuration, you create a test environment by installing and subsequently moving production data for the Identity Management and product Metadata Repository. You then install a test middle-tier instance and deploy applications to this instance. You use the existing customer database, importing the OracleBI Discoverer data.

Figure 11–1 shows this configuration.

Figure 11–1 Creating a Test Environment from a Production Environment and Copying Metadata



11.1.1 Preexisting Configuration Assumptions

This use case assumes the following configuration:

- The production environment includes a middle-tier instance, an Identity Management installation with a Metadata Repository, and an additional Metadata Repository for (OracleAS Portal and OracleBI Discoverer) product metadata. This procedure also applies to configurations in which both Identity Management and product metadata share the same Metadata Repository.
- The production environment accesses a customer database that contains OracleBI Discoverer End User Layers (EULs).
- The production environment is configured with release 9.0.4.*x* or 10.1.2.0.*x*.
- The new test environment has not been created.

11.1.2 Procedure

This procedure contains the following tasks:

- Prerequisite Export Steps
- Task 1: Install Test Identity Management
- Task 2: Set Up Test Product Metadata Repository
- Task 3: Install the Test Middle Tier

- Task 4: Copy Data from Production Identity Management to the Test Environment
- Task 5: Copy Data from the Production Product Metadata Repository to the Test Environment
- Task 6: Deploy Applications to the Test Middle Tier
- Task 7: Copy OracleBI Discoverer Data
- Task 8: Clean Up the Test Environment

In the procedures, *INFRA_HOME* references the Oracle home of OracleAS Infrastructure, and *MIDTIER_HOME* references the Oracle home of Oracle Application Server.

> **See Also:** *Oracle Identity Management User Reference* for more information about the Oracle Internet Directory management tools mentioned in this procedure, including `ldapaddmt`, `ldapmodify`, `ldapsearch`, `ldifwrite`, `bulkload`, and `bulkdelete`

Note: To run the Oracle Internet Directory management tools on Windows operating systems, you need one of the following UNIX emulation utilities:

- Cygwin 1.3.2.2-1 or later. Visit: `http://sources.redhat.com`
- MKS Toolkit 6.1. Visit: `http://www.datafocus.com`

Note: Before running any of the Oracle Internet Directory management tools mentioned throughout this procedure, you must set the `NLS_LANG` environment variable before running the directory command.

To determine the character set, run the following query from SQL*Plus as the `SYS` user:

```
SQL> SELECT userenv('language') from dual;
```

The output looks similar to the following:

```
USERENV('LANGUAGE')
----------------------------------------------------
AMERICAN_AMERICA.WE8MSWIN1252
```

To set the `NLS_LANG` environment variable:

```
setenv NLS_LANG AMERICAN_AMERICA.WE8MSWIN1252
```

Prerequisite Export Steps

To obtain information and export data from the production environment prior to setting up the test environment:

1. Collect the information needed to access the production environment:
   - Operating system login credentials to computer on which the Metadata Repositories reside
   - Operating system login credentials and Oracle home location of the middle-tier computer and location of the Oracle home

- Oracle Internet Directory connection information, including the host name, port, and administration DN and password
- ODS schema password. Use the following command from the Identity Management Metadata Repository repository, not the product Metadata Repository:

```
PRODUCTION_INFRA_HOME/bin/ldapsearch -h production_oid_host
-p production_oid_port -D cn=orcladmin -w production_orcladmin_passwd
-b "orclreferencename=metadata_rep_global_db_name, cn=IAS Infrastructure
Databases, cn=ias, cn=products, cn=oracleContext"
-s base "orclresourcename=ods" orclpasswordattribute
```

  In the syntax:

  production_oid_host is the host of the production directory server.

  production_oid_port is the LDAP port of the production directory server.

  production_orcladmin_password is the password of the superuser DN (cn=orcladmin).

  metadata_rep_global_db_name is the service name of the production directory Metadata Repository.

- PORTAL schema password. Use the following command:

```
PRODUCTION_INFRA_HOME/bin/ldapsearch -h production_oid_host
-p production_oid_port -D cn=orcladmin -w production_orcladmin_passwd
-b "orclreferencename=metadata_rep_global_db_name, cn=IAS Infrastructure
Databases, cn=ias, cn=products, cn=oracleContext"
-s one "orclresourcename=portal" orclpasswordattribute
```

- ORASSO schema password. Use the following command:

```
PRODUCTION_INFRA_HOME/bin/ldapsearch -h production_oid_host
-p production_oid_port -D cn=orcladmin -w production_orcladmin_passwd
-b "orclreferencename=metadata_rep_global_db_name, cn=IAS Infrastructure
Databases, cn=ias, cn=products, cn=oracleContext"
-s one "orclresourcename=orasso" orclpasswordattribute
```

2. Export data from the production Oracle Internet Directory:

   a. Query the production directory server to find the default realm:

```
PRODUCTION_INFRA_HOME/bin/ldapsearch -h production_oid_host
-p production_oid_port -D cn=orcladmin -w production_orcladmin_passwd
-b "cn=common, cn=products, cn=oracleContext"
-s base "objectclass=*" orcldefaultSubscriber
```

   The response returns the orcldefaultSubscriber in LDIF format:

```
cn=Common,cn=Products,cn=OracleContext
orcldefaultsubscriber=dc=us,dc=company,dc=com
```

   You will need this information later when you install the test Identity Management.

   b. Query the production directory server to determine the configured user search bases and group search bases:

```
PRODUCTION_INFRA_HOME/bin/ldapsearch -L
-h production_oid_host -p production_oid_port -D cn=orcladmin
-w production_orcladmin_passwd
-b "cn=common, cn=products, cn=OracleContext, orcldefaultsubscriber"
```

```
-s base "objectclass=*" orclCommonUserSearchBase orclCommonGroupSearchBase
orclCommonNicknameattribute > /tmp/OracleContext.ldif
```

In the syntax, *orcldefaultsubscriber* is the `orcldefaultsubscriber` value returned in Step 2a of this procedure.

The response looks similar to the following:

```
dn: cn=Common,cn=Products,cn=OracleContext,dc=us,dc=company,dc=com
orclcommonusersearchbase: cn=users,dc=us,dc=company,dc=com
orclcommongroupsearchbase: cn=Groups,dc=us,dc=company,dc=com
orclcommonnicknameattribute: uid
```

c. Export all users from the production directory server:

```
PRODUCTION_INFRA_HOME/bin/ldifwrite
-c production_oid_net_service_name -b "orclcommonusersearchbase"
-f /tmp/user.ldif
```

In the syntax:

production_oid_net_service_name specifies the net service name for the production directory, as defined in the `tnsnames.ora` file.

orclcommonusersearchbase is the `orclcommonusersearchbase` subtree written out in LDIF format, such as `cn=users,dc=us, dc=company,dc=com`.

When prompted for the directory password, enter the password of the `ODS` user. This password defaults to the password assigned for administrator `ias_admin` during installation. You should run the `ldifwrite` command once for each user search base value (`orclcommonusersearchbase`) from the Oracle Context settings query.

d. Export all groups from the production directory server:

```
PRODUCTION_INFRA_HOME/bin/ldifwrite
-c production_oid_net_service_name -b "orclcommongroupsearchbase"
-f /tmp/group.ldif
```

In the syntax, *orclcommongroupsearchbase* is the `orclcommongroupsearchbase` subtree written out in LDIF format, such as `cn=Groups,dc=us, dc=company,dc=com`.

When prompted for the directory password, enter the password of the `ODS` user. This password defaults to the password assigned for administrator `ias_admin` during installation. You should run the `ldifwrite` command once for each group search base value (`orclcommongroupsearchbase`) from the Oracle Context settings query.

e. Save the Delegated Administration Services administrative groups subtree from the production directory server:

```
PRODUCTION_INFRA_HOME/bin/ldifwrite
-c production_oid_net_service_name
-b "cn=groups, cn=OracleContext, orcldefaultsubscriber"
-f /tmp/dasAdminGroups.ldif
```

In the syntax, *orcldefaultsubscriber* is the `orcldefaultsubscriber` value returned in Step 2a of this procedure.

f. Copy the password policy configuration entry, and append the entry with the following `ldapsearch` commands:

```
PRODUCTION_INFRA_HOME/bin/ldapsearch -L
-h production_oid_host -p production_oid_port
-D cn=orcladmin -w production_orcladmin_passwd
-b "cn=PwdPolicyEntry, cn=common, cn=products, cn=OracleContext"
-s base "objectclass=*" >> /tmp/pwdPolicy.ldif

PRODUCTION_INFRA_HOME/bin/ldapsearch -L
-h production_oid_host -p production_oid_port
-D cn=orcladmin -w production_orcladmin_passwd
-b "cn=PwdPolicyEntry, cn=common, cn=products, cn=OracleContext,
orcldefaultsubscriber"
-s base "objectclass=*" >> /tmp/pwdPolicy.ldif
```

3. Prepare the product Metadata Repository in the production environment for export:

   a. List the schemas to be exported:

      It is necessary to identify all of the schemas that need to be exported, including the PORTAL and related schemas, the DISCOVERER5 schema, and any schemas used for database providers or OracleAS Portal Portlet Builder components.

      To list all the schemas, run the following query from SQL*Plus in the PORTAL schema:

```
SET  LINESIZE 132

SELECT USERNAME, DEFAULT_TABLESPACE, TEMPORARY_TABLESPACE FROM DBA_USERS
WHERE USERNAME IN (user, user||'_PUBLIC', user||'_DEMO', user||'_APP',
'DISCOVERER5')
OR USERNAME IN (SELECT DISTINCT OWNER
FROM WWAPP_APPLICATION$
WHERE NAME != 'WWV_SYSTEM')
ORDER BY USERNAME;
```

      The response looks similar to the following output:

```
USERNAME                       DEFAULT_TABLESPACE      TEMPORARY_TABLESPACE
------------------------------------------------------------------------------
ARUN                           USERS                        TEMP
DISCOVERER5                    DISCO_PTM5_META              DISCO_PTM5_TEMP
PORTAL                         PORTAL                       PORTAL_TMP
PORTAL_APP                     PORTAL                       PORTAL_TMP
PORTAL_DEMO                    PORTAL                       PORTAL_TMP
PORTAL_PUBLIC                  PORTAL                       PORTAL_TMP
SCOTT                          SYSTEM                       TEMP
TESTER_DAT                     SYSTEM                       TEMP
TESTER_SCH                     SYSTEM                       TEMP
UPG_SCHEMA                     USERS                        TEMP
UPG_SCHEMDB                    USERS                        TEMP
10 rows selected
```

      This command will only list schemas that are directly related to database providers or Portlet Builder components registered in the OracleAS Portal. If any of these schemas additionally reference objects in other schemas, then add them to the list of schemas to be exported.

   b. List the tablespaces used.

It is necessary to ensure that the tablespaces on the target databases match the ones used in the source. To list the tablespaces used in the source database, run the following query from SQL*Plus as the PORTAL user:

```
SQL> SELECT DISTINCT TABLESPACE_NAME FROM DBA_SEGMENTS WHERE OWNER IN
(schema_list)
UNION
SELECT DISTINCT DEFAULT_TABLESPACE FROM DBA_USERS WHERE USERNAME IN
(schema_list)
/
SELECT DISTINCT TEMPORARY_TABLESPACE FROM DBA_USERS WHERE USERNAME IN
(schema_list)
/
```

where *schema_list* contains the schemas generated from Step 3a of this procedure.

The query and response looks similar to the following output:

```
SQL> SELECT DISTINCT TABLESPACE_NAME FROM DBA_SEGMENTS WHERE OWNER IN
  2  ('ARUN','PORTAL','DISCOVERER5','PORTAL_APP','PORTAL_DEMO','PORTAL_
PUBLIC','SCOTT',
  3  'TESTER_DAT','TESTER_SCH','UPG_SCHEMA','UPG_
SCHEMDB','FLIGHTS','PROV9022')
  4  UNION
  5  SELECT DISTINCT DEFAULT_TABLESPACE FROM DBA_USERS WHERE USERNAME IN
  6  ('ARUN','PORTAL', 'DISCOVERER5','PORTAL_APP','PORTAL_DEMO','PORTAL_
PUBLIC','SCOTT',
  7  'TESTER_DAT','TESTER_SCH','UPG_SCHEMA','UPG_
SCHEMDB','FLIGHTS','PROV9022');

TABLESPACE_NAME
------------------------------
INDX
DISCO_PTM5_CACHE
DISCO_PTM5_META
PORTAL
PORTAL_DOC
PORTAL_IDX
PORTAL_LOG
SYSTEM
USERS

7 rows selected.

SQL> SELECT DISTINCT TEMPORARY_TABLESPACE FROM DBA_USERS WHERE USERNAME IN
('ARUN','PORTAL','PORTAL_APP','PORTAL_DEMO','PORTAL_
PUBLIC','SCOTT','TESTER_DAT','TESTER_SCH','UPG_SCHEMA','UPG_
SCHEMDB','FLIGHTS','PROV9022');

TEMPORARY_TABLESPACE
------------------------------
DISCO_PTM5_TEMP
PORTAL_TMP
TEMP
```

c. List schemas that have snapshots created in them.

It is necessary to identify the schemas that have snapshots created in them, run the following query from SQL*Plus as the SYS user:

```
SQL> SELECT OWNER, NAME FROM DBA_SNAPSHOTS WHERE OWNER IN (schema_list);
```

where *schema_list* contains the schemas generated from Step 3a of this procedure.

The query and response looks similar to the following output:

```
SQL> SELECT OWNER, NAME FROM DBA_SNAPSHOTS
  2 WHERE OWNER IN ('ARUN','DISCOVERER5','PORTAL','PORTAL_APP','PORTAL_
DEMO','PORTAL_PUBLIC','SCOTT','TESTER_DAT','TESTER_SCH','UPG_SCHEMA','UPG_
SCHEMDB');
OWNER                          NAME
------------------------------ ------------------------------
PORTAL_DEMO                    EMP_SNAPSHOT
SCOTT                          EMP_SNAPSHOT
```

d. Determine the character set of the production database with the following query from SQL*Plus as the SYS user:

```
SQL> SELECT userenv('language') from dual;
```

The output looks similar to the following:

```
USERENV('LANGUAGE')
----------------------------------------------------
AMERICAN_AMERICA.WE8MSWIN1252
```

Set the NLS_LANG environment variable before performing the database export:

```
setenv NLS_LANG AMERICAN_AMERICA.WE8MSWIN1252
```

e. Prior to exporting data, run the CATEXP.SQL script from the *PRODUCTION_INFRA_HOME*/rdbms/admin directory with SYSDBA privileges:

```
SQL> CONNECT SYS/password AS SYSDBA
@catexp.sql
```

This script creates the data dictionary and public synonyms for many of its views.

4. Export data from the product Metadata Repository in the production environment:

a. Stop all processes managed by OPMN:

```
PRODUCTION_INFRA_HOME/opmn/bin/opmnctl stopall
```

b. Run the Export utility:

The actual export is done with the database exp command as follows, with the example showing just the four core OracleAS Portal schemas. Include any other schemas identified in the SELECT statement from Step 3a of this procedure as well.

```
PRODUCTION_INFRA_HOME/bin/exp 'sys/password@instance AS SYSDBA'
file=/tmp/portal_exp.dmp grants=y statistics=none log=/tmp/portal_exp.log
owner=portal,portal_app,portal_demo,portal_public
```

For UNIX operating systems, you must precede the quotation marks with an escape character, such as the backslash (\).

c. Start all the processes managed by OPMN:

```
PRODUCTION_INFRA_HOME/opmn/bin/opmnctl startall
```

5. Export OracleAS Single Sign-On External Applications data from the production Identity Management.

   This step exports any external applications and password store data. Extract the OracleAS Single Sign-On data with the `ssomig` utility as follows:

   ```
   PRODUCTION_INFRA_HOME/sso/bin/ssomig -s orasso -p orasso_password
   -c production_sso_net_service_name -log_d /tmp -export -log_f ssomig.log
   -d ssomig.dmp
   ```

   The response looks similar to the following output:

   ```
   SSO Migration Tool: Release 10.1.2.0.2 - Production on Fri Feb 25 16:15:19 2005
   Copyright (c) 2002-2005 Oracle. All rights reserved.

   Verifying SSO schema information...

   Data export started at Fri Feb 25 16:15:22 2005

   Log Directory              : /tmp
   Log File Name              : /tmp/ssomig.log
   Export File Name           : /tmp/ssomig.dmp
   SSO Schema Name            : orasso
   SSO Schema Password        : ******
   Connect String             : asdbupg


   Copying data...Done
   Exporting data...Done
   Creating configuration file...Done

   Exported Dump File: /tmp/ssomig.dmp
   Configuration File: /tmp/ssoconf.log

   Data Export Complete
   ```

6. If your production environment includes OracleBI Discoverer, export EUL data:

   For each EUL in the customer database, create a copy using the following command from the production environment:

   ```
   PRODUCTION_MIDTIER_HOME/bin/eulapi -
   connect old_eul_owner/password@database_service_name
   -export /tmp/file1.eex -all -log /tmp/exp.log
   ```

   If you intend to copy data to another customer database in the test environment, use the service name of the test database for *`database_service_name`*.

   Note: The `eulapi` utility does not exist on Windows in releases 9.0.4.*x* and 10.1.2.0.*x*. For environments supporting these releases, download the patches for this utility. The related Automated Release Updates (ARU) number for 9.0.4.*x* is ARU 7462620 and 10.1.2.0.*x* is ARU 7462623.

   You can download these patches from Oracle*Metalink*:

   http://metalink.oracle.com

Task 1: Install Test Identity Management

Install and set up the test Identity Management and its associated Metadata Repository. You can install these OracleAS Infrastructure components on the same computer or on separate computers.

To install the test Identity Management:

1. Identify the character set in use on the database of the production Identity Management.

   Run the following query from SQL*Plus as the `SYS` user:

   ```
   SQL> SELECT userenv('language') from dual;
   ```

   The output looks similar to the following:

   ```
   USERENV('LANGUAGE')
   ----------------------------------------------------
   AMERICAN_AMERICA.WE8MSWIN1252
   ```

2. List the ports used on the production database (optional) in order to use the same port numbers on the copy.
3. Install Oracle Application Server using Oracle Universal Installer.
4. From the Select a Product to Install screen, select **OracleAS Infrastructure**.
5. From the Select Installation Type screen, select the appropriate option for how you want the Identity Management pieces installed:
   - To install test Identity Management and its associated Metadata Repository on the same computer, select **Identity Management and OracleAS Metadata Repository**.
   - To install test Identity Management and its associated Metadata Repository on separate computers, first select the **OracleAS Metadata Repository** option for one of the computers, and then select the **OracleAS Identity Management** option for the other computer.
6. When you are prompted with the **Specify Namespace in Internet Directory** screen, enter the value of the `orcldefaultsubscriber` obtained from the query of the production Oracle Internet Directory server in Step 2a of "Prerequisite Export Steps" on page 11-3.
7. Create the database using the same character set as the production instance that you are copying to avoid any character set conversion issues. You do not have to specify the same port numbers, but it can help simplify the configuration.
8. Perform a backup of the test Identity Management configuration files and Metadata Repository with the OracleAS Backup and Recovery Tool. See Chapter 20.

Task 2: Set Up Test Product Metadata Repository

To install and set up the test product Metadata Repository:

1. Identify the character set in use on the production database, as described in Step 3d of "Prerequisite Export Steps" on page 11-3.
2. Optionally, list the ports used on the production database in order to use the same port numbers on the copy.
3. Install OracleAS Infrastructure and Metadata Repository.
   a. Install Oracle Application Server using Oracle Universal Installer.

b. From the Select a Product to Install screen, select **OracleAS Infrastructure**.

c. From the Select Installation Type screen, select **Metadata Repository**.

d. Create the database using the same character set as the production instance that you are copying to avoid any character set conversion issues. Also, specify the same ports to use, if you want more fidelity. This should not affect the upgrade testing though, even if using different port numbers in the copy.

4. Perform a backup of the Metadata Repository database installed with the infrastructure to make is easier to do import iterations in case of upgrade errors. See Chapter 20.

Task 3: Install the Test Middle Tier

To install the test middle-tier instances and configure them to use the test Identity Management according to what you want to test:

1. Install Oracle Application Server using Oracle Universal Installer.
2. From the Select a Product to Install screen, choose the **Portal and Wireless** middle for environments with OracleAS Portal or the **Business Intelligence and Forms** for environments with OracleBI Discoverer.
3. Apply any required patches to bring the test system up to the same patch-level release as the production system.

   For patches associated with a release, use the Oracle Universal Installer on the production system to determine applied patches. For patches not associated with a release, use the following command to determine applied patches:

   ```
   ORACLE_HOME/OPatch/opatch lsinventory -detail
   ```

4. Perform a backup of the test middle tier, product Metadata Repository, Identity Management configuration files, and Identity Management Metadata Repository with the OracleAS Backup and Recovery Tool. See Chapter 20.

For 10.1.2.0.*x* environments in which both the production and test environments are running on the same operating system, you can either perform a middle-tier installation or clone the production middle-tier instance. To clone the production middle-tier instance, perform tasks in procedure Section 10.4, "Cloning Oracle Application Server Instances" on page 10-5.

Task 4: Copy Data from Production Identity Management to the Test Environment

To copy Identity Management data to the test environment:

1. Prior to importing users and groups to the test directory server, use the `bulkdelete` command to remove the existing default users, groups containers, and Delegated Administration Services administrative groups subtree from the test directory server.

   a. Run the following `ldapsearch` commands to obtain the values:

   ```
   TEST_INFRA_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
   -b "cn=users, orcldefaultsubscriber"
   -s base "objectclass=*"

   TEST_INFRA_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
   -b "cn=groups, orcldefaultsubscriber"
   -s base "objectclass=*"

   TEST_INFRA_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
   -b "cn=groups, cn-OracleContext,orcldefaultsubscriber"
   ```

```
-s base "objectclass=*"
```

In the syntax, *orcldefaultsubscriber* is the orcldefaultsubscriber value returned from Step 2a of "Prerequisite Export Steps" on page 11-3.

b. Stop the directory server with the following command:

```
TEST_INFRA_HOME/opmn/bin/opmnctl stopproc ias-component=OID
```

c. Run the following bulkdelete commands:

```
TEST_INFRA_HOME/ldap/bin/bulkdelete.sh
-connect test_oid_net_service_name
-base "cn=users, orcldefaultsubscriber"

TEST_INFRA_HOME/ldap/bin/bulkdelete.sh
-connect test_oid_net_service_name
-base "cn=groups, orcldefaultsubscriber"

TEST_INFRA_HOME/ldap/bin/bulkdelete.sh
-connect test_oid_net_service_name
-base "cn=groups, cn=OracleContext, orcldefaultsubscriber"
```

In the syntax, *test_oid_net_service_name* specifies the net service name for the test directory, as defined in the tnsnames.ora file.

You will be prompted to provide the password to the ODS schema to complete these commands.

d. Run the same ldapsearch commands in Step 1a of this procedure to ensure the values are removed.

2. Use the bulkload utility to load users from the user.ldif file you created in Step 2c of "Prerequisite Export Steps" on page 11-3:

a. Start the directory server with the following command:

```
TEST_INFRA_HOME/opmn/bin/opmnctl startproc ias-component=OID
```

b. To run the bulkload utility, set the directory server mode to read/modify:

From Oracle Directory Manager, navigate to the **server** entry (the main node under the **Oracle Internet Directory Servers**), and change the **Server Mode** attribute from **Read/Write** to **Read/Modify** from the drop-down list.

If you prefer to use the LDAP command line utilities, use the ldapmodify command:

```
ldapmodify -h test_oid_host -p test_oid_port -D cn=orcladmin
-w test_orcladmin_pwd -v -f rm.ldif
```

where rm.ldif is a file you create, with the following contents:

```
dn:
changetype: modify
replace: orclservermode
orclservermode: rm
```

c. After changing the server mode, you need to stop the Oracle Internet Directory processes:

```
TEST_INFRA_HOME/opmn/bin/opmnctl stopproc ias-component=OID
```

d. If the production Identity Management has been upgraded from release 9.0.2.*x* to 9.0.4.*x*, remove the entries starting with `orclactivestartdate` from the `user.ldif` file. If these entries are not removed, the `bulkload` utility will fail.

e. Load users into the test Oracle Internet Directory by using the `bulkload` utility to load the LDIF file generated from the production system. You created this file in Step 2c of "Prerequisite Export Steps" on page 11-3. When invoking the `bulkload` utility, be sure to specify the absolute path of the LDIF file, even if it is in the current directory.

```
TEST_INFRA_HOME/ldap/bin/bulkload.sh
-connect test_oid_net_service_name -check -generate -restore
-load -append /tmp/user.ldif
```

When invoking the `bulkload` utility, be sure to specify the absolute path of the LDIF file, even if it is in the current directory.

The response looks similar to the following output:

```
Verifying node "orcl"
-----------------------------
This tool can only be executed if you know database user password
for OiD on orcl
Enter OiD password ::
```

f. Provide the password for the schema used by Oracle Internet Directory. This defaults to the password assigned for the `ias_admin` administrator during installation.

This command loads all the users, provided there is no error reported in the check mode on the exported LDIF file.

g. Start the directory server with the following command:

```
TEST_INFRA_HOME/opmn/bin/opmnctl startproc ias-component=OID
```

3. Move the `pwdPolicy.ldif` file you created in Step 2f of "Prerequisite Export Steps" on page 11-3 to the test environment and stop the directory server with the following commands:

```
TEST_INFRA_HOME/bin/ldapaddmt -h test_oid_host
-p test_oid_port -D cn=orcladmin -w test_orcladmin_passwd
-v -f /tmp/pwdPolicy.ldif
TEST_INFRA_HOME/opmn/bin/opmnctl stopproc ias-component=OID
```

Ensure the password policy in the default subscriber matches that of the production environment. Perform the same `ldapsearch` command you performed in Step 2f of "Prerequisite Export Steps" on page 11-3 in the test environment.

4. Use the `bulkload` utility to load groups from the `group.ldif` file you created in Step 2d of "Prerequisite Export Steps" on page 11-3.

```
TEST_INFRA_HOME/ldap/bin/bulkload.sh -connect test_oid_net_service_name
-check -generate -restore -load -append /tmp/group.ldif
```

When invoking the `bulkload` utility, be sure to specify the absolute path of the LDIF file, even if it is in the current directory.

This command loads all the groups, provided there is no error reported in the check mode on the exported LDIF file.

5. Load the administrative groups subtree saved from production in Step 2e of "Prerequisite Export Steps" on page 11-3 into the test environment.

```
TEST_INFRA_HOME/ldap/bin/bulkload.sh -connect test_oid_net_service_name
-check -generate -restore -load -append /tmp/dasAdminGroups.ldif
```

Perform the same `ldifwrite` command you performed in Step 2e of "Prerequisite Export Steps" on page 11-3 in the test environment to ensure the value has been correctly updated.

6. Start the test directory server and other Identity Management components:

```
cd TEST_INFRA_HOME/opmn/bin
opmnctl startall
opmnctl: starting opmn and all managed processes...
```

After performing the bulkload commands, restore the server mode of the directory server to the normal setting:

```
ldapmodify -h test_oid_host -p test_oid_port -D cn=orcladmin
-w test_orcladmin_pwd -v -f rw.ldif
```

where `rw.ldif` is a file you create, with the following contents:

```
dn:
changetype: modify
replace: orclservermode
orclservermode: rw
```

If you use Oracle Directory Manager instead, change the **Server Mode** attribute of the server entry back to **Read/Write**

7. On the test directory server, configure Oracle Context parameters:

   a. In Step 2b of "Prerequisite Export Steps" on page 11-3, you created an `/tmp/OracleContext.ldif` file containing the configured user and group search bases. Edit this file as follows:

   - Use the respective output values for *`value`* returned in Step 2b.

   - `OrclCommonUserSearchBase` might contain only one value.

```
dn: value
Changetype: modify
replace: orclCommonUserSearchBase
OrclCommonUserSearchBase: value1
OrclCommonUserSearchBase: value2
-
replace: orclCommonGroupSearchBase
OrclCommonGroupSearchBase: value1
OrclCommonGroupSearchBase: value2
-
replace: orclCommonNickNameattribute
orclCommonNickNameattribute: value
```

   b. Configure the user and group search base on the test directory server:

```
TEST_INFRA_HOME/bin/ldapmodify -D cn=orcladmin
-w orcladmin_password -h test_oid_host -p test_oid_port
-v -f /tmp/OracleContext.ldif
```

Perform the same `ldapsearch` command you performed in Step 2b of "Prerequisite Export Steps" on page 11-3 in the test environment to ensure the value has been correctly updated.

8. Change the value of the `orclcommonnicknameattribute`:

   a. Run the `ldapsearch` command to see the test nickname attribute:

   ```
   TEST_INFRA_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
   -D cn=orcladmin -w test_orcladmin_passwd
   -b cn=common, cn=products, cn=OracleContext, orcldefaultsubscriber"
   -s base "objectclass=*" orclcommonnicknameattribute
   ```

   The response returns the `orclcommonnicknameattribute` value:

   ```
   orclcommonnicknameattribute: uid
   ```

   b. Compare this value with the production nickname attribute, obtained in Step 2b of "Prerequisite Export Steps" on page 11-3.

   c. For instances which have been upgraded from 9.0.2.*x* to 9.0.4.*x*, the production nickname attribute may not be set to `orclcommonnicknameattribute=cn`. For these instances, you need to update the test directory server with this value.

   You can navigate to that entry from Oracle Directory Manager and change its value.

   If you prefer to use the LDAP command line utilities, use the `ldapmodify` command:

   ```
   ldapmodify -h test_oid_host -p test_oid_port -D cn=orcladmin
   -w test_orcladmin_pwd -f nickname.ldif
   ```

   where `nickname.ldif` is a file you create, with the following contents:

   ```
   dn: cn=Common,cn=Products,cn=OracleContext,dc=us,dc=company,dc=com
   changetype: modify
   replace: orclcommonnicknameattribute
   orclcommonnicknameattribute: cn
   ```

   d. Perform Step 8a of this procedure to ensure the value of `orclcommonnicknameattribute` has been correctly updated.

9. Import OracleAS Single Sign-On External Applications data.

   This step imports any external applications and password store data that was exported in Step 5 of procedure "Prerequisite Export Steps" on page 11-3. Extract the OracleAS Single Sign-On data with the `ssomig` utility in `-import` mode as follows:

   ```
   TEST_INFRA_HOME/sso/bin/ssomig -import -overwrite -s orasso
   -p orasso_pwd -c test_sso_net_service_name -d exp_dumpfile_name
   -log_d dumpfile_dir -discoforce
   ```

   Ensure the *`exp_dumpfile_name`* is filename of the dump file and is in the directory *`dumpfile_dir`* directory. The *`dumpfile_dir`* directory will also contain the `ssoconf.log` file, which was generated when you ran the `ssomig` utility in `-export` mode.

   You obtain the password for the `ORASSO` schema (`orasso_pwd`) from Oracle Internet Directory as follows:

   ```
   TEST_INFRA_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
   ```

```
-D cn=orcladmin -w test_orcladmin_pwd
-b "orclreferencename=test_oid_service_name,cn=IAS Infrastructure
Databases,cn=ias,cn=products,cn=oraclecontext"
-s sub "(orclresourcename=orasso)" orclpasswordattribute
```

The response for `ssomig` looks similar to the following output:

```
SSO Migration Tool: Release 10.1.2.0.2 - Production on Sun Feb 27 12:10:27 2005
Copyright (c) 2002-2005 Oracle. All rights reserved.

Verifying SSO schema information...

Data import started at Sun Feb 27 12:10:28 2005

Log Directory              : /tmp
Log File Name              : /tmp/ssomig.log
Import File Name           : ssomig.dmp
SSO Schema Name            : orasso
SSO Schema Password        : ******
Connect String             : orcl
Mode                       : overwrite

Loading data into the SSO schema...Done

Data import completed.
```

See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for more information

10. Perform a backup of the test Identity Management configuration files and Metadata Repository with the OracleAS Backup and Recovery Tool. See Chapter 20.

Task 5: Copy Data from the Production Product Metadata Repository to the Test Environment

To copy data from the product Metadata Repository in the production environment to the test environment:

1. Drop the OracleAS Portal schemas from the product Metadata Repository in the test environment:

   a. Take note of the schema passwords currently assigned to the schemas to be dropped. You will be re-creating these schemas later and you should provide the same password when you create them.

      You can query the currently assigned passwords from the newly installed test Oracle Internet Directory:

      ```
      TEST_INFRA_HOME/bin/ldapsearch -p test_oid_port -h test_oid_host -D
      cn=orcladmin -w test_orcladmin_password
      -b "orclReferenceName=portal_service_name,cn=IAS Infrastructure
      Databases,cn=IAS,cn=Products,cn=OracleContext" -s
      sub "(|(orclResourceName=PORTAL*)(orclResourceName=DISCOVERER5))"
       orclpasswordattribute
      ```

      Take note of the passwords listed for the following schemas: `DISCOVERER5`, `PORTAL`, `PORTAL_PUBLIC`, `PORTAL_APP`, and `PORTAL_DEMO`.

      The response looks similar to the following output:

      ```
      OrclResourceName=PORTAL,orclReferenceName=portal.us.company.com,cn=IAS
      ```

```
Infrastructure Databases,cn=IAS,cn=Products,cn=OracleContext
orclpasswordattribute=DTFDKD58

OrclResourceName=PORTAL_
PUBLIC,orclReferenceName=portal.us.company.com,cn=IAS Infrastructure
Databases,cn=IAS,cn=Products,cn=OracleContext
orclpasswordattribute=X9g57NMP

OrclResourceName=PORTAL_DEMO,orclReferenceName=portal.us.company.com,cn=IAS
Infrastructure Databases,cn=IAS,cn=Products,cn=OracleContext
orclpasswordattribute=ZyKR9805

OrclResourceName=PORTAL_APP,orclReferenceName=portal.us.company.com,cn=IAS
Infrastructure Databases,cn=IAS,cn=Products,cn=OracleContext
orclpasswordattribute=IGqvwL9c

OrclResourceName=DISCOVERER5,orclReferenceName=portal.us.company.com,cn=IAS
Infrastructure Databases,cn=IAS,cn=Products,cn=OracleContext
orclpasswordattribute=Alu23B8w
```

b. You need to stop all SQL*Plus connections to the `PORTAL` schema before dropping it, or else the attempt to drop it will throw an `ORA-01940: cannot drop a user that is currently connected`.

c. Drop the OracleAS Portal-related schemas from the test product Metadata Repository just installed.

From SQL*Plus as the `SYS` user, drop the OracleAS Portal schemas from the product Metadata Repository:

```
SQL> DROP USER portal_public cascade;

User dropped.

SQL> DROP USER portal_app cascade;

User dropped.

SQL> DROP USER portal_demo cascade;

User dropped.

SQL> DROP USER portal cascade;

User dropped.

SQL> DROP USER discoverer5 cascade;

User dropped.
```

2. Prepare the product Metadata Repository in the test environment for import:

a. Create or alter tablespaces in the test database.

Check that the required tablespaces exist in the test database. The tablespaces in the test database must at least include all the ones listed from the production instance. To list all the tablespaces on the target, run the following from SQL*Plus as the `SYS` user:

```
SQL> SET PAGESIZE 65
SELECT TABLESPACE_NAME FROM DBA_TABLESPACES ORDER BY TABLESPACE_NAME;
```

The response looks similar to the following output

```
TABLESPACE_NAME
------------------------------
B2B_DT
B2B_IDX
B2B_LOB
B2B_RT
DCM
DISCO_PTM5_CACHE
DISCO_PTM5_META
DSGATEWAY_TAB
IAS_META
OCATS
OLTS_ATTRSTORE
OLTS_BATTRSTORE
OLTS_CT_STORE
OLTS_DEFAULT
OLTS_SVRMGSTORE
PORTAL
PORTAL_DOC
PORTAL_IDX
PORTAL_LOG
SYSAUX
SYSTEM
TEMP
UDDISYS_TS
UNDOTBS1
USERS
WCRSYS_TS

26 rows selected.
```

Comparing this example to the list of tablespaces obtained from the production instance, we see that we need to create the `INDX` (permanent), `PORTAL_TMP` (temporary) and `DISCO_PTM5_TEMP` (temporary) tablespaces.

To create a new tablespace, use the `CREATE TABLESPACE` or `CREATE TEMPORARY TABLESPACE` commands in SQL*Plus, in the `SYS` schema. For example:

```
SQL> CREATE TABLESPACE INDX
  2  DATAFILE '/u01/app/oracle/product/oradata/orcl/indx.dbf' SIZE 20M
AUTOEXTEND ON
  3  DEFAULT STORAGE (INITIAL 1M NEXT 2M MINEXTENTS 2);

Tablespace created.

SQL> CREATE TEMPORARY TABLESPACE PORTAL_TMP
  2  TEMPFILE '/u01/app/oracle/product/oradata/orcl/portal_tmp.dbf' SIZE
20M
  3  AUTOEXTEND ON;

Tablespace created.

SQL> CREATE TEMPORARY TABLESPACE DISCO_PTM5_TEMP
  2  TEMPFILE '/u01/app/oracle/product/oradata/asdb/disco_ptm5_temp.dbf'
SIZE 20M
  3  AUTOEXTEND ON;

Tablespace created.
```

For any tablespaces that already exist in the target database, it is recommended that they be set to autoextend or they must be sized large enough to hold the imported PORTAL schemas. You can use the following script to enable autoextend on all datafiles:

```
SET DEFINE OFF
SET HEAD OFF
SET LINES 4000
SPOOL DATAFILES.SQL
SELECT 'ALTER DATABASE DATAFILE '''||FILE_NAME||''' AUTOEXTEND ON;'
FROM DBA_DATA_FILES ;
SPOOL OFF
```

At this point, you can edit out any extraneous lines captured in the spool file DATAFILES.SQL, and then run it in the SYS schema to alter the data files:

```
@DATAFILES.SQL
```

b. From the production middle-tier instance, run SQL*Plus connect to the test Metadata Repository from SQL*Plus as the SYS user with SYSDBA privileges, and re-create the PORTAL schema by running the wbisys.sql script from the *PRODUCTION_MIDTIER_HOME*/portal/admin/plsql/wwv directory.

```
SQL> CONNECT SYS/password AS SYSDBA@test_metadata_repository_net_service_
name
SQL> @wdbisys.sql PORTAL portal_default_tablespace portal_temporary_
tablespace wdbisys.log
```

In the syntax, *test_metadata_repository_net_service_name* specifies the net service name for the product Metadata Repository in the test environment, as defined in the tnsnames.ora file.

The response looks similar to the following output:

```
SQL> @wdbisys.sql PORTAL PORTAL PORTAL_TMP wdbisys.log
B E G I N   S Y S   I N S T A L L

...start: Saturday  26 February , 2005 12:01:23

I.    CREATE USER PORTAL and GRANT PRIVS
old   1: create user &&1 identified by &&1
new   1: create user PORTAL identified by PORTAL

User created.
…
PL/SQL procedure successfully completed.

...end of SYS install
Disconnected from Oracle Database 10g Enterprise Edition Release 10.1.0.3.1
- Production
With the Partitioning, OLAP and Data Mining options
```

c. Change the PORTAL password from SQL*Plus as the SYS user:

```
SQL> ALTER USER PORTAL IDENTIFIED BY password_from_test_oid;
```

This command creates the PORTAL schema and grants all of the necessary privileges.

For example:

```
SQL> ALTER USER PORTAL IDENTIFIED BY DTFDKD58;
User altered
```

d. Create the PORTAL_PUBLIC schema.

Change to the *PRODUCTION_MIDTIER_HOME*/portal/admin/plsql/wws directory and run the following script from SQL*Plus as the SYS user:

```
SQL> CONNECT SYS/password AS SYSDBA@test_metadata_repository_net_service_
name
@cruser.sql PORTAL PORTAL portal_default_tablespace portal_temporary_
tablespace
```

The response looks similar to the following output:

```
SQL> @cruser.sql PORTAL PORTAL PORTAL PORTAL_TMP
old   1: select ('&&1'||'_public') t1 from dual
new   1: select ('PORTAL'||'_public') t1 from dual
...
User altered.

No errors.
Disconnected from Oracle Database 10g Enterprise Edition Release 10.1.0.3.1
- Production
With the Partitioning, OLAP and Data Mining options

SQL*Plus: Release 10.1.0.3.0 - Production on Tue Mar 1 08:28:17 2005

Copyright (c) 1982, 2005, Oracle.  All rights reserved.

Connected to:
Oracle Database 10g Enterprise Edition Release 10.1.0.3.1 - Production
With the Partitioning, OLAP and Data Mining options

SQL> ALTER USER PORTAL_PUBLIC IDENTIFIED BY password_from_oid;
User altered.
```

e. Change the PORTAL_PUBLIC password from SQL*Plus as the SYS user:

For example:

```
SQL> ALTER USER PORTAL_PUBLIC IDENTIFIED BY X9g57NMP;
User altered.
```

f. Create the auxiliary schemas from SQL*Plus as the SYS user.

Check the list of schemas that will be imported from Step 3a of "Prerequisite Export Steps" on page 11-3. If the schemas already exist in the test database, then drop them. Before dropping any schemas, ensure that those schemas are not in use by other applications. To create the new schemas, use the following syntax:

```
SQL> GRANT CONNECT, RESOURCE TO schema IDENTIFIED BY password;
ALTER USER username DEFAULT TABLESPACE default_tablespace TEMPORARY
TABLESPACE temporary_tablespace;
```

You must create a schema for each schema in the list from Step 3a. Use the ALTER USER command to adjust any user properties as necessary. For instance, the default and temporary tablespaces should be set to the ones specified by the results from the query in Step 3a.

When creating the PORTAL_APP, PORTAL_DEMO, or DISCOVERER5 schemas, use the passwords you extracted from Oracle Internet Directory.

Following is an example for the standard schemas:

```
GRANT CONNECT,RESOURCE TO portal_app IDENTIFIED BY IGqvwL9c;
ALTER USER portal_app default tablespace PORTAL temporary tablespace
PORTAL_TMP;
GRANT CONNECT,RESOURCE TO portal_demo IDENTIFIED BY ZyKR9805;
ALTER USER portal_demo default tablespace PORTAL temporary tablespace
PORTAL_TMP;
GRANT CONNECT,RESOURCE TO discoverer5 IDENTIFIED BY AHb10z3b;
ALTER USER discoverer5 default tablespace PORTAL temporary tablespace
PORTAL_TMP;
```

g. If Step 3c "Prerequisite Export Steps" on page 11-3 returned any schemas that had snapshots in them, grant privilege to define snapshots to those schemas from SQL*Plus as the SYS user:

```
SQL> GRANT CREATE SNAPSHOT TO schema;
```

where *schema* contains the schemas returned by performing Step 3c. If the schemas returned are PORTAL_DEMO and SCOTT, then you would issue the following SQL commands:

```
SQL> GRANT CREATE SNAPSHOT TO PORTAL_DEMO;
SQL> GRANT CREATE SNAPSHOT TO SCOTT;
```

h. For a Standard Edition database only, prepare the database for import:

From SQL*Plus, initialize the portal login trigger for import as the PORTAL user:

```
cd TEST_INFRA_HOME/portal/admin/plsql/wwhost
sqlplus portal/password@portal_net_service_name
SQL> @insttrig.sql PORTAL
```

i. Ensure the character set of the product Metadata Repository in the production environment matches the character set of the product Metadata Repository in the production environment.

The product Metadata Repository in the test environment must match the character set of the data in the import dump produced in the production environment.

Set the NLS_LANG environment variable on the test Metadata Repository to match the character set established in Step 3d of "Prerequisite Export Steps" on page 11-3 in production environment:

```
setenv NLS_LANG AMERICAN_AMERICA.WE8MSWIN1252
```

j. Run the catexp.sql script from the *TEST_INFRA_HOME*/rdbms/admin directory with SYSDBA privileges:

```
SQL> CONNECT SYS/password AS SYSDBA
@catexp.sql
```

This script creates the data dictionary and public synonyms for many of its views.

3. Import schemas into the product Metadata Repository in the test environment:

a. Run the Import utility.

Make sure that the database version that you are importing into is the same or a later version than the database you exported from. The actual import is done with the database `imp` command as follows, with the example showing just the four core OracleAS Portal schemas. Include any other schemas identified in the `SELECT` statement from Step 3a of "Prerequisite Export Steps" on page 11-3.

```
TEST_INFRA_HOME/bin/imp 'sys/password@instance AS SYSDBA'
file=/tmp/portal_exp.dmp grants=y log=/tmp/portal_imp.log
fromuser=portal,portal_app,portal_demo,portal_public touser=portal,portal_
app,portal_demo,portal_public
```

For UNIX operating systems, you must precede the quotation marks with an escape character, such as the backslash.

Enter the list of schemas as a comma-separated list.

You can ignore the following errors in the import log:

```
IMP-00015: following statement failed because the object already exists:
IMP-00041: Warning: object created with compilation warnings.
IMP-00061: Warning: Object type "SYS"."ODCIPARTINFO" already exists with a
different identifier
IMP-00092: Java object "SCOTT"."DeleteDbc" already exists, cannot be
created
```

If you get other errors, such as missing tablespaces or missing schemas, you need to restore your database from your offline backup and repeat the import step after resolving the issue. Resolve this issue as indicated by the error message. For example, if a referenced schema was missing, then include that schema in the export and re-export the set of schemas from the production server. The schema needs to be created in the test system prior to import, as described in Step 3f of this procedure. This may typically happen for schemas that are not automatically identified by the `SELECT` statement in Step 3a of "Prerequisite Export Steps" on page 11-3.

b. Resolve any job conflict errors in the import log.

The following example shows an example of a job conflict in the import log:

```
IMP-00017: following statement failed with ORACLE error 1:
"BEGIN DBMS_JOB.ISUBMIT(JOB=>15,WHAT=>'begin execute immediate"
"''begin wwctx_sso.cleanup_sessions( p_hours_old => 168 ); e"
"nd;'' ; exception when others then null;
end;',NEXT_DATE=>"
"TO_DATE('2005-03-23:14:11:08','YYYY-MM-DD:HH24:MI:SS'),INTERVAL=>'SYSDATE
+"
" 24/24',NO_PARSE=>TRUE); END;"
IMP-00003: ORACLE error 1 encountered
ORA-00001: unique constraint (SYS.I_JOB_JOB) violated
ORA-06512: at "SYS.DBMS_JOB", line 97
ORA-06512: at line 1
```

If such an error is seen during the import, then run the following from SQL*Plus as the `SYS` user and look for conflicting jobs:

```
SQL> SELECT * FROM dba_jobs;
```

c. After identifying conflicts, revert to the backup created in Step 4 of "Task 2: Set Up Test Product Metadata Repository" on page 11-10.

For each conflict identified in the previous step (Step 3b), re-create the failing jobs, and update the imported program to reference the new ID of the re-created job.

Using that backup, resolve the conflicts, and rerun all the steps of Task 5: Copy Data from the Production Product Metadata Repository to the Test Environment.

d. Give jobs back to the PORTAL user by running the following command from SQL*Plus as the SYS user:

```
SQL> UPDATE dba_jobs SET log_user='PORTAL', priv_user='PORTAL' where
schema_user='PORTAL';
SQL> commit;
```

e. Resolve other errors in the import log. Table 11–1 lists some of the common errors and their workarounds.

Table 11–1 Errors in the Import Log When Copying Data to a Test Environment

Error in Import Log	Workaround
IMP-00017: following statement failed with ORACLE error 604: "ALTER TABLE "WWPRO_ADAPTER_KEY$" ENABLE CONSTRAINT "WWSEC_ADAPTER_KEY$_CK1""	As the SYS user, run the SELECT statement to find the owners of these tables from SQL*Plus, and then run the ALTER statement to prefix the schema to the table name, using the statement identified within the quotes ("): SQL> SELECT OWNER FROM dba_tables WHERE table_name = 'WWPRO_ADAPTER_KEY$' ------------------------------ ORASSO PORTAL SQL> ALTER TABLE orasso.wwpro_adapter_key$ DISABLE constraint wwsec_adapter_key$_ck1; Table altered. SQL> ALTER TABLE portal.wwpro_adapter_key$ DISABLE constraint wwsec_adapter_key$_ck1; Table altered.
IMP-00017: following statement failed with ORACLE error 604: "ALTER TABLE "WWPRO_ADAPTER_KEY$" ENABLE CONSTRAINT "WWSEC_ADAPTER_KEY$_CK1"" IMP-00003: ORACLE error 2298 encountered ORA-02298: cannot validate (PORTAL.WWSTO_SESS_FK1) - parent keys not found	As the SYS user, delete that data that is specific to the session, and then re-enable the constraint: SQL> DELETE from wwsto_session_data$ / DELETE from wwsto_session_session$ / commit; SQL> ALTER TABLE portal.wwsto_session_session$ ENABLE constraint wwsto_sess_fk1; Table altered.

Table 11–1 (Cont.) Errors in the Import Log When Copying Data to a Test Environment

Error in Import Log	Workaround

`IMP-00003: ORACLE error 30510 encountered`	As the `PORTAL` user, manually re-create the logoff trigger:
	`SQL> CREATE TRIGGER logoff_trigger`
	`before logoff on schema`
	`begin`
	`  -- Call wwsec_oid.unbind to close open OID connections if any.`
	`  wwsec_oid.unbind;`
	`exception`
	`  when others then`
	`  -- Ignore all the errors encountered while unbinding.`
	`  null;`
	`end logoff_trigger;`
	`/`

`IMP-00017: following statement failed with ORACLE error 921:`
`"ALTER TABLE "WWSRC_PREFERENCE$" ADD "`
`IMP-00003: ORACLE error 921 encountered`
`ORA-00921: unexpected end of SQL command`
	Then, as the `PORTAL` user, manually create the primary key:
	`SQL> ALTER TABLE "WWSRC_PREFERENCE$" add constraint wwsrc_preference_pk`
	`primary key (subscriber_id, id)`
	`using index wwsrc_preference_idx1`
	`/`
	`begin`
	`DBMS_RLS.ADD_POLICY ('', 'WWSRC_PREFERENCE$',`
	`'WEBDB_VPD_POLICY',`
	`'', 'webdb_vpd_sec', 'select, insert, update, delete', TRUE,`
	`static_policy=>true);`
	`end ;`
	`/`

f. Compile all the invalid objects from the imported schemas.

Run the following script from SQL*Plus as the `SYS` user from the *`TEST_INFRA_HOME`*`/rdbms/admin` directory:

```
@utlrp.sql
```

g. If the following query in the `PORTAL` schema returns more than `PORTAL_PUBLIC`:

```
SQL> SELECT DISTINCT DB_USER FROM WWSEC_PERSON$;
```

Then, execute the following commands from SQL*Plus as the `PORTAL` user:

```
SET HEAD OFF
SET LINES 4000
SET SQLPROMPT '--'
SET SQLNUMBER off
SET LINESIZE 132
SPOOL DBUSERS.SQL
SELECT DISTINCT 'ALTER USER '||DB_USER ||' GRANT CONNECT THROUGH PORTAL;'
FROM WWSEC_PERSON$;
SPOOL OFF
```

Run `DBUSERS.SQL` in the target portal instance to grant connect through privilege to database users associated with portal users. In most cases, this will simply be `PORTAL_PUBLIC`, which already has the necessary grant.

h. Drop the temporary login trigger.

This step is only necessary if the test server is a Standard Edition database and you performed Step 2h of this procedure; this step is not needed for an Enterprise Edition database.

Run the following script from SQL*Plus as the `PORTAL` user from the *`TEST_INFRA_HOME`*`/portal/admin/plsql/wwhost`:

```
@droptrig.sql PORTAL
```

i. Re-create and re-index the intermedia OracleAS Portal table.

Run the following scripts from SQL*Plus as the `PORTAL` user from the *`MIDTIER_HOME`*`/portal/admin/plsql/wws` directory:

```
@inctxgrn.sql
@ctxcrind.sql
```

4. Update the OracleAS Portal instance from the Portal Dependency Settings file (`iasconfig.xml`) file by running the `ptlconfig` script from the *`TEST_MIDTIER_HOME`*`/portal/conf` directory:

```
ptlconfig -dad dad_name
```

In the syntax, *`dad_name`* is Database Access Descriptor (DAD) name. The default name set at installation is `portal`. Substitute the DAD name if it is not set to `portal`.

See Also: *Oracle Application Server Portal Configuration Guide*.

5. Perform a backup of the test product Metadata Repository, Identity Management configuration files, and Identity Management Metadata Repository with the OracleAS Backup and Recovery Tool. See Chapter 20.

Task 6: Deploy Applications to the Test Middle Tier

Deploy J2EE application EAR files to the test middle tier. You can use one of the following mechanisms:

1. Deploy J2EE application EAR files to the test middle tier. You can use one of the following mechanisms:

- Use the DCM `deployApplication` command
- Navigate to the **OC4J Home** page -> **Applications** tab in Oracle Enterprise Manager 10*g* Application Server Control Console, and click **Deploy EAR file**.

See Also:

- *Oracle Application Server Containers for J2EE User's Guide* for more information about redeploying application EAR files
- "Deploying a New OC4J Application" in Enterprise Manager Online Help for instructions

2. Copy the configuration settings from the production environment to the test environment for all OracleAS Portal and OracleBI Discoverer applications.

For OracleBI Discoverer, you need to copy the following files to the production environment:

- `configuration.xml` and `configuration.xsd` files

  (10.1.2.0.*x* on Unix) *ORACLE_HOME*/discoverer/config
  (9.0.4.*x* on Unix) *ORACLE_HOME*/j2ee/OC4J_BI_
  FORMS/applications/discoverer/web/WEB-INF

  (10.1.2.0.*x* on Windows) *ORACLE_HOME*\discoverer\config
  (9.0.4.*x* on Windows) *ORACLE_HOME*\j2ee\OC4J_BI_
  FORMS\applications\discoverer\web\WEB-INF

- `web.xml` file

  (10.1.2.0.*x* on Unix) *ORACLE_HOME*/j2ee/OC4J_BI_
  Forms/applications/discoverer/discoverer/WEB-INF
  (9.0.4.*x* on Unix) *ORACLE_HOME*\j2ee\OC4J_BI_
  FORMS\applications\discoverer\web\WEB-INF

  (10.1.2.0.*x* on Windows) *ORACLE_HOME*\j2ee\OC4J_BI_
  Forms\applications\discoverer\discoverer\WEB-INF
  (9.0.4.*x* on Windows) *ORACLE_HOME*\j2ee\OC4J_BI_
  FORMS\applications\discoverer\web\WEB-INF

- `prefs.txt` file

  (10.1.2.0.*x* on UNIX) *ORACLE_HOME*/discoverer
  (10.1.2.0.*x* on Windows) *ORACLE_HOME*\discoverer

  `prefs.txt` is not available in 9.0.4.*x* environments.

- `.reg_key.dc` file

  (10.1.2.0.*x* and 9.0.4.*x* on UNIX) *ORACLE_HOME*/discoverer
  (10.1.2.0.*x* and 9.0.4.*x* on Windows) *ORACLE_HOME*\discoverer

  You cannot copy this file across different operating systems. If this migration is happening between little-endian to big-endian operating systems, then run the `convertreg.pl` script:

  perl convertreg.pl *test_reg_key.dc_file production_reg_key.dc_file*

  You can use the PERL that installs with Oracle Application Server. You can find `convertreg.pl` in the following directory:

  (UNIX) *ORACLE_HOME*/discovere/util
  (Windows) *ORACLE_HOME*\discoverer\util

  Windows and Linux are examples of little-endian operating systems and Solaris and HP-UX are examples of big-endian operating systems. Do not use this utility when moving from little-endian to little-endian operating systems or big-endian to big-endian operating systems. For these migrations, omit this step (in which case the destination system will have a default set of user preferences).

Note: The `convertreg.pl` script does not exist on Windows in releases 9.0.4.*x*. For environments supporting this release, download the patch for this utility. The related Automated Release Updates (ARU) number is ARU 7462620.

You can download this patches from Oracle*Metalink*:

`http://metalink.oracle.com`

---

3. If OracleAS Portal is pointing to a J2EE application provider, then if desired, deploy these applications to the test middle tier and configure OracleAS Portal to point to the test middle tier.
4. Perform a backup of the test middle tier configuration files, product Metadata Repository, Identity Management configuration files, and Identity Management Metadata Repository with the OracleAS Backup and Recovery Tool. See Chapter 20.

Task 7: Copy OracleBI Discoverer Data

To copy the OracleBI Discoverer data:

1. Create a new database schema containing an empty EUL:

```
TEST_MIDTIER_HOME/bin/eulapi -connect dba/password@database_service_name
-create_eul [-apps_mode] -default_tablespace default_tablespace_name
-temporary_tablespace temp_tablespace_name -user new_eul_schema
-password new_eul_schema_password -log /tmp/createeul.log
```

   Use the `-apps_mode` flag if you are creating a copy of an Oracle Applications mode EUL.

   If you intend to copy data to another customer database in the test environment, use the service name of the test database for *`database_service_name`*.

2. Import the copied contents into the new EUL schema:

```
TEST_MIDTIER_HOME/bin/eulapi -connect
new_eul_owner/password@database_service_name
-import /tmp/file1.eex -log /tmp/imp.log
```

   The mapping of source EUL schemas to new EUL schemas should be noted in order to support the migration of Discoverer connections as detailed in the next section.

Note: The `eulapi` utility does not exist on Windows in releases 9.0.4.*x* and 10.1.2.0.*x*. For environments supporting these releases, download the patches for this utility. The related Automated Release Updates (ARU) number for 9.0.4.*x* is ARU 7462620 and 10.1.2.0.*x* is ARU 7462623.

You can download these patches from Oracle*Metalink*:

`http://metalink.oracle.com`

Task 8: Clean Up the Test Environment

To cleanup the test environment prior to using it:

1. If your OracleAS Single Sign-On migration data includes OracleBI Discoverer connection data, perform the following to update the connection data:

   a. Connect to the test Identity Management database as `ORASSO` and run the following SQL:

   ```
   SQL> UPDATE ORASSO_DS.WWSSO_PSEX_USER_INFO$
   SET FVAL2 = '&NewEulName'
   WHERE FNAME2 = 'eul' and FVAL2 = '&OldEulName' and
         FNAME5 = 'database' and FVAL5 = '&DbConnectString';
   ```

   Where:

   – `&NewEULName` is the name of the new EUL that was created after copying the EUL

   – `&OldEulName` is the name of the production EUL

   – `&DbConnectString` is the net service name or the connect descriptor for the database where the EUL resides and that was used when creating the Discoverer Connection pointing to this EUL

   Note: When you deploy OracleBI Discoverer in non-OracleAS Single Sign-On mode, users create private connections, which are stored in the PSTORE. However, OracleBI Discoverer needs to store a cookie on the browser for identifying the browser that created the connections. These cookies will not be available to the test server and users will not be able to get to their non-OracleAS Single Sign-On private connections stored in PSTORE. However, users can re-create these connections and all existing workbooks will be preserved.

   b. To copy the OLAP connections, connect to the Identity Management database as `ORASSO` and run the following SQL:

   ```
   SQL> UPDATE ORASSO_DS.WWSSO_PSEX_USER_INFO$
   SET  FVAL4 = '&New_Db_Details'
   WHERE user_prefs LIKE '%olap_zone=true%' AND
         FNAME4 = 'database' and FVAL4 = '&Old_Db_Details';
   ```

   Where:

   – `&New_Db_Details` is in the format:

   `computer``:`*`listener_port`*`:`*`database_SID`*

   For example, `testserver.us.company.com:1521:testdb`

   – `&Old_Db_Details` is in the format:

   `computer``:`*`listener_port`*`:`*`database_SID`*

   For example, `prodserver.us.company.com:1521:proddb`

2. Clean up OracleAS Single Sign-On data:

   After the OracleAS Single Sign-On import, partner applications registered with the production system are also imported to the test environment. This includes the partner registrations with the production host name. You should remove all of these.

   a. Log into the OracleAS Single Sign-On system by accessing the following URL:

   `http://`*`test_infra.domain.com`*`:`*`port`*`/pls/orasso`

b. Click **SSO Server Administration**.

c. Click **Administer Partner Applications**.

d. Scan the list of links by holding the mouse over the partner application name, and delete all the entries that pertain to the production host. Corresponding entries should now exist for the test host.

3. Change the registration of Discoverer Portlet Provider on the test OracleAS Portal instance.

   Modify the registration of the Discoverer Portlet Provider in the OracleAS Portal instance and change the URL for the provider from:

   `http://`*`production_host`*`:`*`production_OHS_port`*`/discoverer/portletprovider`

   To:

   `http://`*`test_host`*`:`*`production_OHS_port`*`/discoverer/portletprovider`

   a. Log in to OracleAS Portal as the administrator (for example, `PORTAL`).

   b. Click the **Administer** tab.

   c. Click the **Portlets** sub-tab.

   d. In the Remote Providers portlet, enter the Discoverer Portlet Provider name in the **Name** field. Click **Edit**.

   e. Click the **Connection** tab.

   f. In the **URL** field, update the port to the new port number. Click **Apply**.

   g. Click **OK**.

   Modify all Web providers to use the new URL.

11.2 Upgrading the Test Environment

Once you establish the test environment, use it for testing upgrades or applying patchsets.

After creating the test environment, consider transforming it to a highly available installation.

See Also:

- *Oracle Application Server Upgrade and Compatibility Guide* for further information about upgrading
- *Oracle Application Server High Availability Guide* for more information about transforming the test environment to a highly available environment

12

Changing from a Test to a Production Environment

This chapter provides use cases for changing from a test to a production environment. You can develop and test applications in a test environment, and then eventually roll out the test applications and, optionally, test data to your production environment. You can also use this approach for testing and rolling out upgrades.

It contains the following topics:

- Understanding the Options for Creating a Production Middle Tier
- Case 1: Moving J2EE Applications to a Production Environment
- Case 2: Moving Non-J2EE Applications to a Production Environment
- Case 3: Moving Product-Specific Metadata from Test Metadata Repository to Production Metadata Repository

Table 12–1 provides guidance on how to find the use case scenario that applies to your application and configuration environment.

Table 12–1 Test-to-Production Use Cases

Type of Application	Configuration Assumptions	Refer To This Use Case Scenario:
J2EE		
Scenario 1	**Test Environment**: Middle-tier instance already exists. **Production Environment**: Middle-tier instance already exists.	**See Also:** Section 12.2.1, "Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance"
Scenario 2	**Test Environment**: Middle-tier instance already exists. **Production Environment**: The production environment does not exist. You want to create a middle-tier instance.	**See Also:** Section 12.2.2, "Scenario 2: Moving J2EE Applications from a Test Middle Tier Without Identity Management to a New Production Environment"
Scenario 3	**Test Environment Includes**: Middle-tier instance and Identity Management already exists. **Production Environment**: The production environment does not exist. You want to create a middle-tier instance and Identity Management.	**See Also:** Section 12.2.3, "Scenario 3: Moving J2EE Applications from a Test Middle Tier with Identity Management to a New Production Environment"
Non-J2EE		

Table 12–1 (Cont.) Test-to-Production Use Cases

Type of Application	Configuration Assumptions	Refer To This Use Case Scenario:
Scenario 1	**Test Environment**: The test environment does not exist. You want to create a middle-tier instance and Identity Management. **Production Environment**: Identity Management already exists. You want to create a middle-tier instance.	**See Also:** Section 12.3.1, "Scenario 1: Moving Applications from a Test Middle Tier with Identity Management to a Production Environment with a Preexisting Identity Management"
Scenario 2	**Test Environment**: The test environment does not exist. You want to create a middle-tier instance, Identity Management, and a Metadata Repository for product metadata. **Production Environment**: Identity Management already exists. You want to create a middle-tier instance, and a Metadata Repository for product metadata.	**See Also:** Section 12.3.2, "Scenario 2: Moving Applications from a Test Middle Tier with Identity Management and a Product Metadata Repository to an Existing Production Environment with Identity Management"
Scenario 3	**Test Environment**: The test environment does not exist. You want to create multiple middle-tier instances, each with Identity Management and Metadata Repository for product metadata. **Production Environment**: Identity Management already exists. You want to create multiple middle-tier instances, each with a Metadata Repository for product metadata.	**See Also:** Section 12.3.3, "Scenario 3: Moving Applications from Multiple Test Middle Tiers with Dedicated Identity Management Metadata Repositories"

In addition to J2EE and non-J2EE use case, this chapter also provides guidance on moving incremental product-specific data from a test environment to a production environment.

See Also:

- Section 12.4.1, "OracleAS Portal"
- Section 12.4.2, "OracleBI Discoverer"

12.1 Understanding the Options for Creating a Production Middle Tier

Many of the scenarios presented in this chapter describe creating a production middle-tier instance in a configuration that already includes a test middle-tier instance for application development. For these scenarios, you have the choice of three options. You can:

- Clone the test middle-tier instance.

  Use this option to preserve configuration settings.

- Point the test middle-tier instance to the production Identity Management.

  Use this option if you want to repurpose the test middle-tier instance as the production middle-tier instance.

- Install a middle-tier instance into the production environment, and then redeploy applications.

Use this option when you are creating the production middle-tier instance in a different operating system than the test middle-tier instance.

12.2 Case 1: Moving J2EE Applications to a Production Environment

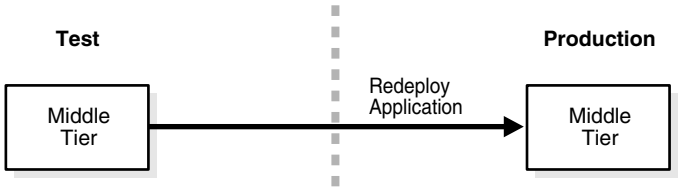
This section demonstrates how to move J2EE applications from a test environment to a production environment. The following topics present typical scenarios:

- Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance
- Scenario 2: Moving J2EE Applications from a Test Middle Tier Without Identity Management to a New Production Environment
- Scenario 3: Moving J2EE Applications from a Test Middle Tier with Identity Management to a New Production Environment

12.2.1 Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance

In this scenario, you have a J2EE application on a test middle-tier instance, and you want to redeploy it to an existing production middle tier. Figure 12–1 shows this scenario.

Figure 12–1 Redeploying a J2EE Application to an Existing Production Middle Tier



12.2.1.1 Preexisting Configuration Assumptions

This use case assumes the following configuration:

- Separate test and production environments already exist.
- You have a J2EE application in the test middle-tier instance.

12.2.1.2 Procedure

Redeploy J2EE application EAR files to the new middle tier. You can use one of the following mechanisms:

- Use the DCM `redeployApplication` command
- Navigate to the **OC4J Home** page -> **Applications** tab in Oracle Enterprise Manager 10*g* Application Server Control Console and click **Deploy EAR file**.

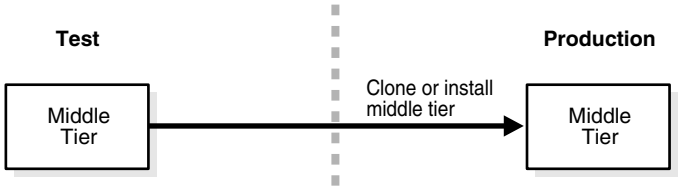
See Also:

- *Oracle Application Server Containers for J2EE User's Guide* for more information about redeploying application EAR files
- "Deploying a New OC4J Application" in Enterprise Manager Online Help for instructions

12.2.2 Scenario 2: Moving J2EE Applications from a Test Middle Tier Without Identity Management to a New Production Environment

In this scenario, you have a J2EE application on a test middle-tier instance without Identity Management. You want to create a new production environment that includes a middle-tier instance, and move the J2EE application. Figure 12–2 shows this scenario.

Figure 12–2 Moving a J2EE Application to a New Production Middle Tier Without Identity Management



12.2.2.1 Preexisting Configuration Assumptions

This use case assumes the following configuration:

- You have an existing test environment that includes a middle-tier instance with a J2EE application.
- The production environment does not exist.

12.2.2.2 Procedure

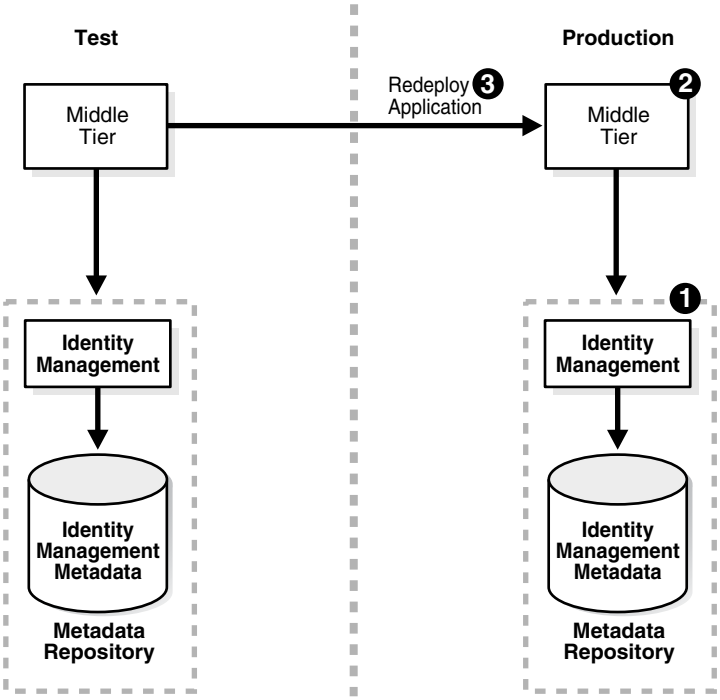
For this scenario, you must create the production middle-tier instance. You have a choice of two configuration options. You can either:

- Clone the test middle-tier instance to make a production middle-tier instance. See Section 10.4, "Cloning Oracle Application Server Instances".
- Install a middle tier into the production environment, and then redeploy the J2EE application, as described in Section 12.2.1, "Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance".

12.2.3 Scenario 3: Moving J2EE Applications from a Test Middle Tier with Identity Management to a New Production Environment

In this scenario, you have a J2EE application on a test middle-tier instance with Identity Management. You want to create a new production environment that includes a middle-tier instance with the J2EE application and Identity Management with a Metadata Repository. Figure 12–3 shows this scenario.

Figure 12–3 Moving a J2EE Application from a Test Middle Tier with Identity Management



12.2.3.1 Preexisting Configuration Assumptions

This scenario assumes the following configuration:

- The test environment includes a middle-tier instance with a J2EE application and an Identity Management installation with a Metadata Repository.
- The production middle-tier instance does not the exist, and the production Identity Management may exist.

12.2.3.2 Procedure

For this scenario, you create the production environment by following these tasks:

1. If the production Identity Management and Metadata Repository does not exist, install and configure it:
   a. Install Oracle Application Server using Oracle Universal Installer.
   b. From the Select a Product to Install screen, choose OracleAS Infrastructure.
   c. From the Select Installation Type, choose Identity Management and OracleAS Metadata Repository.
   d. From the Select Configuration Options screen, choose Oracle Internet Directory.
2. Install the production middle-tier instance.
   a. Install Oracle Application Server using Oracle Universal Installer.
   b. From the Select a Product to Install screen, choose the appropriate middle tier type for your environment.

3. Redeploy the J2EE application, as described in Section 12.2.1, "Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance".

12.3 Case 2: Moving Non-J2EE Applications to a Production Environment

In this use case, a test environment is created from a preexisting production environment. Once data is tested, it is moved back to the production environment. This use case is useful for environments in which you want to create a test environment that simulates a production environment, such as testing and rolling out patches. The following topics present typical scenarios:

- Scenario 1: Moving Applications from a Test Middle Tier with Identity Management to a Production Environment with a Preexisting Identity Management
- Scenario 2: Moving Applications from a Test Middle Tier with Identity Management and a Product Metadata Repository to an Existing Production Environment with Identity Management
- Scenario 3: Moving Applications from Multiple Test Middle Tiers with Dedicated Identity Management Metadata Repositories

12.3.1 Scenario 1: Moving Applications from a Test Middle Tier with Identity Management to a Production Environment with a Preexisting Identity Management

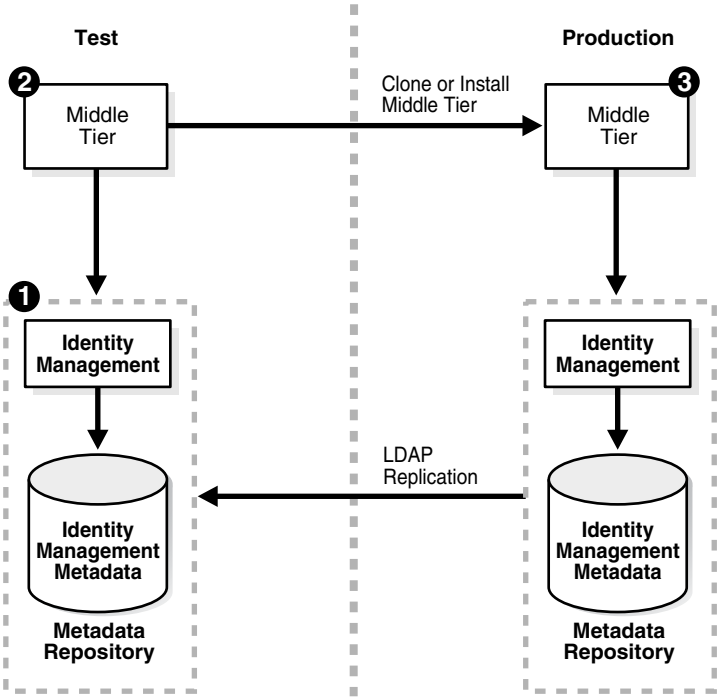
In this scenario, you have an existing production environment that includes an Identity Management installation with a Metadata Repository. You would like to create a test environment for developing and testing applications. You would then like to roll out these applications to the production environment.

For this scenario, you create a test environment by installing and setting up a replica of the production Identity Management. The Identity Management has its own Metadata Repository. The Oracle Internet Directory in the test Identity Management is an LDAP-based replica of the production Oracle Internet Directory. Replication takes place constantly from the production Oracle Internet Directory to the test Oracle Internet Directory. This replica has its own Metadata Repository. You then install a test middle-tier instance to use the test Identity Management.

After developing and testing your applications, you then create a production middle-tier instance by either cloning the test middle-tier instance, or installing a middle tier into the production environment, and then redeploying the applications.

Figure 12–4 shows an example of this scenario.

Figure 12–4 Moving an Application from a Test Middle Tier with Identity Management to a New Production Environment



12.3.1.1 Preexisting Configuration Assumptions

This scenario assumes the following configuration:

- The test environment does not exist.
- The production environment includes only Identity Management with a Metadata Repository.

12.3.1.2 Procedure

This procedure contains the following tasks:

- Task 1: Configure the Test Identity Management and Metadata Repository
- Task 2: Set Up the Test Middle-Tier Instance
- Task 3: Set Up the Production Middle-Tier Instance

Task 1: Configure the Test Identity Management and Metadata Repository

To configure the test Identity Management and Metadata Repository, set up Identity Management in the test environment. Use these subtasks to perform this configuration:

1. Perform procedure "Install and Set Up the Test Identity Management and Metadata Repository" on page 12-15.
2. Perform procedure "Identify the Test Oracle Internet Directory as a Pilot" on page 12-15.

Task 2: Set Up the Test Middle-Tier Instance

To configure the test middle-tier instance, install the middle-tier instance and develop and test applications. Use these subtasks to perform this configuration:

1. Perform procedure "Install Test Middle-Tier Instance" on page 12-15.
2. Perform procedure "Develop and Test Your Applications" on page 12-15.

Task 3: Set Up the Production Middle-Tier Instance

To create a production middle-tier instance, you can either clone the test middle-tier instance or perform a middle-tier installation. If you do not want to create a separate production middle-tier instance, you can choose to point the test middle-tier instance to the production Identity Management.

When you clone a test middle-tier instance, you must also migrate data from the test Identity Management to the production Identity Management, and associate the production middle-tier instance with the production Identity Management. Perform the following procedures to clone the test middle-tier instance:

1. Perform procedure "Clean Up Test Oracle Internet Directory" on page 12-15.
2. Perform procedure "Quiesce the Distributed Directory Environment" on page 12-16.
3. Perform procedure "End Pilot Mode on the Test Oracle Internet Directory" on page 12-16.
4. Perform procedure "Migrate Oracle Internet Directory Data to Production" on page 12-17.
5. Perform tasks in procedure Section 10.4, "Cloning Oracle Application Server Instances" on page 10-5.
6. Perform procedure "Change Middle-Tier Instance to the Production Identity Management" on page 12-21.

To point the test middle-tier instance to the production Identity Management, perform the same tasks for cloning, except for Task 5.

To install the production middle-tier instance:

1. Install the production middle-tier instance.
   a. Install Oracle Application Server using Oracle Universal Installer.
   b. From the Select a Product to Install screen, choose the appropriate middle tier type for your environment.
2. Redeploy the application, as described in Section 12.2.1, "Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance".

   When you install, data in the test Identity Management is not migrated from to the production environment.

12.3.1.3 Creating a Second Middle-Tier Instance in the Production Environment

If you want to deploy a test application to another middle-tier instance in the production environment, perform these tasks to create a second middle-tier instance:

1. Perform subtask 2 in "Task 1: Configure the Test Identity Management and Metadata Repository" on page 12-7.
2. Perform procedure "Task 2: Set Up the Test Middle-Tier Instance".
3. Perform procedure "Task 3: Set Up the Production Middle-Tier Instance".

12.3.2 Scenario 2: Moving Applications from a Test Middle Tier with Identity Management and a Product Metadata Repository to an Existing Production Environment with Identity Management

This scenario is similar to Section 12.3.1, "Scenario 1: Moving Applications from a Test Middle Tier with Identity Management to a Production Environment with a Preexisting Identity Management", except the test middle-tier instance has an additional Metadata Repository for product metadata. With this scenario, you develop and test one application or a set of related applications against the same Identity Management. You then roll out these applications at the same time to the production environment. After deploying the first set of applications, you can then develop, test, and deploy a second set of applications. In this manner, this scenario works like an assembly line.

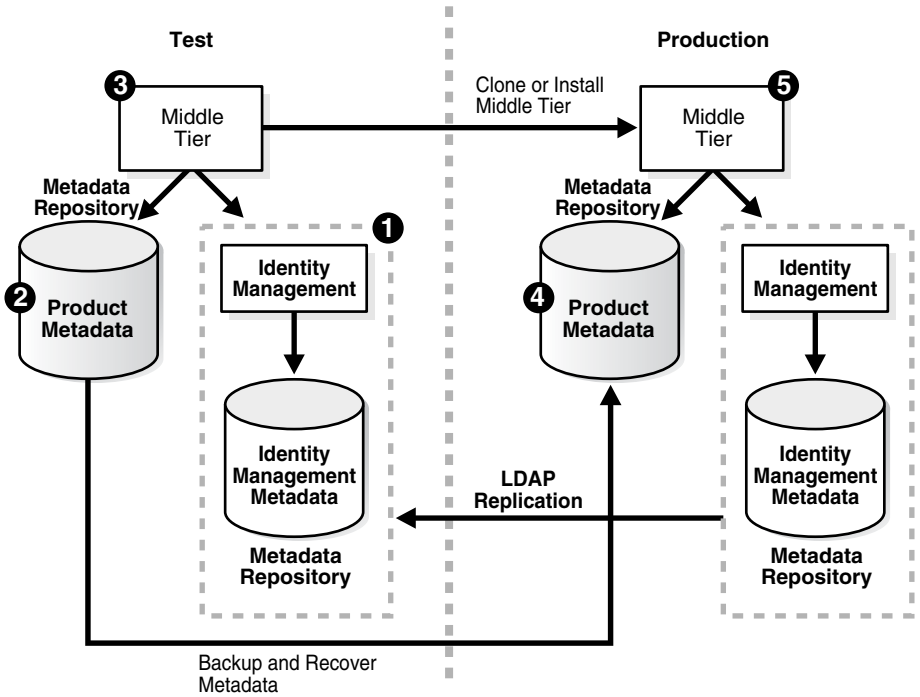
If you prefer to instead deploy applications at different times, then consider Section 12.3.3, "Scenario 3: Moving Applications from Multiple Test Middle Tiers with Dedicated Identity Management Metadata Repositories".

Like Scenario 1, you start by creating a test environment with a replica of the production Identity Management. You then install a test middle-tier instance to use the test Identity Management and a separate Metadata Repository for the product metadata.

You then configure the production environment. You move the test product Metadata Repository to the production environment. You then create a production middle-tier instance by either cloning the test middle-tier instance, or installing a middle tier into the production environment, and then redeploying the applications.

Figure 12–5 shows an example of this scenario.

Figure 12–5 Moving an Application from a Test Middle Tier with Identity Management and Product Metadata Repository to an Existing Production Environment with Identity Management



12.3.2.1 Preexisting Configuration Assumptions

This scenario assumes the following configuration:

- The test environment does not exist.
- The production environment includes only Identity Management with a Metadata Repository.

12.3.2.2 Procedure

This procedure contains the following tasks:

- Task 1: Configure the Test Identity Management and Metadata Repository
- Task 2: Create the Test Product Metadata Repository
- Task 3: Set Up the Test Middle-Tier Instance
- Task 4: Move Test Product Metadata Repository to Production Environment
- Task 5: Set Up the Production Middle-Tier Instance

Task 1: Configure the Test Identity Management and Metadata Repository

To configure the test Identity Management and Metadata Repository, set up Identity Management in the test environment. Follow these procedures to perform this configuration:

1. Perform procedure "Install and Set Up the Test Identity Management and Metadata Repository" on page 12-15.
2. Perform procedure "Identify the Test Oracle Internet Directory as a Pilot" on page 12-15.

Task 2: Create the Test Product Metadata Repository

To configure the test product metadata repository, follow the procedure "Install and Populate Test Product Metadata Repository" on page 12-15.

Task 3: Set Up the Test Middle-Tier Instance

To configure the test middle-tier instance, install the middle-tier instance and develop and test applications. Follow these procedures to perform this configuration:

1. Perform procedure "Install Test Middle-Tier Instance" on page 12-15.
2. Perform procedure "Develop and Test Your Applications" on page 12-15.

Task 4: Move Test Product Metadata Repository to Production Environment

To configure the production product Metadata Repository, follow the procedure "Move the Test Product Metadata Repository to Production" on page 12-17.

Task 5: Set Up the Production Middle-Tier Instance

To create a production middle-tier instance, you can either clone the test middle-tier instance or perform a middle-tier installation. If you do not want to create a separate production middle-tier instance, you can choose to point the test middle-tier instance to the production Identity Management.

When you clone a test middle-tier instance, you must also migrate data from the test Identity Management to the production Identity Management, and associate the production middle-tier instance with the production Identity Management. Perform these procedures to clone the test middle-tier instance:

1. Perform procedure "Clean Up Test Oracle Internet Directory" on page 12-15.
2. Perform procedure "Quiesce the Distributed Directory Environment" on page 12-16.
3. Perform procedure "End Pilot Mode on the Test Oracle Internet Directory" on page 12-16.
4. Perform procedure "Migrate Oracle Internet Directory Data to Production" on page 12-17.
5. Perform tasks in procedure Section 10.4, "Cloning Oracle Application Server Instances" on page 10-5.
6. Perform procedure "Change Middle-Tier Instance to the Production Identity Management" on page 12-21.

To point the test middle-tier instance to the production Identity Management, perform the same tasks for cloning, except for Task 5.

To install the production middle-tier instance:

1. Install the production middle-tier instance.
2. Redeploy the application, as described in Section 12.2.1, "Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance".

   When you install, test data in the test Identity Management is not migrated from to the production environment.

   See Also: Section 12.3.1.3, "Creating a Second Middle-Tier Instance in the Production Environment" for further information

12.3.3 Scenario 3: Moving Applications from Multiple Test Middle Tiers with Dedicated Identity Management Metadata Repositories

This scenario demonstrates how to move application data from multiple test middle-tier instances to a production environment. With this scenario, you develop and test set applications or two sets of related applications in separate test environments, each with a dedicated Identity Management. You then roll out these applications at different deployment times to the production environment.

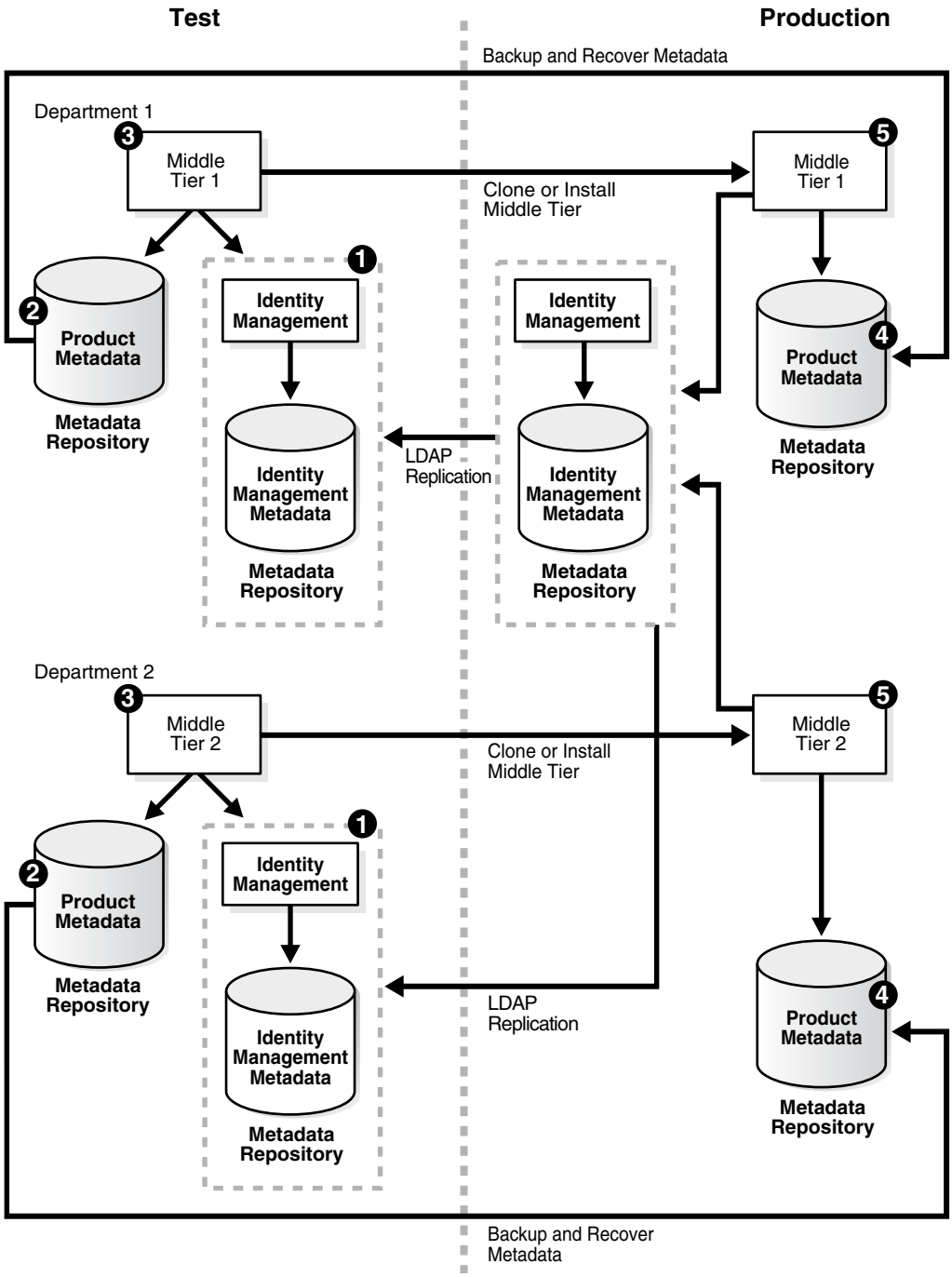
If you prefer to instead develop, test, and deploy one set of applications at a time, then consider Section 12.3.2, "Scenario 2: Moving Applications from a Test Middle Tier with Identity Management and a Product Metadata Repository to an Existing Production Environment with Identity Management".

In this scenario, you have an existing production environment that includes an Identity Management installation with a Metadata Repository. You would like to create a test environment for developing and testing your application. You want this test environment to includes two middle-tier instances, Department 1 and 2, each with a dedicated Identity Management and product Metadata Repositories. Once the applications are tested, you would like to roll them out to a production environment that also includes middle tiers for Department 1 and 2, each with dedicated product Metadata Repositories.

To accomplish this move without data conflicts of Identity Management, you move data from the departments in a serial manner.

Figure 12–6 shows an example of this scenario.

Figure 12–6 Moving Applications from Separate Test Middle Tiers



12.3.3.1 Preexisting Configuration Assumptions

This scenario assumes the following configuration:

- The test environment does not exist.
- The production environment includes only Identity Management with a Metadata Repository.

12.3.3.2 Procedure

This procedure contains the following tasks:

- Task 1: Configure the Test Identity Management and Metadata Repository
- Task 2: Create the Test Product Metadata Repository
- Task 3: Set Up the Test Middle-Tier Instance
- Task 4: Move Test Product Metadata Repository to Production Environment
- Task 5: Set Up the Production Middle-Tier Instance

Task 1: Configure the Test Identity Management and Metadata Repository for Department 1 and 2

To configure the test Identity Management and Metadata Repository, set up Identity Management in the test environment in both departments. Follow these procedures to perform this configuration:

1. Perform procedure "Install and Set Up the Test Identity Management and Metadata Repository" on page 12-15.
2. Perform procedure "Identify the Test Oracle Internet Directory as a Pilot" on page 12-15.

Ensure each Identity Management and Metadata Repository has distinct user sets of data.

Task 2: Create the Test Product Metadata Repository

To configure the test product metadata repository, follow the procedure "Install and Populate Test Product Metadata Repository" on page 12-15.

Configure each product Metadata Repository in each department with a unique host name, instance name, global database name, and Oracle System Identifier (SID).

Task 3: Set Up the Test Middle-Tier Instance

To configure the test middle-tier instances, install the middle-tier instance and develop and test applications. Follow these procedures to perform this configuration:

1. Install Test Middle-Tier Instance
2. Develop and Test Your Applications

Task 4: Move Test Product Metadata Repository to Production Environment

To configure the production product Metadata Repositories, follow the procedure "Move the Test Product Metadata Repository to Production" on page 12-17 first for Department 1 and then for Department 2. When you merge the data from Department 2 with information already present from Department 1, you must merge user groups and ensure there are no user data conflicts.

Task 5: Set Up the Production Middle-Tier Instance

To create a production middle-tier instance for each department, you can either clone the test middle-tier instance or perform a middle-tier installation. If you do not want to create a separate production middle-tier instance, you can choose to point the test middle-tier instance to the production Identity Management.

When you clone a test middle-tier instance, you must also migrate data from the test Identity Management to the production Identity Management, and associate the production middle-tier instance with the production Identity Management. Perform these procedures to clone the test middle-tier instance:

1. Perform procedure "Clean Up Test Oracle Internet Directory" on page 12-15.

2. Perform procedure "Quiesce the Distributed Directory Environment" on page 12-15.
3. Perform procedure "End Pilot Mode on the Test Oracle Internet Directory" on page 12-16.
4. Perform procedure "Migrate Oracle Internet Directory Data to Production" on page 12-17.
5. Perform tasks in procedure Section 10.4, "Cloning Oracle Application Server Instances" on page 10-5.
6. Perform procedure "Change Middle-Tier Instance to the Production Identity Management" on page 12-21.

To point the test middle-tier instance to the production Identity Management, perform the same tasks for cloning, except for Task 5.

To install the production middle-tier instance:

1. Install the production middle-tier instance.
   a. Install Oracle Application Server using Oracle Universal Installer.
   b. From the Select a Product to Install screen, choose the appropriate middle tier type for your environment.
2. Redeploy the application, as described in Section 12.2.1, "Scenario 1: Redeploying J2EE Applications to an Existing Production Environment with a Middle-Tier Instance".

   When you install, test data in the test Identity Management is not migrated from to the production environment.

   See Also: Section 12.3.1.3, "Creating a Second Middle-Tier Instance in the Production Environment" for further information

12.3.4 Common Procedures for Scenarios in Use Case 2

Common procedures for Section 12.3, "Case 2: Moving Non-J2EE Applications to a Production Environment" include:

- Install and Set Up the Test Identity Management and Metadata Repository
- Identify the Test Oracle Internet Directory as a Pilot
- Install and Populate Test Product Metadata Repository
- Install Test Middle-Tier Instance
- Develop and Test Your Applications
- Clean Up Test Oracle Internet Directory
- Quiesce the Distributed Directory Environment
- End Pilot Mode on the Test Oracle Internet Directory
- Move the Test Product Metadata Repository to Production
- Migrate Oracle Internet Directory Data to Production
- Change Middle-Tier Instance to the Production Identity Management

Install and Set Up the Test Identity Management and Metadata Repository

In this procedure, you install and set up the test Identity Management and its associated Metadata Repository. The test Identity Management is an LDAP-based replica of the original Identity Management.

1. Read Section H.1, "About LDAP-Based Replicas" on page H-1 to learn about LDAP-based Replicas and how they are used for this procedure.
2. Follow the procedure in Section H.2, "Installing and Setting Up an LDAP-Based Replica" on page H-3 to install and set up the test Identity Management and Metadata Repository.

Identify the Test Oracle Internet Directory as a Pilot

Run the following command from the Oracle home of the test Oracle Internet Directory:

```
remtool -pilotreplica begin -bind test_oid_host:test_oid_port/test_replication_dn_passwd
```

In the syntax:

test_oid_host is the host name of the test directory server.

test_oid_port is the LDAP port of the test directory server.

test_replication_dn_passwd is the password of the replication DN of the test directory server. By default, it is the same as the superuser DN (`cn=orcladmin`) password.

See Also:

- Appendix H for more information about LDAP replication
- *Oracle Identity Management User Reference* for more information about `remtool` and directory replication

Install and Populate Test Product Metadata Repository

Create a new database and populate it with the OracleAS Metadata Repository.

1. Install Oracle Application Server using Oracle Universal Installer.
2. From the Select a Product to Install screen, select **OracleAS Infrastructure**.
3. From the Select Installation Type, select **Metadata Repository**.

Install Test Middle-Tier Instance

Install your test middle-tier instances and configure them to use the test Identity Management according to what you want to test:

1. Install Oracle Application Server using Oracle Universal Installer.
2. From the Select a Product to Install screen, choose the appropriate middle tier type for your environment.

Develop and Test Your Applications

Develop and test applications in your test environment.

Clean Up Test Oracle Internet Directory

You can clean up (delete) the data that is modified or added on the test Oracle Internet Directory so that it is not migrated to the production Oracle Internet Directory. This

might be a requirement of a middle-tier component or might be desired by the administrator who maintains Oracle Internet Directory consistency in the production Oracle Internet Directory.

To clean up the data, use the `ldapdelete` command-line utility and delete entries that should not be migrated.

> **See Also:** *Oracle Identity Management User Reference* for more information about the `ldapdelete` command

Quiesce the Distributed Directory Environment

It is very important to quiesce the distributed directory environment while the data migration from the test to the production takes place. This ensures that there are no conflicting updates, and therefore no data loss or corruption.

To quiesce the distributed directory environment:

1. Make sure both the test and production Oracle Internet Directories are up and running.
2. Change the directory server on the test node to read-only mode.

   On the test host, create an LDIF file named `readonly.ldif` that contains the following lines:

   ```
   dn:
   changetype:modify
   replace:orclservermode
   orclservermode:r
   ```

   Run the following command:

   ```
   TEST_HOME/bin/ldapmodify -p test_oid_port -D cn=orcladmin
    -w test_orcladmin_passwd -v -f readonly.ldif
   ```

   In the syntax:

   `test_oid_port` is the LDAP port of the test directory server.

   `test_orcladmin_password` is the password of the superuser DN (`cn=orcladmin`).
3. Wait until all the pending changes are applied to both nodes and the nodes are completely in sync. There is no tool to automatically detect this, but you can monitor the replication log files and make sure there are no new changes being processed by any node in the directory replication group, which ensures that the directory replication group is in a quiesced state.

End Pilot Mode on the Test Oracle Internet Directory

Run the following command from the Oracle home of the test Oracle Internet Directory:

```
remtool -pilotreplica end -bind test_oid_host:test_oid_port/test_replication_dn_
passwd [-bkup fname]
```

In the syntax:

`test_oid_host` is the host name of the test directory server.

`test_oid_port` is the LDAP port of the test directory server.

test_replication_dn_passwd is the password of the replication DN of the test directory server. By default, it is the same as the superuser DN (`cn=orcladmin`) password.

fname specifies the backup file in which to store entries that were modified after pilot mode was started. The entries are in LDIF format. You will use this file in procedure "Migrate Oracle Internet Directory Data to Production" on page 12-17.

See Also:

- Appendix H for more information about LDAP replication
- *Oracle Identity Management User Reference* for more information about `remtool` and directory replication

Move the Test Product Metadata Repository to Production

You have several options for moving your test product Metadata Repository to your production environment:

- You can continue to use the test Metadata Repository in your production environment, thereby deeming it to be a production Metadata Repository.

  In this case, no further action is required.

- You can copy the Metadata Repository to a production host and change your middle-tier instances to use it.

  Follow the procedure in Section 9.5, "Changing the Metadata Repository Used by a Middle-Tier Instance".

- If you do not want to retain the test data in the Metadata Repository, you can install a new Metadata Repository in the production environment, and change the middle-tier instances to use that.

  Install an Infrastructure using Oracle Universal Installer. Select the Metadata Repository only option. Register the Metadata Repository with the production Identity Management.

  Change each of the former test middle-tier instances to use the new Metadata Repository. On each middle-tier instance:

  1. Using the Application Server Control Console, navigate to the Instance Home Page for the middle-tier instance.
  2. Click **Infrastructure**.
  3. On the Infrastructure Page, in the Metadata Repository section, click **Change**.
  4. Follow the tasks in the wizard for supplying the new Metadata Repository information.
  5. When the wizard is finished, navigate to the Instance Home Page and start your instance by clicking **Start All**.

Migrate Oracle Internet Directory Data to Production

This procedure describes how to migrate Oracle Internet Directory data from a test Identity Management to the production Identity Management.

Note: Make sure the `ORACLE_HOME` and `ORACLE_SID` environment variables are set before you begin. This applies to all operating systems.

---

1. Migrate test Oracle Internet Directory data to the production environment by running the following command.

   ```
   PRODUCTION_HOME/bin/ldapaddmt -h production_oid_host
   -p production_oid_port -D "cn=orcladmin"
   -w production_orcladmin_passwd -r -f fname
   ```

   Make sure you specify the `-r` argument to migrate data and resolve conflicts. Also, ensure you specify the LDIF file you obtained in procedure "End Pilot Mode on the Test Oracle Internet Directory" on page 12-16 for the `-f` argument.

   In the syntax:

   `production_oid_host` is the host of the production directory server.

   `production_oid_port` is the LDAP port of the production directory server.

   `production_orcladmin_password` is the password of the superuser DN (`cn=orcladmin`).

   `fname` specifies the LDIF file you specified in procedure "End Pilot Mode on the Test Oracle Internet Directory" on page 12-16.

2. Validation step. Verify that the migration of Oracle Internet Directory data succeeded.

   Verify that `ldapaddmt` reported success. You can check the `add.log` file for errors, which is created in the directory from which you ran the `ldapaddmt` command.

   - If `add.log` is empty, the command succeeded.
   - If `add.log` contains errors such as `Additional Info: Parent entry not found in the directory`, then the entries in the LDIF file are not in the correct order—the child entry is before the parent entry. Run `ldapaddmt` again and this will take care of adding the child entries.

     See Also: *Oracle Internet Directory Administrator's Guide* for information on interpreting messages in log files

   If necessary, repeat step 1.

3. Migrate OracleAS Single Sign-On and Directory Integration and Provisioning data from the test Metadata Repository to the production Metadata Repository.

   To migrate the OracleAS Single Sign-On data:

   a. Obtain the `ORASSO` schema password on the test Metadata Repository:

      ```
      TEST_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
      -D "cn=orcladmin" -w test_orcladmin_passwd
      -b "orclresourcename=orasso, orclreferencename=test_oid_global_db_name,
       cn=ias infrastructure databases, cn=ias, cn=products, cn=oraclecontext" -s
       base "objectclass=*" orclpasswordattribute
      ```

      In the syntax:

      `test_oid_host` is the host of the test directory server.

      `test_oid_port` is the LDAP port of the test directory server.

      `test_orcladmin_password` is the password of the superuser DN (`cn=orcladmin`).

test_oid_global_dbname is the global database name of the test Metadata Repository.

This command prints the ORASSO password in a line like the following:

```
orclpasswordattribute=LAetjdQ5
```

b. Export the OracleAS Single Sign-On data from the test environment, ensuring that the ORACLE_HOME environment variable is set before you run this command:

```
TEST_HOME/sso/bin/ssomig -export -s orasso -p test_orasso_passwd
-c test_net_service_name -log_d $TEST_HOME/sso/log
```

In the syntax:

test_orasso_passwd is the ORASSO password obtained in the previous step.

test_net_service_name is the database name of the test Metadata Repository.

c. Copy the ssomig.dmp and ssoconf.log files from the test to the production directory server, preserving the exact full path for each file:

```
cp TEST_HOME/sso/log/ssomig.dmp PRODUCTION_HOME/sso/log/ssomig.dmp
cp TEST_HOME/sso/log/ssoconf.log PRODUCTION_HOME/sso/log/ssoconf.log
```

d. Obtain the ORASSO schema password on the production Metadata Repository:

```
PRODUCTION_HOME/bin/ldapsearch -h production_oid_host -D "cn=orcladmin"
-p production_oid_port
-w production_orcladmin_password -b "orclresourcename=orasso,
 orclreferencename=production_global_db_name, cn=ias infrastructure
 databases, cn=ias, cn=products, cn=oraclecontext"
-s base "objectclass=*" orclpasswordattribute
```

In the syntax:

production_oid_host is the host of the production directory server.

production_oid_port is the LDAP port of the production directory server.

production_orcladmin_password is the password of the superuser DN (cn=orcladmin).

production_oid_global_dbname is the global database name of the production Metadata Repository.

e. Import the OracleAS Single Sign-On data to the production Metadata Repository:

```
PRODUCTION_HOME/sso/bin/ssomig -import -overwrite -s orasso
-p production_orasso_passwd -c production_net_service_name
-log_d $PRODUCTION_HOME/sso/log -discoforce
```

In the syntax:

production_orasso_passwd is the ORASSO password obtained in the previous step.

production_net_service_name is the database name of the production Metadata Repository.

f. Validation step: Verify that the export and import of OracleAS Single Sign-On succeeded.

Verify that the OracleAS Single Sign-On migration tool reported success. You can also check the following log files for errors:

```
TEST_HOME/sso/log/ssomig.log
PRODUCTION_HOME/sso/log/ssomig.log
```

See Also: *Oracle Application Server Single Sign-On Administrator's Guide* for information on interpreting messages in the log files

To migrate the Directory Integration and Provisioning Data data:

See Also: Directory Integration and Provisioning Data documentation in the *Oracle Internet Directory Administrator's Guide* for running the following commands using the HTTPS port in environments in which the Oracle Internet Directory HTTP port is disabled

a. Stop the Directory Integration and Provisioning Data server on the test directory server:

```
TEST_HOME/bin/oidctl server=odisrv instance=1 stop
```

b. Migrate the Directory Integration and Provisioning Data to the production Metadata Repository:

```
TEST_HOME/bin/dipassistant reassociate -src_ldap_host
 test_oid_host -src_ldap_port test_oid_port
-dst_ldap_host production_oid_host -dst_ldap_port
 production_oid_port -src_ldap_passwd
 test_orcladmin_passwd -dst_ldap_passwd production_orcladmin_passwd
```

This command prints log messages to:

```
TEST_HOME/ldap/odi/log/reassociate.log
```

In the syntax:

test_oid_host is the host of the test directory server.

test_oid_port is the LDAP port of the test directory server.

production_oid_host is the host of the production directory server.

production_oid_port is the LDAP port of the production directory server.

test_orcladmin_password is the password of the superuser DN (`cn=orcladmin`) for the test directory server.

production_orcladmin_password is the password of the superuser DN (`cn=orcladmin`) for the production directory server.

c. Stop the Directory Integration and Provisioning Data server on the production directory server:

```
PRODUCTION_HOME/bin/oidctl server=odisrv instance=1 stop
```

d. Register the Directory Integration and Provisioning Data server on the production directory server:

```
PRODUCTION_HOME/bin/odisrvreg -D "cn=orcladmin"
 -w production_orcladmin_passwd -p production_oid_port
```

```
-h production_oid_host
```

In the syntax:

production_orcladmin_password is the password of the superuser DN (cn=orcladmin).

production_oid_port is the LDAP port of the production directory server.

production_oid_host is the host of the production directory server.

e. Start the Directory Integration and Provisioning Data server on the production directory server:

```
PRODUCTION_HOME/bin/oidctl server=odisrv instance=1
flags="port=production_oid_port" start
```

In the syntax, *production_oid_port* is the LDAP port of the production directory server.

4. (Optional) Perform post-migration cleanup tasks.

Some middle-tier components might have special cleanup requirements after you have changed to the production environment. You can perform these cleanup tasks to the test environment after the middle-tier instances have been changed to the production node.

Change Middle-Tier Instance to the Production Identity Management

In each production middle-tier instance, run the Change Identity Management wizard and restart the instance:

1. Using the Application Server Control Console, navigate to the Instance Home Page for the middle-tier instance.
2. Click **Infrastructure**.
3. On the Infrastructure Page, in the Identity Management section, click **Change**.
4. Follow the tasks in the wizard for supplying the production Identity Management information.
5. When the wizard is finished, navigate to the Instance Home Page and start your instance by clicking **Start All**.

12.4 Case 3: Moving Product-Specific Metadata from Test Metadata Repository to Production Metadata Repository

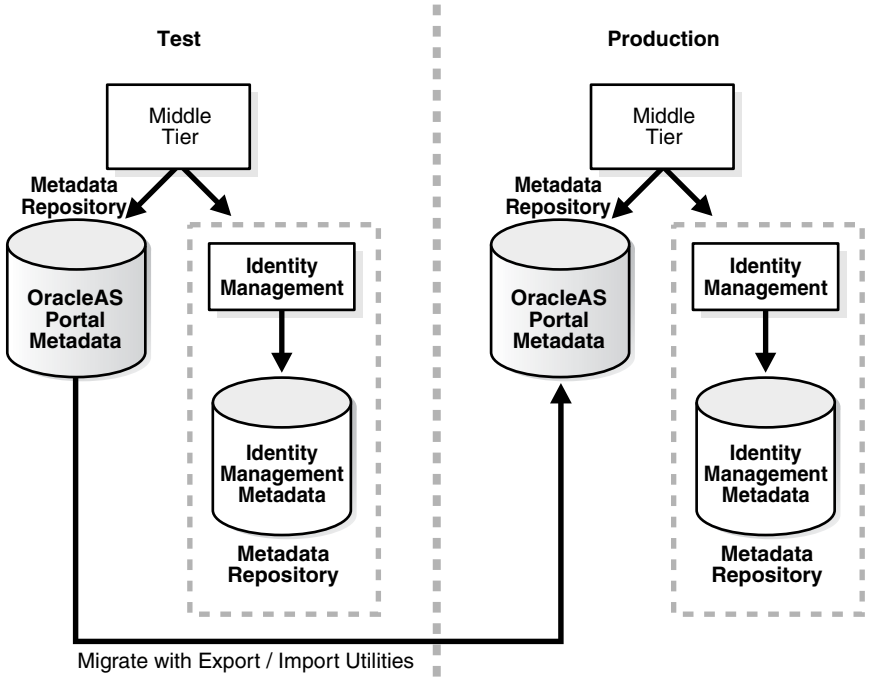
In this use case, you want to only migrate the incremental data that was used for testing applications in a configuration with preexisting test and production environments. How you migrate the data depends upon the application. This section describes how the migration process works for following components:

- OracleAS Portal
- OracleBI Discoverer

12.4.1 OracleAS Portal

In this scenario, you have existing test and production environments with OracleAS Portal Metadata Repositories. You want to move metadata from the test Metadata Repository to the production environment. To move OracleAS Portal metadata, you use the export/import utilities to move content. Figure 12–7 shows this scenario.

Figure 12–7 Moving Test OracleAS Portal Metadata to a Production Environment



12.4.1.1 Preexisting Configuration Assumptions

You have existing test and production environments that each include a middle-tier instance, an Identity Management installation with a Metadata Repository, and an additional Metadata Repository for OracleAS Portal metadata. OracleAS Portal metadata already exists in the test environment.

12.4.1.2 Procedure

To use export and import to move OracleAS Portal metadata:

1. Create transport sets and extract the content to transport tables. Transport sets contain the portal objects that you are planning to export to your target portal environment. This information is displayed in a manifest. The manifest is simply the list of objects in a transport set, used to provide a granular level of control over the export.
2. Move the transport sets from one system (source) to another (target) using Portal export/import command-line scripts to generate a transport set dump file.
3. Transfer the script and dump file to the target system using FTP or other file transfer utilities.
4. Invoke the command line script to import the dump file to the transport tables on your target system.
5. Import the objects from the transport tables to the target portal repository using the Transport Set Manager portlet.

> **See Also:** *Oracle Application Server Portal Configuration Guide* for more information

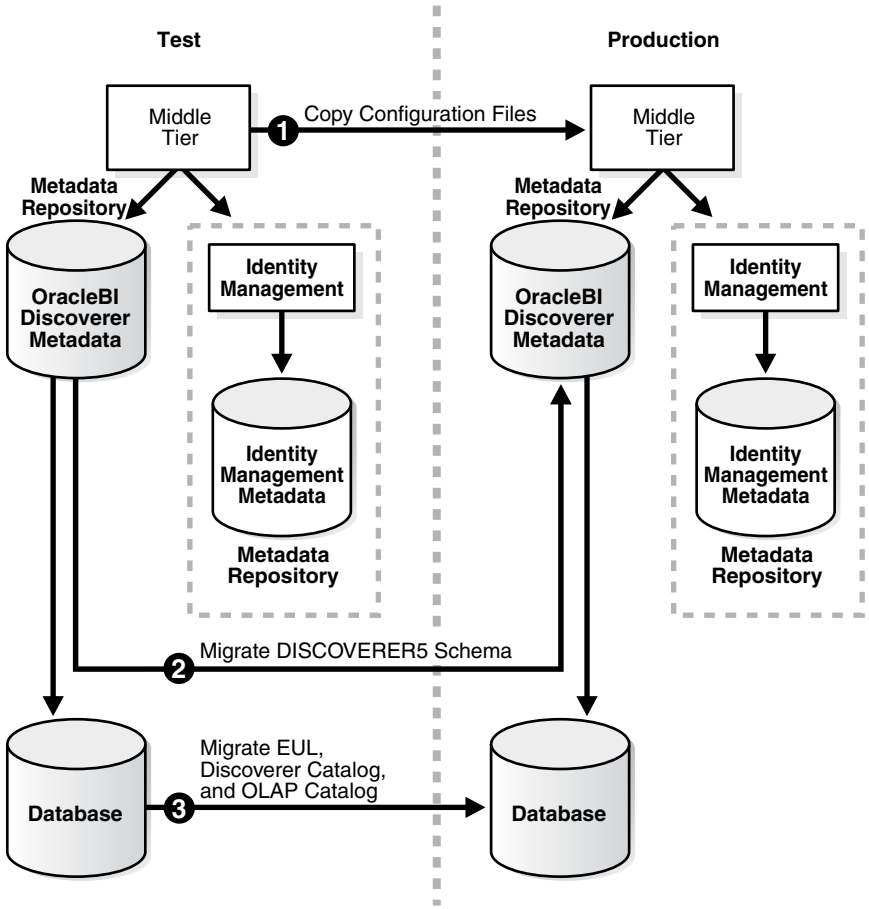
12.4.2 OracleBI Discoverer

In this scenario, you have existing test and production environments. Test OracleBI Discoverer data already resides on the middle tier, Metadata Repository, and a database. You primarily use the test environment to create End User Layers (EULs) for developing a business area without compromising the performance of production systems.

The migration process involves three main tasks. You first move configuration files from the middle tier, and then move the OracleBI Discoverer metadata from the test Metadata Repository to the production environment. Finally, you migrate the OracleBI Discoverer information from the test database to the production database, including the EULs.

Figure 12–8 shows this scenario.

Figure 12–8 Moving Test OracleBI Discoverer Data to a Production Environment



12.4.2.1 Preexisting Configuration Assumptions

You have existing test and production environments that each include a middle-tier instance, an Identity Management installation with a Metadata Repository, and an additional Metadata Repository for OracleBI Discoverer metadata, and a database. OracleAS Portal metadata already exists in the test environment.

12.4.2.2 Procedure

To migrate OracleBI Discoverer configuration, metadata, and database data:

- Task 1: Copy Test Configuration Files
- Task 2: Migrate the Test DISCOVERER5 Schema
- Task 3: Migrate the Test EUL and Catalog Data

Task 1: Copy Test Configuration Files

1. If you have changed the user preferences, copy the `pref.txt` file to the production environment.
2. If you have changed the OracleBI Discoverer settings, copy the `configuration.xml` file to the production environment.
3. If you have changed the server configuration settings, copy the `opmn.xml` file to the production environment.
4. If you have want to use the same database service entries in the production area, copy the `tnsnames.ora` file to the production environment.

Task 2: Migrate the Test DISCOVERER5 Schema

Perform "Move the Test Product Metadata Repository to Production" on page 12-17, ensuring that you also move the `DISCOVERER5` schema, which stores the OracleBI Discoverer metadata.

Task 3: Migrate the Test EUL and Catalog Data

1. Use the Discoverer Administrator to export the EUL schema from the test database, and then import it into the production database.
2. Run the `eul5_id.sql` script to give the new EUL a unique reference number. Having run the `eul5_id.sql` script, you can grant the entire Discoverer end user community access to the EUL.
3. Use the Application Server Control Console to export the Discoverer Catalog from the test database and import it into the production database.
4. Migrate the OLAP Catalog.

See Also:

- *Oracle Business Intelligence Discoverer Configuration Guide* in this documentation library for further information about configuring OracleBI Discoverer
- *Oracle Business Intelligence Discoverer Administration Guide* from the Business Intelligence Tools product CD for further information about migrating EUL data
- *Oracle OLAP Reference Guide* from the Oracle Database documentation library for further information about migrating the OLAP Catalog

Part IV

Secure Sockets Layer (SSL)

This part contains the following chapters:

13

Overview of Secure Sockets Layer (SSL) in Oracle Application Server

In Oracle Application Server, components send requests to and receive responses from other components. These components can be Oracle Application Server components (such as OracleAS Single Sign-On, OracleAS Web Cache, or Oracle HTTP Server) or external clients such as browsers.

To secure these communications, you can configure Oracle Application Server to use SSL, which is an industry standard for securing communications. Oracle Application Server supports SSL versions 2 and 3, as well as TLS version 1.

This chapter provides an overview of SSL and how you can use it with Oracle Application Server. It contains the following topics:

13.1 What SSL Provides

SSL secures communication by providing message encryption, integrity, and authentication. The SSL standard allows the involved components (such as browsers and HTTP servers) to negotiate which encryption, authentication, and integrity mechanisms to use.

- Encryption allows only the intended recipient to read the message. SSL can use different encryption algorithms to encrypt messages. During the SSL handshake that occurs at the start of each SSL session, the client and the server negotiate which algorithm to use. Examples of encryption algorithms supported by SSL include AES, RC4, and 3DES.
- Integrity ensures that a message sent by a client is received intact by the server, untampered. To ensure message integrity, the client hashes the message into a digest using a hash function and sends this **message digest** to the server. The server also hashes the message into a digest and compares the digests. Because SSL uses hash functions that make it computationally infeasible to produce the same digest from two different messages, the server can tell that if the digests do

not match, then someone had tampered with the message. An example of a hash function supported by SSL is SHA1.

- Authentication enables the server and client to check that the other party is who it claims to be. When a client initiates an SSL session, the server typically sends its certificate to the client. Certificates are digital identities that are issued by trusted certificate authorities, such as Verisign. Section 13.5, "Certificates and Oracle Wallets" describes certificates in more detail.

  The client verifies that the server is authentic and not an imposter by validating the certificate chain in the server certificate. The server certificate is guaranteed by the certificate authority (CA) who signed the server certificate.

  The server can also require the client to have a certificate, if the server needs to authenticate the identity of the client.

13.2 About Private and Public Key Cryptography

To provide message integrity, authentication, and encryption, SSL uses both private and public key cryptography.

Private Key Cryptography

Private, or symmetric, key cryptography requires a single, secret key shared by two or more parties to secure communication. This key is used to encrypt and decrypt secure messages sent between the parties. This requires prior and secure distribution of the key to each party. The problem with this method is that it is difficult to securely transmit and store the key.

In SSL, each party calculates the secret key individually using random values known to each side. The parties then send messages encrypted using the secret key.

Public Key Cryptography

Public key cryptography solves this problem by employing public and private key pairs and a secure method for key distribution. The freely available public key is used to encrypt messages that can *only* be decrypted by the holder of the associated private key. The private key is securely stored, together with other security credentials, in an encrypted container such as an Oracle wallet.

Public key algorithms can guarantee the secrecy of a message, but they do not necessarily guarantee secure communication because they do not verify the identities of the communicating parties. To establish secure communication, it is important to verify that the public key used to encrypt a message does in fact belong to the target recipient. Otherwise, a third party can potentially eavesdrop on the communication and intercept public key requests, substituting its own public key for a legitimate key (the **man-in-the-middle** attack).

To avoid such an attack, it is necessary to verify the owner of the public key, a process called authentication. Authentication can be accomplished through a certificate authority (CA), which is a third party trusted by both of the communicating parties.

The CA issues public key certificates that contain an entity's name, public key, and certain other security credentials. Such credentials typically include the CA name, the CA signature, and the certificate effective dates (From Date, To Date).

The CA uses its private key to encrypt a message, while the public key is used to decrypt it, thus verifying that the message was encrypted by the CA. The CA public key is well known, and does not have to be authenticated each time it is accessed. Such CA public keys are stored in wallets.

13.3 How an SSL Session Is Set Up (the "SSL Handshake")

The SSL protocol has two phases: the handshake phase and the data transfer phase. The handshake phase authenticates the server and optionally the client, and establishes the cryptographic keys that will be used to protect the data to be transmitted in the data transfer phase.

When a client requests an SSL connection to a server, the client and server first exchange messages in the handshake phase. (A common scenario is a browser requesting a page using the `https://` (instead of `http://`) protocol from a server. The https protocol indicates the usage of SSL with HTTP.)

Figure 13–1 shows the handshake messages for a typical SSL connection between a Web server and a browser. The following steps are shown in the figure:

1. The client sends a Hello message to the server.

   The message includes a list of algorithms supported by the client and a random number that will be used to generate the keys.

2. The server responds by sending a Hello message to the client. This message includes:

   - The algorithm to use. The server selected this from the list sent by the client.
   - A random number, which will be used to generate the keys.

3. The server sends its certificate to the client.
4. The client authenticates the server using the server's certificate.
5. The client generates a random value ("pre-master secret"), encrypts it using the server's public key, and sends it to the server.
6. The server uses its private key to decrypt the message to retrieve the pre-master secret.
7. The client and server separately calculate the keys that will be used in the SSL session.

   These keys are not sent to each other because the keys are calculated based on the pre-master secret and the random numbers, which are known to each side. The keys include:

   - Encryption key that the client uses to encrypt data before sending it to the server
   - Encryption key that the server uses to encrypt data before sending it to the client
   - Key that the client uses to create a message digest of the data
   - Key that the server uses to create a message digest of the data

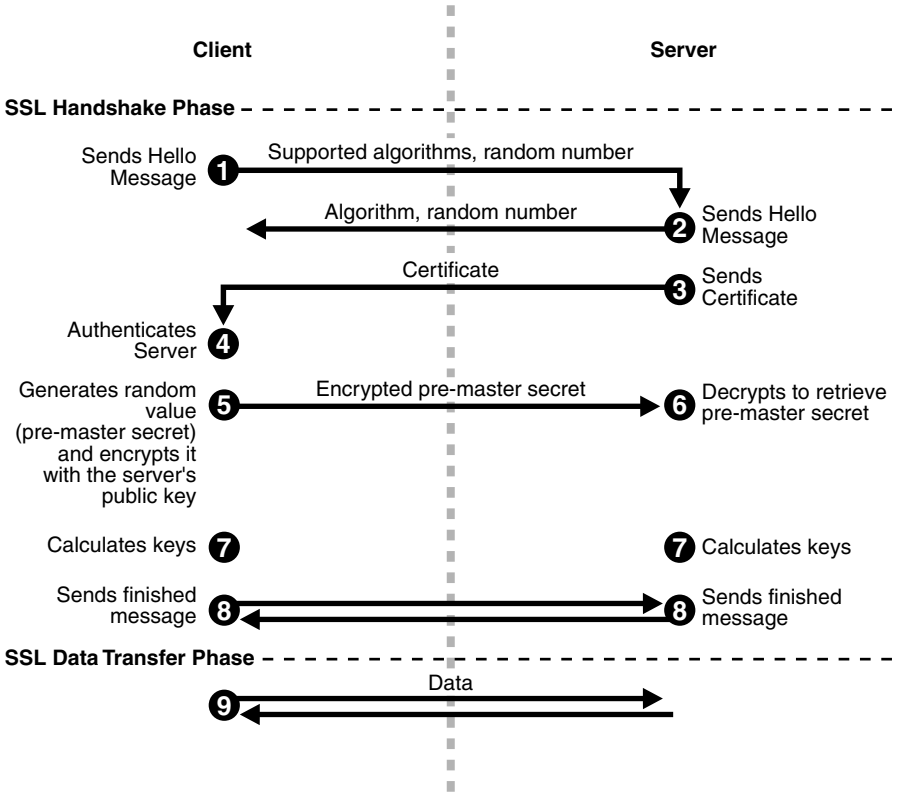
   The encryption keys are symmetric, that is, the same key is used to encrypt and decrypt the data.

8. The client and server send a Finished message to each other. These are the first messages that are sent using the keys generated in the previous step (the first "secure" messages).

   The Finished message includes all the previous handshake messages that each side sent. Each side verifies that the previous messages that it received match the messages included in the Finished message. This checks that the handshake messages were not tampered with.

9. The client and server now transfer data using the encryption and hashing keys and algorithms.

Figure 13–1 SSL Handshake



13.4 Requirements for Using SSL in Oracle Application Server

To use SSL in Oracle Application Server:

- You need a certificate and an Oracle wallet for your site. This certificate is used by clients to verify that they are not connecting to an imposter site.
- If you need to authenticate your clients, the clients will need certificates as well.
- You need to configure Oracle Application Server components (for example, Oracle HTTP Server) to accept and transmit messages over SSL.
- SSL is resource-intensive. If you expect heavy SSL traffic, then you should consider getting an SSL accelerator.

The following sections in this chapter describe these topics in more detail.

13.5 Certificates and Oracle Wallets

You need a certificate for your site. When clients connect to your site requesting SSL communication, you usually have to send your certificate to them so that they can authenticate you.

Oracle Application Server supports X.509 V3 certificates, and certificates that conform to the PKIX standard (RFC 3280).

13.5.1 How to Get a Certificate

You get certificates from certificate authorities (CAs). CAs are trusted entities who sign the certificates that they issue with their private key. Clients can verify the issuer of a certificate (by using the CA's public key). Examples of CAs include Verisign (`http://www.verisign.com`) and Thawte (`http://www.thawte.com`).

Oracle Application Server also has a certificate authority, called OracleAS Certificate Authority (OCA). You can use it to set up your own certificate authority. See the *Oracle Application Server Certificate Authority Administrator's Guide* for details.

To get a certificate, you submit a certificate request to a CA. The certificate request contains your information including your public key. You can use tools to generate a certificate request; these tools can generate private and public key pairs for you. Examples of tools that can generate certificate requests include Oracle Wallet Manager and Sun's `keytool` (for OC4J only). For information on Oracle Wallet Manager, see Chapter 15, "Managing Wallets and Certificates".

Among other items, a certificate includes the following pieces of data:

- Certificate owner's name
- Certificate owner's public key
- CA's name
- Certificate expiration date
- Certificate serial number

Certificates are valid until they expire or until they are revoked.

Note that if you use OracleAS Certificate Authority (OCA) to create certificates for your server, most browsers will not accept these certificates without input from the browser user. This is because most browsers are preconfigured to accept certificates from certain CAs, and OCA is not one of them. The browser will reject the certificate from the server unless the user chooses to accept certificates from the server or import the CA's certificate.

This problem exists for all CAs until the CA's certificate is imported into the browsers. For more information, see the *Oracle Application Server Certificate Authority Administrator's Guide*.

13.5.2 Oracle Wallet

An Oracle wallet is a container that stores your credentials, such as certificates, certificate requests, and private keys. You can store Oracle wallets on the file system or in LDAP directories such as Oracle Internet Directory. Oracle wallets are password-protected.

You manage Oracle wallets using Oracle Wallet Manager. Use Oracle Wallet Manager to perform tasks such as creating Oracle wallets, creating certificate requests, importing certificates into the wallet, and uploading wallets to an LDAP directory.

Oracle Wallet Manager supports PKCS #11 and PKCS #12 wallets:

- Scenario 1: You generate a certificate request using Oracle Wallet Manager and decide to store the private key on the file system. When you get your certificate from the CA, you can import it into an Oracle wallet. This wallet uses the PKCS #12 format. See Section 15.1.4.2.1, "Creating a Standard Wallet" for details.
- Scenario 2: You generate a certificate request using Oracle Wallet Manager and decide to store the private key on a hardware security module. When you get your

certificate from the CA, you can import it into an Oracle wallet. This wallet uses the PKCS #11 format. See Section 15.1.4.2.2, "Creating a Wallet to Store Hardware Security Module Credentials" for details.

- Scenario 3: You already have a certificate in a wallet that uses the PKCS #12 format, and you want to use it with Oracle Application Server. The wallet was created using a third-party tool. In this case, use the tool that was used to create the wallet and export the wallet to a file on the file system. You can then import the wallet. See Section 15.1.5.1.3, "Importing Certificates Created with a Third-Party Tool" for details.

Components that Use Oracle Wallets

Oracle Application Server components that act as SSL servers need Oracle wallets (the wallet already contains the certificate that you want the server to use). Examples of these components include Oracle HTTP Server, OracleAS Web Cache, OPMN, Oracle Internet Directory, and the Port Tunneling daemon (`iaspt`).

You configure the component with the location of the Oracle wallet. For example, to configure Oracle HTTP Server for SSL, you specify the location of the wallet using the `SSLWallet` directive. Refer to the component guide for specific instructions on how to specify the wallet location for the component.

Note: The OC4J component uses a *keystore* instead of an Oracle wallet to store its certificate. You use a tool called `keytool` to import certificates into keystores. See the *Oracle Application Server Containers for J2EE Security Guide* for details on keystores and `keytool`.

13.5.3 Client Certificates

If you need to authenticate your clients, you can configure the Oracle HTTP Server to require clients to send their certificates. Clients can also get their certificates from CAs.

If the clients are Oracle components, for example, OracleAS Web Cache can act as a client when communicating with Oracle HTTP Server, the client component can store its certificate in an Oracle wallet. OPMN also acts as a client when configured for SSL.

If the client is a browser, the client does not need an Oracle wallet. You can just import the certificate into the browser.

Other types of clients, such as SOAP or Web Services clients, have their own ways of configuring certificates and certificate stores.

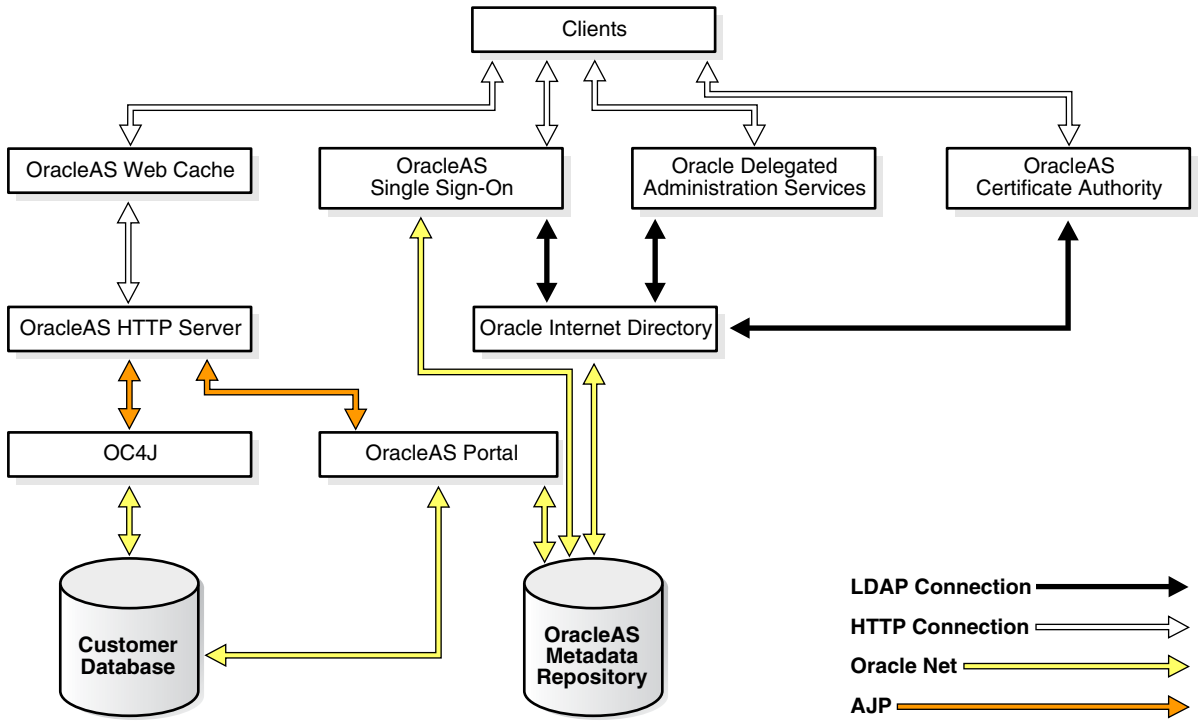
13.6 SSL Configuration Overview

You enable components in Oracle Application Server to use SSL using the Application Server Control. In some cases, you edit configuration files by hand.

SSL secures communication between two parties: a client and a server. If three or more parties are involved, for example, client browser, OracleAS Web Cache, Oracle HTTP Server, and OC4J, then you may have to configure all components to use SSL.

Figure 13–2 shows typical communication paths between Oracle Application Server components and the protocols that they use. For example, browsers use HTTP to communicate with OracleAS Web Cache, and Oracle HTTP Server uses AJP to communicate with OC4J. All these protocols can work with SSL.

Figure 13–2 Communication Paths Between Components in Oracle Application Server



13.6.1 Default SSL Configuration

If you select the default options in the Oracle Application Server installation, none of the components are configured for SSL.

On installation screens where you specify the Oracle Internet Directory host and port, there is an option marked "Use only SSL connections with this Oracle Internet Directory." If you select this option, you need to provide Oracle Internet Directory's SSL port number, and the installer configures the components to use SSL only to communicate with Oracle Internet Directory during runtime.

13.6.2 Partial SSL Configuration

Oracle Application Server enables you to configure SSL for only the paths you want to secure. There are many paths used by components, as shown in Figure 13–2.

You might not want to secure all paths for the following reasons:

- SSL is resource-intensive. If you have heavy SSL traffic, then you probably need to offload SSL processing to an SSL accelerator. See Section 13.7, "Integration with Hardware Security Modules" for details.
- If your computers are behind firewalls, you might need to secure only paths that are accessed by the public. For example, you might need to secure only OracleAS Web Cache and Oracle HTTP Server if the public can access only these components.

13.7 Integration with Hardware Security Modules

When clients connect to your site using SSL, the extra processing required for SSL strains your servers, and your site as a whole (SSL as well as non-SSL connections) will

experience slower performance and throughput. You should consider using SSL accelerator hardware to offload SSL computations and improve performance.

Types of SSL accelerators:

- Section 13.7.1, "Protocol Converters"
- Section 13.7.2, "Mathematics Accelerators (PKCS #11 Integration)"

13.7.1 Protocol Converters

Protocol converters convert HTTPS traffic to HTTP. Protocol converters are standalone hardware machines. Oracle Application Server supports protocol converters from companies such as:

- F5 (http://www.f5.com)
- Cisco (http://www.cisco.com)
- SonicWall (http://www.sonicwall.com)

Note: SSL connections to protocol converters terminate at the protocol converter. When the converters forward the requests to Oracle Application Server, most of them do so in an **unencrypted** fashion.

For the protocol converters that forward the requests to Oracle Application Server using SSL, this is still faster than not using a protocol converter because using a protocol converter eliminates most SSL key exchanges (which is the expensive operation).

13.7.2 Mathematics Accelerators (PKCS #11 Integration)

Mathematics accelerators improve the speed of math operations used by SSL. Such devices are usually plugged into a server (often through TCP/IP). Such devices often have additional features such as key management and secure key stores.

Oracle Application Server supports mathematics accelerators that follow the PKCS #11 standard. For a list of certified accelerators, you can check the Oracle*MetaLink* site, http://www.oracle.com/support/metalink/index.html.

14

Using the SSL Configuration Tool

This chapter describes how to use the SSL Configuration Tool. The following topics are covered:

14.1 Overview

The SSL Configuration Tool is designed to be run after a successful Oracle Application Server installation to automate many of the manual steps currently required for securing HTTP. This means that all Oracle homes you plan to install are successfully installed. If you have a topology where both an OracleAS Infrastructure and middle tier are present, be sure to run the SSL Configuration Tool against the OracleAS Infrastructure first, then the middle tier.

> **Note:** The SSL Configuration Tool is only supported for Oracle Application Server 10*g* Release 2 (10.1.2).

If you install Oracle Application Server and choose to make some configuration changes before running the SSL Configuration Tool, you should run the tool and then refer to the SSL Configuration Tool log files to verify that your changes were not overwritten. The SSL Configuration Tool creates log files in the directory from which the tool is run. A new log file is created each time the tool is run. For these reasons, it is suggested that you create a separate directory from which you can run the SSL Configuration Tool.

If you encounter any problems, you should run the SSL Configuration Tool with the `-rollback` option to revert back to your configuration environment prior to running the tool. See Section 14.6 for information about troubleshooting the SSL Configuration Tool.

The SSL Configuration Tool is available with any Oracle Application Server installation type. OracleAS Infrastructure installations are the only installation type that support SSL configuration during the installation. This option is available on one

of the installation screens. See *Oracle Application Server Installation Guide* for more information.

> **Note:** OracleAS Web Cache is the only standalone type supported by the SSL Configuration Tool. All other standalone types (for example, Apache) are not supported.

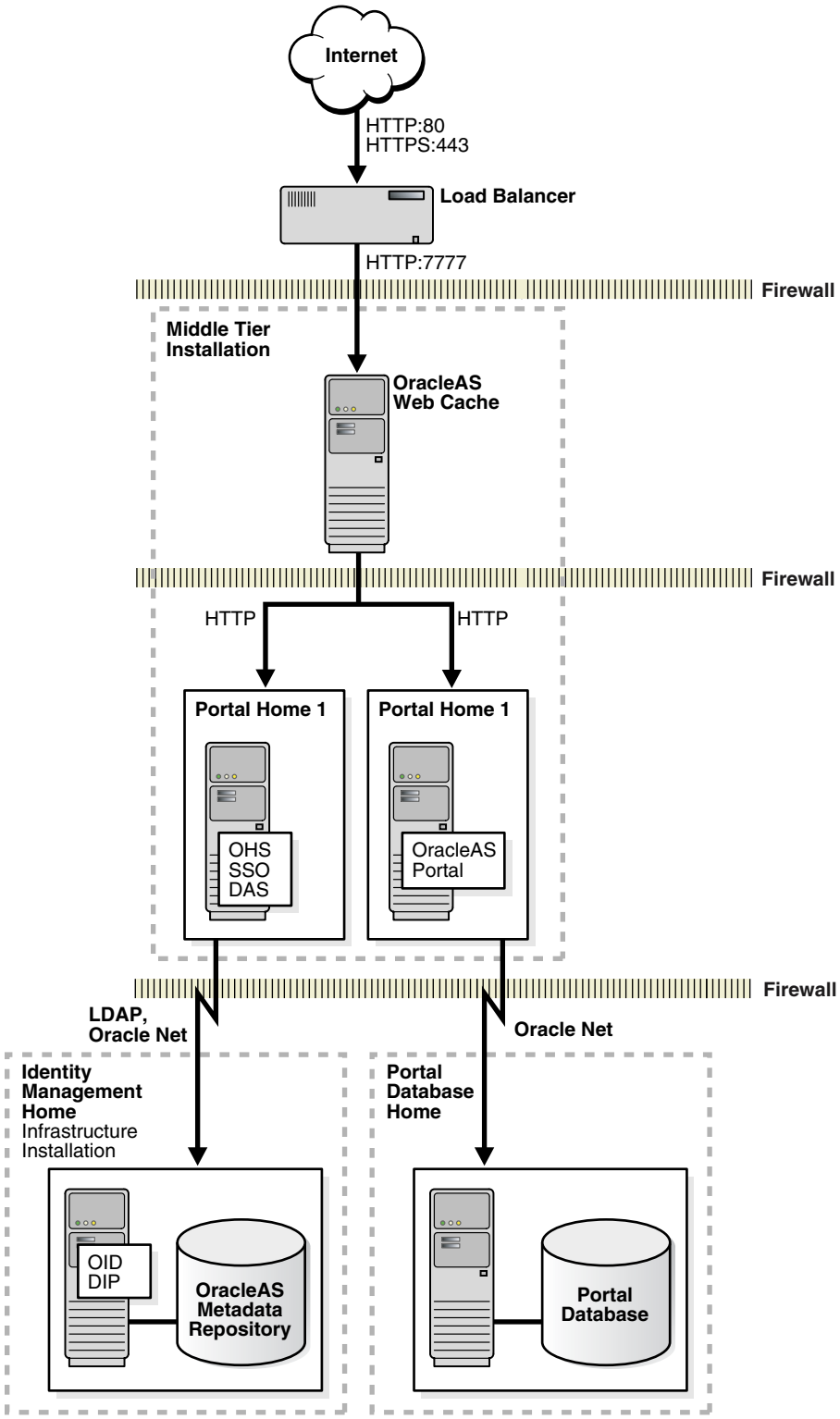
In some cases, the SSL Configuration Tool cannot completely configure SSL for your specific topology. When this occurs, you should refer to the appropriate component documentation for instructions on how to complete your SSL configuration manually. For some links to documentation containing manual steps, see Section 14.5, "Manual Steps".

> **Note:** The SSL Configuration Tool will shut down all necessary components before making any changes. Therefore, you do not need to perform any manual component shutdowns before running the SSL Configuration Tool.

14.2 Understanding SSL Termination

Oracle Application Server ships Oracle HTTP Server (OHS) as the web server software application. It can be configured to serve HTTP requests directly from the Internet. Optionally, OracleAS Web Cache can be installed in front of Oracle HTTP Server to improve performance and scalability. Many customers choose to further increase scalability by putting a load balancer (LBR) in front of both OracleAS Web Cache and Oracle HTTP Server, as illustrated in Figure 14–1.

Figure 14–1 Common Oracle Application Server Topology



HTTPS requests can be allowed to reach the load balancer, OracleAS Web Cache, or Oracle HTTP Server. Any one of these components can be configured as the SSL termination point, while any component before the termination point will be SSL

secured. For example, in Figure 14–1, if OracleAS Web Cache is configured as the termination point, then the load balancer would be SSL secured.

SSL Termination at the Load Balancer

HTTPS requests are terminated at the load balancer in the following scenarios:

- HTTPS Request --> Load Balancer (with SSL accelerator) --> OracleAS Web Cache --> Oracle HTTP Server
- HTTPS Request --> Load Balancer (with SSL accelerator) --> Oracle HTTP Server

With SSL accelerator turned on, HTTPS traffic will terminate at the load balancer, meaning that the load balancer performs decryption and sends plain HTTP traffic to either OracleAS Web Cache or Oracle HTTP Server.

SSL Termination at OracleAS Web Cache

HTTPS requests are terminated at OracleAS Web Cache in the following scenarios:

- HTTPS Requests --> Load Balancer (without SSL accelerator) --> OracleAS Web Cache (with HTTPS termination) --> Oracle HTTP Server
- HTTPS Requests --> OracleAS Web Cache (with HTTPS termination) --> Oracle HTTP Server

The load balancer without SSL accelerator turned on sends HTTPS traffic to OracleAS Web Cache. OracleAS Web Cache, in turn, configured with SSL termination, performs decryption and sends plain HTTP traffic to Oracle HTTP Server.

SSL Termination at Oracle HTTP Server

HTTPS requests are terminated at Oracle HTTP Server in the following scenarios:

- HTTPS Requests --> Load Balancer (without SSL accelerator) --> OracleAS Web Cache (without HTTPS termination) --> Oracle HTTP Server
- HTTPS Requests --> OracleAS Web Cache (without HTTPS termination) --> Oracle HTTP Server
- HTTPS requests --> Load Balancer (without SSL accelerator) --> Oracle HTTP Server

The load balancer without SSL accelerator and OracleAS Web Cache without HTTPS termination will both accept and forward HTTPS requests. When these requests reach Oracle HTTP Server with SSL termination configured, Oracle HTTP Server will perform decryption and send plain HTTP traffic to other Oracle Application Server components.

14.3 Command Line Interface

This section describes how to use the `SSLConfigTool` command. It contains the following sections:

- Where Can I Find the SSL Configuration Tool?
- Syntax
- Configuration File for Silent Mode
- Default Wallet Locations

14.3.1 Where Can I Find the SSL Configuration Tool?

The SSLConfigTool executable is located in the *ORACLE_HOME*/bin directory.

14.3.2 Syntax

The SSLConfigTool command is used as follows:

```
SSLConfigTool ( -config_w_prompt
              | -config_w_file input_file_name
              | -config_w_default
              | -rollback )
              [-dry_run]
              [-wc_for_infra]
              [-secure_admin]
              [-opwd orcladmin_pwd]
              [-ptl_dad dad_name]
              [-ptl_inv_pwd ptl_inv_pwd]
```

Table 14–1 describes the command line options for the SSLConfigTool command.

Table 14–1 SSL Configuration Tool Command Line Options

Parameter	Description
-config_w_prompt	Run in interactive mode.
-config_w_file *input_file_name*	Run in silent mode using the values specified in the *input_file_name* file. This input file should be an XML file. For more information, see Section 14.3.3, "Configuration File for Silent Mode".
-config_w_default	Run in silent mode using the values specified in the portlist.ini and ias.properties files.
-rollback	Revert to the prior state before the command was last run. SSO registration will be done using virtual host and port.
-dry_run	Print the steps without implementing them.
-wc_for_infra	Forces an OracleAS Web Cache to be used as a load balancer for an infrastructure environment.
-secure_admin	Secure the OracleAS Web Cache and Enterprise Manager administration ports (the ports used to display Application Server Control Console).
-opwd *orcladmin_pwd*	Set the Oracle administrator password. This parameter is required.
-ptl_dad *dad-name*	Set the Portal dad name. If no name is specified, the default "portal" will be used.
-ptl_inv_pwd *ptl_inv_pwd*	Set the Portal invalidation password used to send invalidation to OracleAS Web Cache. This parameter is required if you installed OracleAS Portal. If you are running SSLConfigTool with the -rollback parameter, this parameter is not required.

Note that the -config_w_prompt, -config_w_file, -config_w_default, and -rollback parameters are mutually exclusive; only one can be used with the SSLConfigTool command.

If you choose to run the tool interactively with the -config_w_prompt parameter, you will be prompted for the appropriate information one question at a time.

If you choose to run the tool silently by specifying a configuration file with the `-config_w_file` parameter, you should read Section 14.3.3, "Configuration File for Silent Mode" for information about constructing a valid input file.

14.3.3 Configuration File for Silent Mode

If you run `SSLConfigTool` in silent mode, you must provide an input file describing the components in the deployment topology.

The input file contains two main sections, `<mid_tier>` and `<infra>`, inside the `<sslconfig>` element:

```
<sslconfig>
   <mid_tier>
      ...
   </mid_tier>
   <infra>
      ...
   </infra>
</sslconfig>
```

The `<mid_tier>` and `<infra>` elements contain information the SSL Configuration Tool needs to know about this Oracle Application Server instance. The content inside both the `<mid_tier>` and `<infra>` elements must look like this:

```
<virtual_address ssl="on|off"
                 host="..."
                 port="..."
                 inv_port="..."
                 ssl_terminate="lbr|wc|ohs" />
<lbr loopback_port="..." />
<wc wallet="..." os_wallet="..." />
<ohs wallet="...">
   <servers>
      <server host="..." port="..." />
   </servers>
</ohs>
```

Each element is described in further detail in the remainder of this section. All elements and attributes have default values assigned in either the `portlist.ini` or `ias.properties` file.

<virtual_address> Element

This is a required element used to describe this virtual host. Its attributes are summarized in Table 14–2.

Table 14–2 Attributes for the <virtual_address> Element

Attribute	Description
`ssl`	Required attribute. Sets whether SSL is on or off. Valid values are: 1. `on` (enable HTTPS) 2. `off` (enable HTTP)
`host`	Required attribute. Virtual host name.
`port`	Required attribute. Virtual host port number.

Table 14–2 (Cont.) Attributes for the <virtual_address> Element

Attribute	Description
`inv_port`	Optional attribute. Specify the OracleAS Web Cache invalidation port. This attribute is only relevant for OracleAS Portal installations.
`ssl_terminate`	Required attribute when the `ssl` value is "on." Sets the SSL termination point. All components up to the specified termination point will be secured in SSL. Valid values are: ■ `lbr` (load balancer) ■ `wc` (OracleAS Web Cache) ■ `ohs` (Oracle HTTP Server) For more information, see Section 14.2, "Understanding SSL Termination".

<lbr> Element

This element is required if there is a load balancer present in your topology. The `<lbr>` element takes one attribute, `loopback_port`, which is used to specify the loopback port number.

<wc> Element

This element is required if there is an OracleAS Web Cache present in your topology. The `<wc>` element takes the following optional attributes:

- `wallet`—Location of the OracleAS Web Cache front-end wallet (used to communicate with an external browser).
- `os_wallet`—Location of the OracleAS Web Cache back-end wallet (used to communicate with Oracle HTTP Server).
- `loopback_port`—Loopback port number when there is no load balancer present in your topology and OracleAS Web Cache is acting as the load balancer. This attribute is valid for OracleAS Portal installations only.

See Section 14.3.4 for default wallet locations.

<ohs> Element

This element is used in conjunction with the `<wc>` element; if there is no OracleAS Web Cache present in your topology, then this element is not needed. In addition, this element is also not needed if OracleAS Web Cache and Oracle HTTP Server are installed on the same machine, and OracleAS Web Cache maps to the Oracle HTTP Server in the same Oracle home.

The `<ohs>` element takes one optional attribute, `wallet`, which is used to specify the location of the Oracle HTTP Server (Apache) wallet. See Section 14.3.4 for default wallet locations.

Additionally, the `<ohs>` element requires one `<servers>` element for each Oracle HTTP Server in the topology. The `<servers>` element takes the following attributes:

- `host`—Name of the Oracle HTTP Server.
- `port`—Oracle HTTP Server listen port.

There must be one `<servers>` element for each Oracle HTTP Server in your topology.

14.3.4 Default Wallet Locations

Default wallet locations are listed in Table 14–3.

Table 14–3 Default Wallet Locations

Wallet	Default Location (File Path)
Web Cache Front-End Wallet	*ORACLE_HOME*/webcache/wallets/default
Web Cache Back-End Wallet	*ORACLE_HOME*/webcache/wallets/default
Apache Wallet	*ORACLE_HOME*/Apache/Apache/conf/ssl.wlt/default

To specify your own wallet location, you must supply either the absolute path from the root (/) directory or a relative path from the Oracle home directory.

The following are some examples of an absolute path:

```
/etc/wallets/absolute/path/to/my/companys/wallet (UNIX)
C:\product\OracleAS\10.1.2\absolute\path\to\my\companys\wallet (Windows)
```

The following are some examples of a relative path:

```
%ORACLE_HOME%/relative/path/to/my/companys/wallet (UNIX)
%ORACLE_HOME%\relative\path\to\my\companys\wallet (Windows)
```

14.4 Common SSL Configuration Scenarios

This section describes how to use the SSL Configuration Tool for the following common topologies:

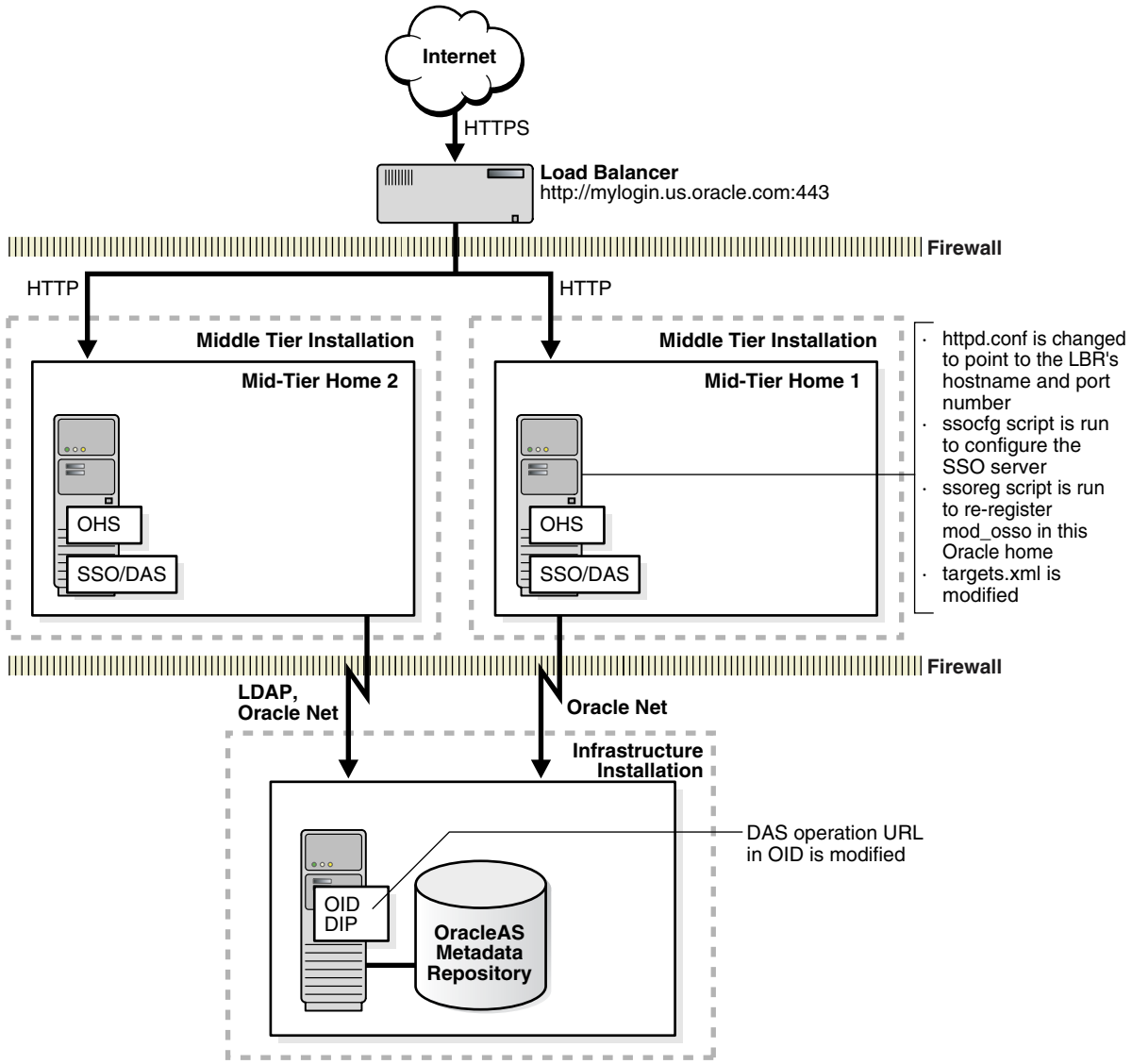
- Configuring SSL to Load Balancer for OracleAS Single Sign-On/Oracle Delegated Administration Services
- Configuring SSL to Load Balancer for OracleAS Portal
- Configuring SSL to Oracle HTTP Server for Oracle HTTP Server/Oracle Application Server Containers for J2EE
- Configuring SSL to OracleAS Web Cache for J2EE
- Configuring SSL to Oracle HTTP Server for OracleAS Single Sign-On/Oracle Delegated Administration Services
- Configuring SSL to Oracle HTTP Server for OracleAS Portal
- Configuring an HTTP Instance
- Configuring SSL for Cluster Configurations

14.4.1 Configuring SSL to Load Balancer for OracleAS Single Sign-On/Oracle Delegated Administration Services

This configuration enables SSL at the load balancer for OracleAS Single Sign-On (SSO)/Oracle Delegated Administration Services (DAS). The load balancer acts as the front end for the SSO server (Figure 14–2).

SSL terminates at load balancer, meaning that the load balancer performs decryption and sends plain HTTP traffic directly to Oracle HTTP Server for infrastructure installations.

Figure 14–2 Topology and Summary of Changes



14.4.1.1 What it Does

The SSL Configuration Tool performs the following to enable HTTPS:

- Change the `httpd.conf` file to refer to the load balancer's host and port.
- Run the `ssocfg` script to configure the SSO server.
- Run the `ssoreg` script to re-register `mod_osso` in the current `ORACLE_HOME`.
- Modify the DAS operation URL in the Oracle Internet Directory.
- Modify the `targets.xml` file.

If you have multiple SSO/DAS homes in a high availability environment, you must run `SSLConfigTool` in each home, then perform step number 6 in the list of manual steps in Section 14.5.

14.4.1.2 Running the SSL Configuration Tool

Run the following command to configure SSL for this scenario. The name of the input configuration file is `sslct_config.xml` and the Oracle administrator password is "welcome1."

```
SSLConfigTool -config_w_file sslct_config.xml -opwd welcome1
```

The following are the contents of the `sslct_config.xml` input configuration file:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="on"
                       host="mylogin.us.oracle.com"
                       port="443"
                       ssl_terminate="lbr" />
   </mid_tier>
</sslconfig>
```

To configure SSL interactively, use the `-config_w_prompt` option, as shown in the following example. The answers to the questions are in **bold**:

```
SSLConfigTool -config_w_prompt -opwd welcome1

Welcome to the OracleAS SSL Configuration Tool.
Below you will be guided with a series of questions.
If a question has the default answer,
the answer will be enclosed inside [square brackets].
Let's start now...

Do you want to configure your site to accept browser requests using SSL protocol?
[y]: y
What is the virtual host name for your site? [mylogin.us.oracle.com]:
mylogin.us.oracle.com
What is the virtual port number for your site? [4443]: 443
Does your site have an external load balancer (LBR)?
Note: Do NOT include OracleAS Web Cache as LBR here. [y]: y
Does your site have OracleAS Web Cache? [y]: n
Does your Oracle HTTP Server (OHS) accept requests in SSL protocol? [y]: n
You have supplied all the information. Are you ready to continue? [y]: y
```

14.4.1.3 For More Information

For detailed information about configuring SSL in this scenario, see:

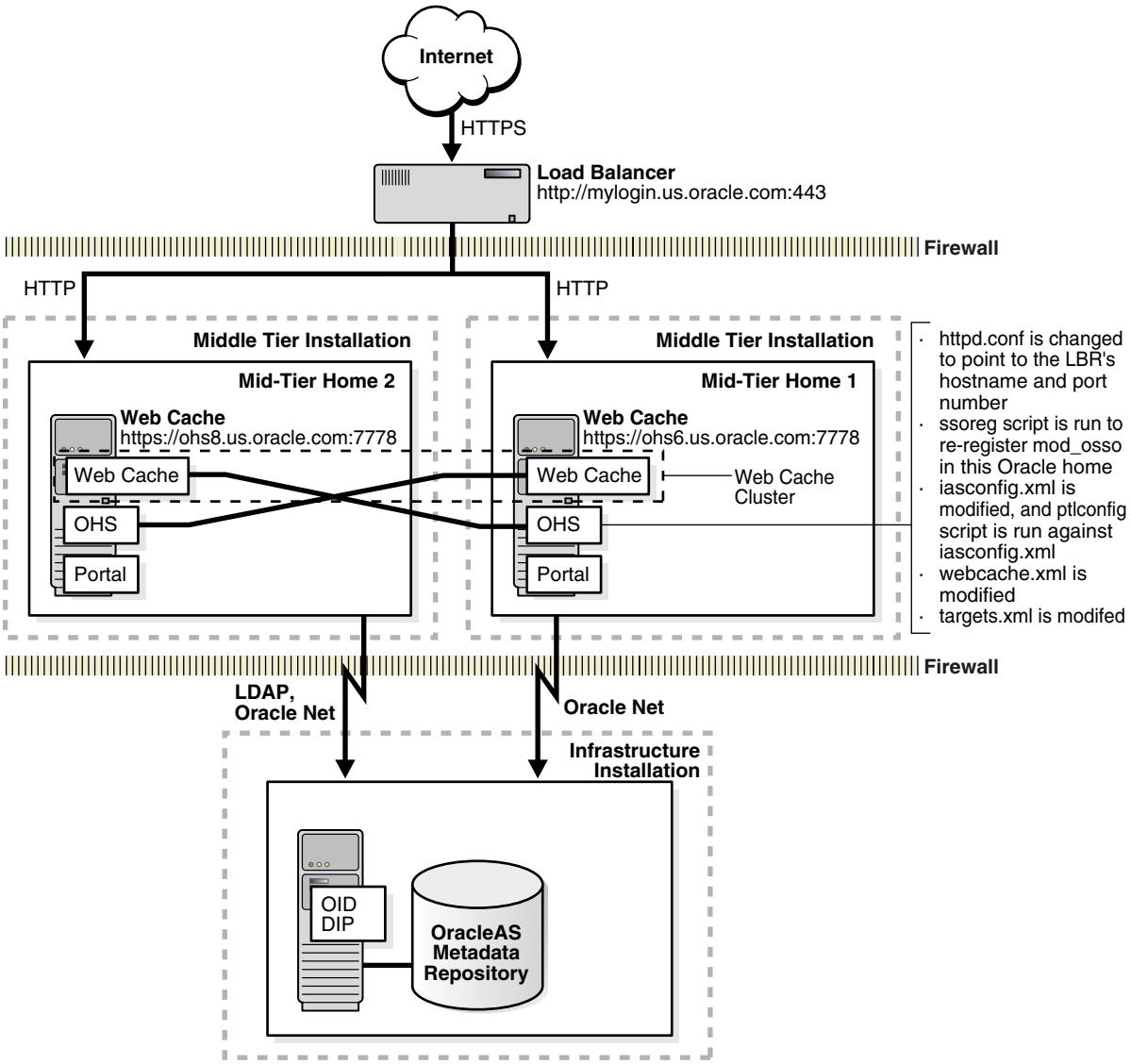
- "Chapter 7, Enabling SSL" in the *Oracle Application Server Single Sign-On Administrator's Guide*.
- "Appendix E, Enabling SSL and PKI on SSO" in the *Oracle Application Server Certificate Authority Administrator's Guide*.

14.4.2 Configuring SSL to Load Balancer for OracleAS Portal

This configuration enables SSL for Portal. A load balancer acts as the front end for two Portal middle tiers, each with its own OracleAS Web Cache and HTTP Server instances in the same `ORACLE_HOME`. In addition, the two OracleAS Web Caches have been manually clustered together (Figure 14–3).

SSL terminates at load balancer, meaning that the load balancer performs decryption and sends plain HTTP traffic directly to OracleAS Web Cache and then to Oracle HTTP Server.

Figure 14–3 Topology and Summary of Changes



14.4.2.1 What it Does

The SSL Configuration Tool performs the following to enable HTTPS:

- Change the `httpd.conf` file to refer to the load balancer's host and port.
- Run the `ssoreg` script to re-register `mod_osso` in the current `ORACLE_HOME`.
- Modify the `iasconfig.xml` file and then run the `ptlconfig` script against it.
- Modify the `webcache.xml` file to create a new site, do the proper site-to-server mappings, and point to any user-specified wallet locations.
- Modify the `targets.xml` file.

If you have multiple OracleAS Portal instances, you must run `SSLConfigTool` for each Portal instance, then perform steps 2, 3, 4, and 6 in the list of manual steps in Section 14.5. If you are using OracleAS Wireless with OracleAS Portal, you must also perform step 5.

14.4.2.2 Running the SSL Configuration Tool

Run the following command to configure SSL for this scenario. The name of the input configuration file is `sslct_config.xml`, the Oracle administrator password is "welcome1," and the Portal invalidation password is also "welcome1."

```
SSLConfigTool -config_w_file sslct_config.xml -opwd welcome1 -ptl_inv_pwd welcome1
```

The following shows the contents of the `sslct_config.xml` input configuration file:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="on"
                       host="mylogin.us.oracle.com"
                       port="443"
                       inv_port="4001"
                       ssl_terminate="lbr" />
   <lbr loopback_port="7780" />
   <wc/>
   <ohs>
      <servers>
         <server host="ohs6.us.oracle.com" port="7778" />
         <server host="ohs8.us.oracle.com" port="7778" />
      </servers>
   </ohs>
   </mid_tier>
</sslconfig>
```

To configure SSL interactively, use the `-config_w_prompt` option, as shown in the following example. The answers to the questions are in **bold**:

```
SSLConfigTool -config_w_prompt -opwd welcome1 -ptl_inv_pwd welcome1

Welcome to the OracleAS SSL Configuration Tool.
Below you will be guided with a series of questions.
If a question has the default answer,
the answer will be enclosed inside [square brackets].
Let's start now...

Do you want to configure your site to accept browser requests using SSL protocol?
[y]: y
What is the virtual host name for your site? [mylogin.us.oracle.com]:
mylogin.us.oracle.com
What is the virtual port number for your site? [4443]: 443
What is the invalidation port number your Portal uses? [4001]: 4001
Does your site have an external load balancer (LBR)?
Note: Do NOT include OracleAS Web Cache as LBR here. [y]: y
Does your site have OracleAS Web Cache? [y]: y
Does your Web Cache accept requests in SSL protocol? [y]: n
Does your Oracle HTTP Server (OHS) accept requests in SSL protocol? [y]: n
What is the Portal loop-back port in LBR or Web Cache? [7780]: 7780
How many OHS instances does your Web cache route traffic to? [1]: 2
Please enter host name for OHS #1: ohs6.us.oracle.com
Please enter port number for OHS #1: 7778
Please enter host name for OHS #2: ohs8.us.oracle.com
Please enter port number for OHS #2: 7778
You have supplied all the information. Are you ready to continue? [y]: y
```

14.4.2.3 For More Information

For detailed information about configuring SSL in this scenario, see:

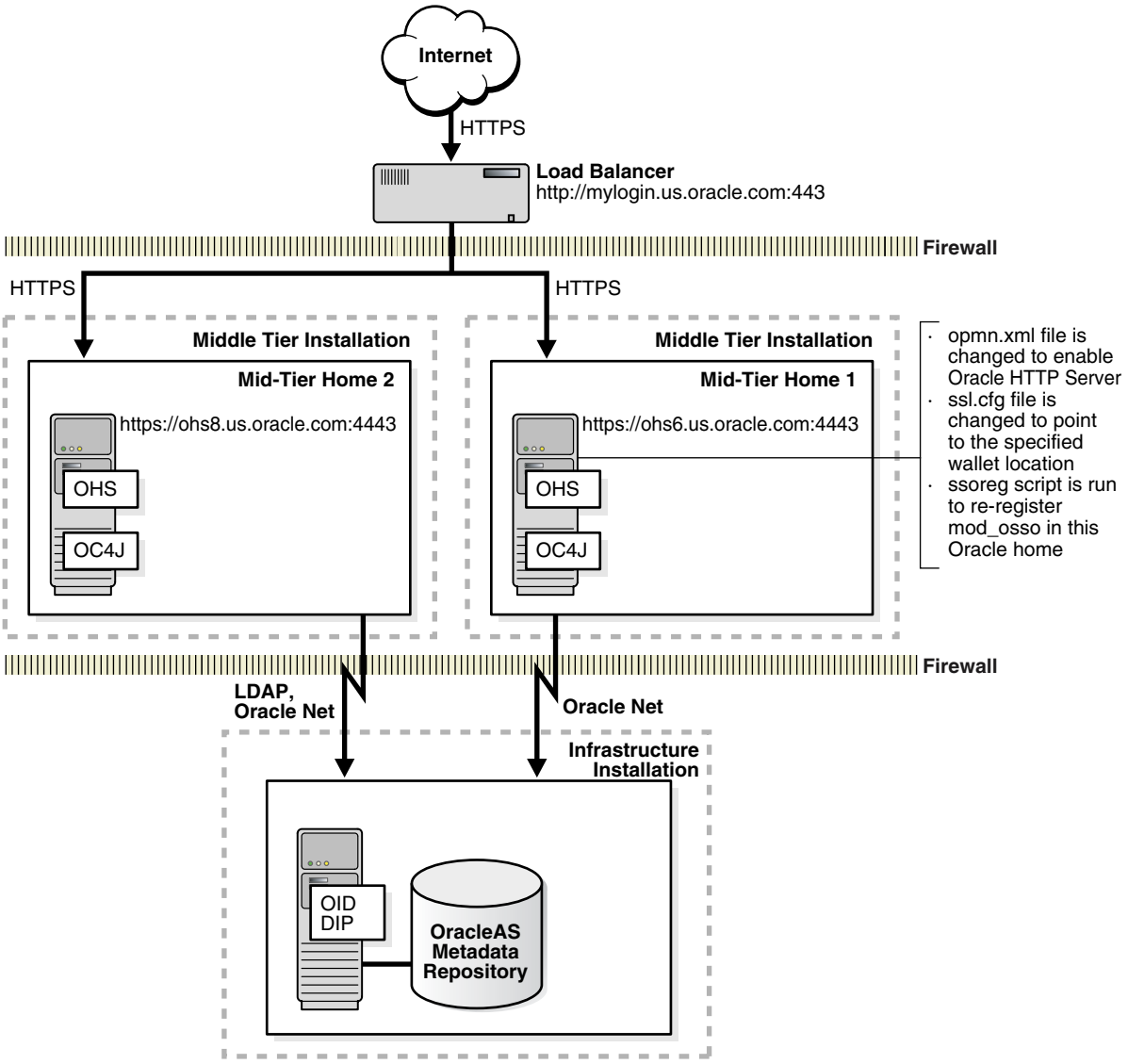
- "Section 1.3, A Standard Enterprise Deployment for Portal Applications: myPortalCompany.com" in the *Oracle Application Server Enterprise Deployment Guide*.
- "Chapter 4, Configuring the Application Infrastructure for myPortalCompany.com" in the *Oracle Application Server Enterprise Deployment Guide*.

14.4.3 Configuring SSL to Oracle HTTP Server for Oracle HTTP Server/Oracle Application Server Containers for J2EE

This configuration enables SSL for OHS/OC4J. Two Oracle HTTP Servers are configured in a high availability environment and both are configured to accept HTTPS requests from a front-end load balancer (Figure 14–4).

SSL terminates at Oracle HTTP Server, meaning that Oracle HTTP Server performs decryption and sends plain HTTP traffic directly to other Oracle Application Server components.

Figure 14–4 Topology and Summary of Changes



14.4.3.1 What it Does

The SSL Configuration Tool does the following to enable HTTPS:

- Change the `opmn.xml` file to enable Oracle HTTP Server.
- Change the `ssl.conf` file to point to the specified wallet location.
- Run the `ssoreg` script to re-register `mod_osso` in the current `ORACLE_HOME` if SSO is enabled.

For multiple Oracle Application Server Containers for J2EE (OC4J) homes in a high availability environment, you must run `SSLConfigTool` in each home, then perform step number 6 in the list of manual steps in Section 14.5 if OC4J is protected by OracleAS Single Sign-On.

14.4.3.2 Running the SSL Configuration Tool

Run the following command to configure SSL for this scenario. The name of the input configuration file is `sslct_config.xml` and the Oracle administrator password is "welcome1."

```
SSLConfigTool -config_w_file sslct_config.xml -opwd welcome1
```

The following shows the contents of the `sslct_config.xml` input configuration file:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="on"
                       host="mylogin.us.oracle.com"
                       port="443"
                       ssl_terminate="ohs" />
   </mid_tier>
</sslconfig>
```

To configure SSL interactively, use the `-config_w_prompt` option, as shown in the following example. The answers to the questions are in **bold**:

```
SSLConfigTool -config_w_prompt -opwd welcome1

Welcome to the OracleAS SSL Configuration Tool.
Below you will be guided with a series of questions.
If a question has the default answer,
the answer will be enclosed inside [square brackets].
Let's start now...

Do you want to configure your site to accept browser requests using SSL protocol?
[y]: y
What is the virtual host name for your site? [mylogin.us.oracle.com]:
mylogin.us.oracle.com
What is the virtual port number for your site? [4443]: 443
Does your site have an external load balancer (LBR)?
Note: Do NOT include OracleAS Web Cache as LBR here. [y]: y
Does your site have OracleAS Web Cache? [y]: n
Do you want to supply your own wallet location for OHS? [n]: n
You have supplied all the information. Are you ready to continue? [y]: y
```

14.4.3.3 For More Information

For detailed information about configuring SSL in this scenario, see:

- "Section 1.2, A Standard Enterprise Deployment for J2EE Applications: myJ2EECompany.com" in the *Oracle Application Server Enterprise Deployment Guide*.

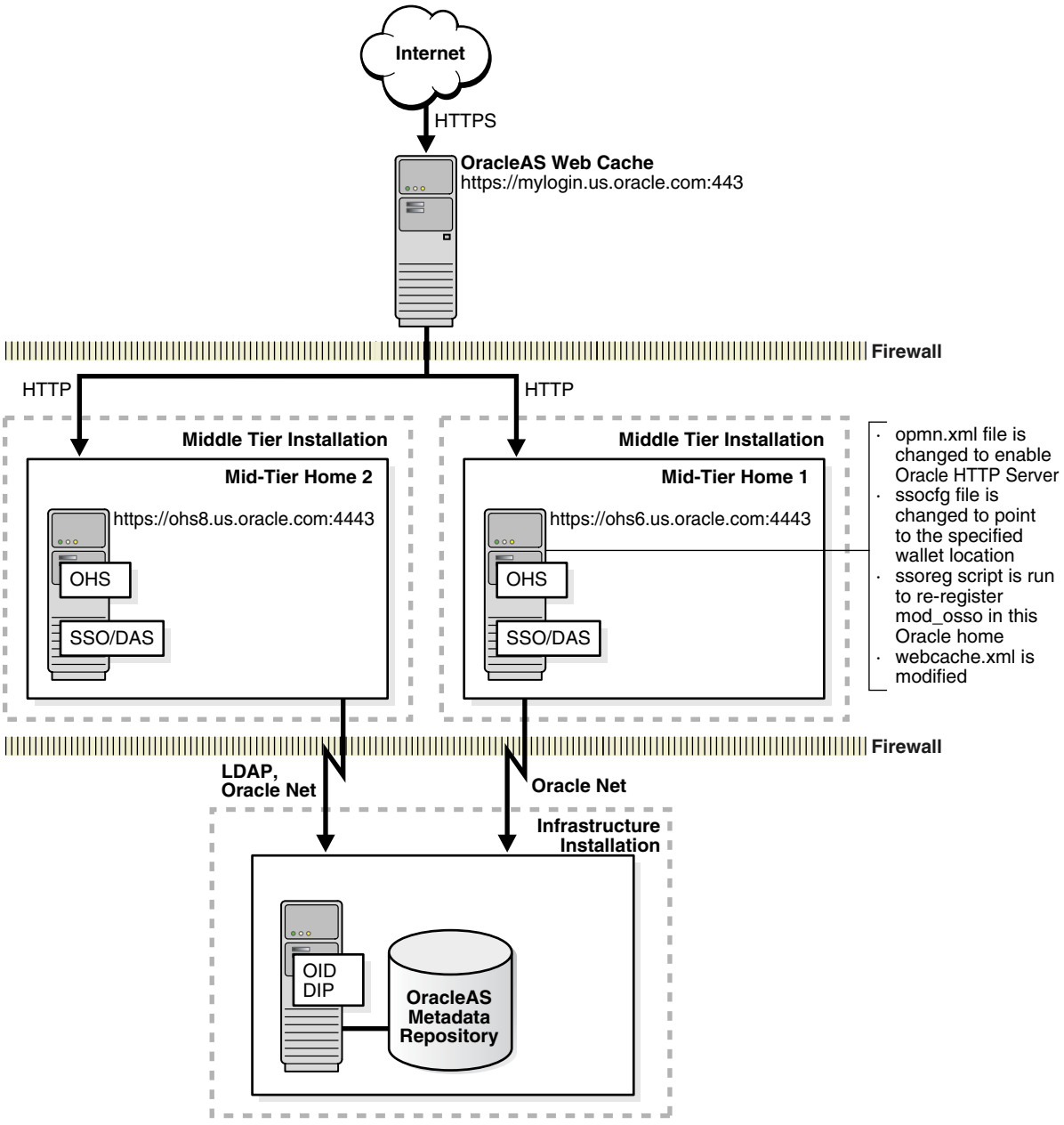
- "Chapter 3, Configuring the Application Infrastructure for myJ2EECompany.com" in the *Oracle Application Server Enterprise Deployment Guide*.

14.4.4 Configuring SSL to OracleAS Web Cache for J2EE

This scenario enables SSL for J2EE in smaller Oracle Application Server shops, where there is no load balancer and OracleAS Web Cache is used as the load balancer (Figure 14–5).

SSL terminates at OracleAS Web Cache, meaning that OracleAS Web Cache performs decryption and sends plain HTTP traffic directly to Oracle HTTP Server. Alternatively, if OracleAS Web Cache is not used, then SSL would terminate at Oracle HTTP Server.

Figure 14–5 Topology and Summary of Changes



14.4.4.1 What it Does

The SSL Configuration Tool does the following to enable HTTPS:

- Change the `opmn.xml` file to enable Oracle HTTP Server.
- Change the `ssl.conf` file to point to the load balancer's host and port.
- Run the `ssoreg` script to re-register `mod_osso` in the current `ORACLE_HOME` if SSO is enabled.
- Modify the `webcache.xml` file to define the new site, origin server, listen port, do the proper site-to-server mappings, and point to any user-specified wallet locations.

14.4.4.2 Running the SSL Configuration Tool

Run the following command to configure SSL for this scenario. The name of the input configuration file is `sslct_config.xml` and the Oracle administrator password is "welcome1."

```
SSLConfigTool -config_w_file sslct_config.xml -opwd welcome1
```

The following shows the contents of the `sslct_config.xml` input configuration file:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="on"
                       host="mylogin.us.oracle.com"
                       port="443"
                       ssl_terminate="wc" />
   <wc/>
   <ohs>
      <servers>
         <server host="ohs6.us.oracle.com" port="7778" />
         <server host="ohs8.us.oracle.com" port="7778" />
      </servers>
   </ohs>
   </mid_tier>
</sslconfig>
```

To configure SSL interactively, use the `-config_w_prompt` option, as shown in the following example. The answers to the questions are in **bold**:

```
SSLConfigTool -config_w_prompt -opwd welcome1

Welcome to the OracleAS SSL Configuration Tool.
Below you will be guided with a series of questions.
If a question has the default answer,
the answer will be enclosed inside [square brackets].
Let's start now...

Do you want to configure your site to accept browser requests using SSL protocol?
[y]: y
What is the virtual host name for your site? [mylogin.us.oracle.com]:
mylogin.us.oracle.com
What is the virtual port number for your site? [4443]: 443
Does your site have an external load balancer (LBR)?
Note: Do NOT include OracleAS Web Cache as LBR here. [y]: n
Does your Oracle HTTP Server (OHS) accept requests in SSL protocol? [y]: n
Do you want to supply your own wallet locations for Web Cache? [n]: n
How many OHS instances does your Web cache route traffic to? [1]: 2
Please enter host name for OHS #1: ohs6.us.oracle.com
```

```
Please enter port number for OHS #1: 4443
Please enter host name for OHS #2: ohs8.us.oracle.com
Please enter port number for OHS #2: 4443
You have supplied all the information. Are you ready to continue? [y]: y
```

14.4.4.3 For More Information

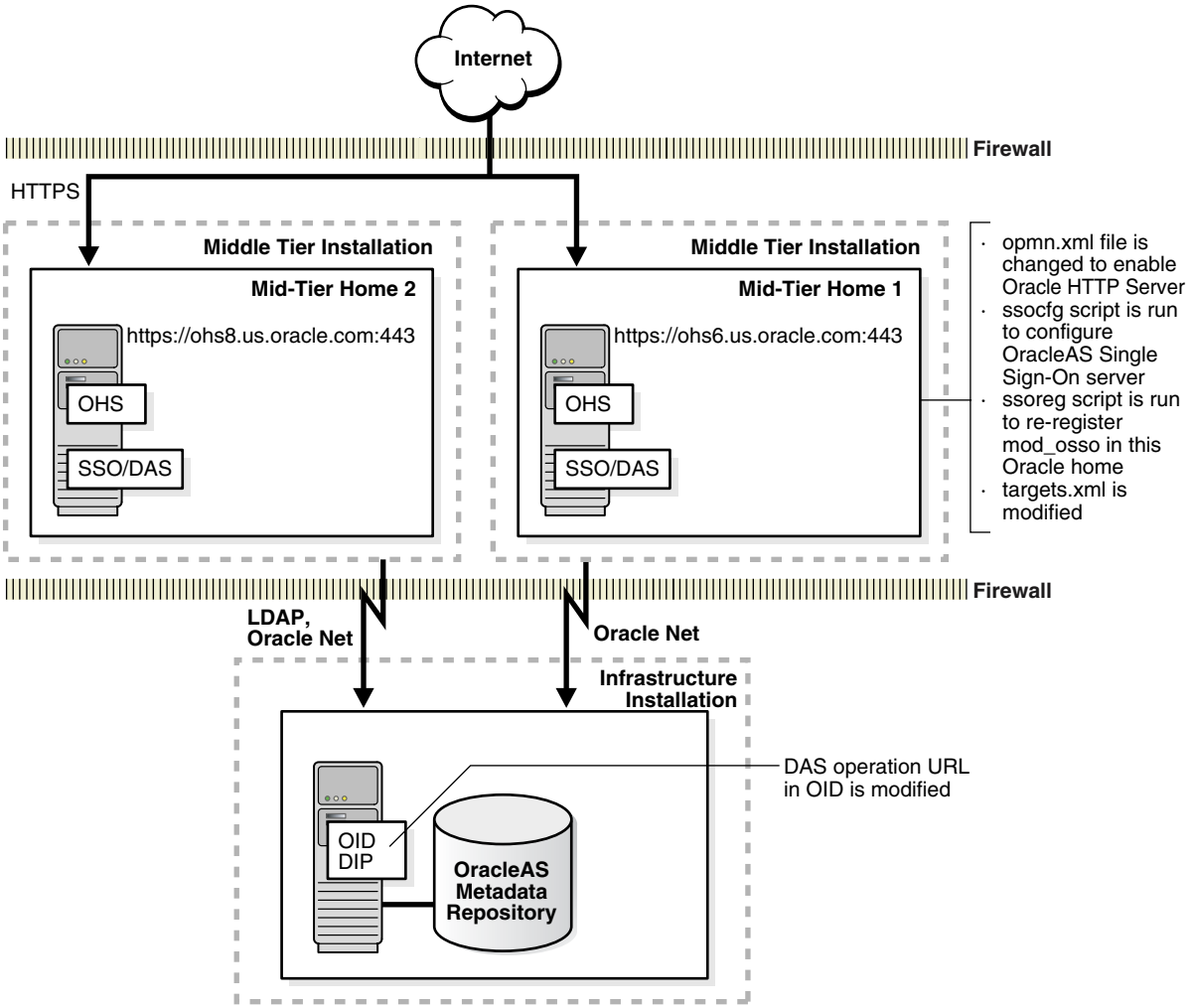
For detailed information about configuring SSL in this scenario, see "Chapter 9, Configuring OracleAS Web Cache for HTTPS Requests" in *Oracle Application Server Web Cache Administrator's Guide*.

14.4.5 Configuring SSL to Oracle HTTP Server for OracleAS Single Sign-On/Oracle Delegated Administration Services

This scenario enables SSL for SSO/DAS with Oracle HTTP Server acting as the front end. There is no load balancer or front-end OracleAS Web Cache in this scenario (Figure 14–6).

SSL terminates at the Oracle HTTP Server tier. This means Oracle HTTP Server performs decryption and sends plain HTTP traffic directly to other Oracle Application Server components.

Figure 14–6 Topology and Summary of Changes



14.4.5.1 What it Does

The SSL Configuration Tool does the following to enable HTTPS:

- Change the `opmn.xml` file to enable Oracle HTTP Server.
- Change the `ssl.conf` file to point to the load balancer's host and port.
- Run the `ssocfg` script to configure the OracleAS Single Sign-On Server.
- Run the `ssoreg` script to re-register `mod_osso` in the current `ORACLE_HOME` if SSO is enabled.
- Modify the Oracle Delegated Administration Services operation URL in Oracle Identity Management and the `ssl.conf` file to add rewrite directives.
- Modify the `targets.xml` file.

If you have multiple SSO/DAS homes in a high availability environment, you must run `SSLConfigTool` in each home, then perform step number 6 in the list of manual steps in Section 14.5.

14.4.5.2 Running the SSL Configuration Tool

Run the following command to configure SSL for this scenario. The name of the input configuration file is `sslct_config.xml` and the Oracle administrator password is "welcome1."

```
SSLConfigTool -config_w_file sslct_config.xml -opwd welcome1
```

The following shows the contents of the `sslct_config.xml` input configuration file:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="on"
                       host="ohs6.us.oracle.com"
                       port="443"
                       ssl_terminate="ohs" />
   </mid_tier>
</sslconfig>
```

To configure SSL interactively, use the `-config_w_prompt` option, as shown in the following example. The answers to the questions are in **bold**:

```
SSLConfigTool -config_w_prompt -opwd welcome1

Welcome to the OracleAS SSL Configuration Tool.
Below you will be guided with a series of questions.
If a question has the default answer,
the answer will be enclosed inside [square brackets].
Let's start now...

Do you want to configure your site to accept browser requests using SSL protocol?
[y]: y
What is the virtual host name for your site? [mylogin.us.oracle.com]:
ohs6.us.oracle.com
What is the virtual port number for your site? [4443]: 443
Does your site have an external load balancer (LBR)?
Note: Do NOT include OracleAS Web Cache as LBR here. [y]: n
Does your site have OracleAS Web Cache? [y]: n
Do you want to supply your own wallet location for OHS? [n]: n
You have supplied all the information. Are you ready to continue? [y]: y
```

14.4.5.3 For More Information

For detailed information about configuring SSL in this scenario, see:

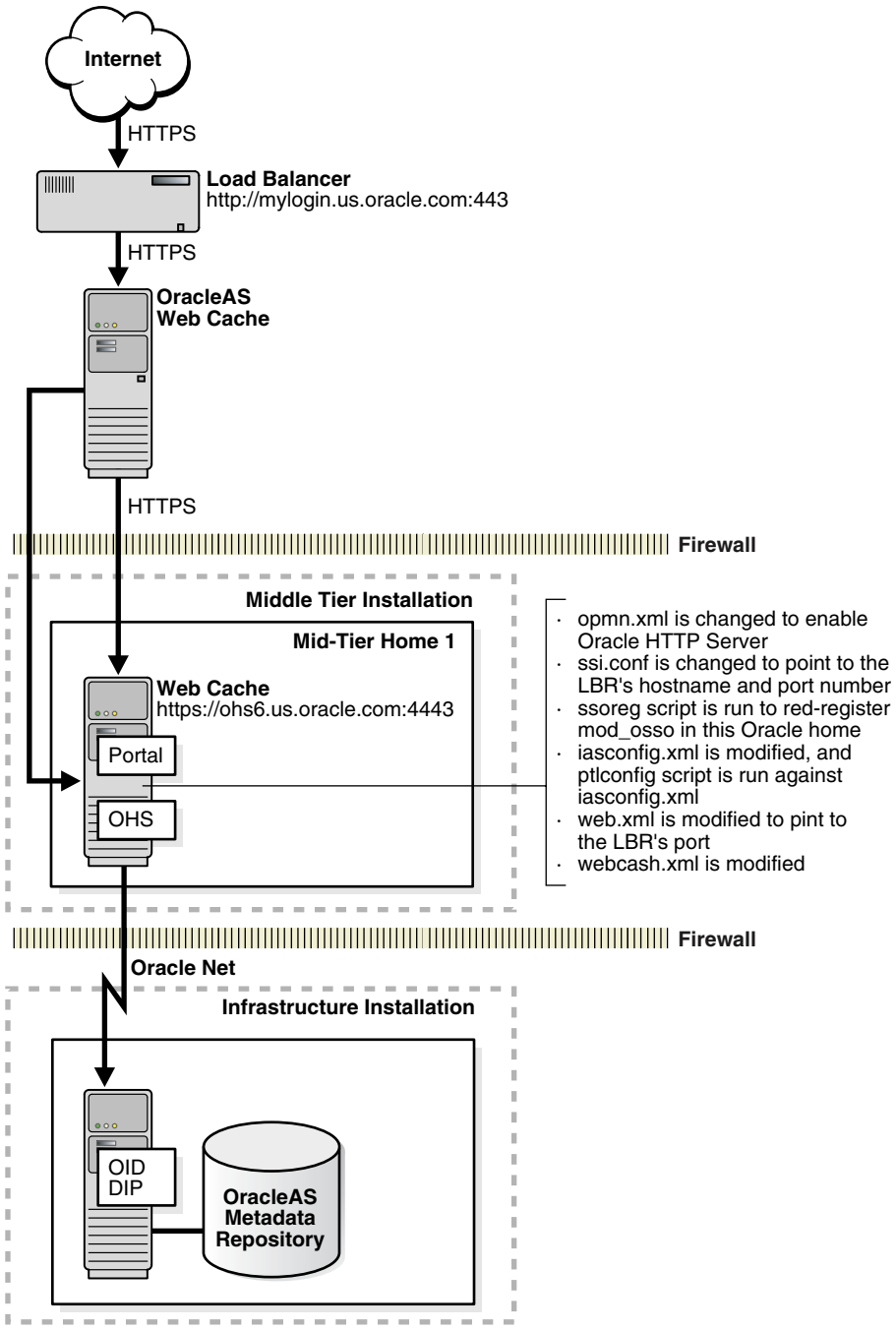
- "Chapter 7, Enabling SSL" in the *Oracle Application Server Single Sign-On Administrator's Guide*.
- "Appendix E, Enabling SSL and PKI on SSO" in the *Oracle Application Server Certificate Authority Administrator's Guide*.

14.4.6 Configuring SSL to Oracle HTTP Server for OracleAS Portal

This scenario enables SSL for OracleAS Portal. A load balancer and OracleAS Web Cache are both in front of OracleAS Portal. The OracleAS Web Cache is a standalone installation, and uses the Oracle HTTP Server in the middle tier. The OracleAS Web Cache from the OracleAS Portal installation is rendered inactive (Figure 14–7).

For the purposes of this configuration, you should specify Oracle HTTP Server as the point of SSL termination. HTTPS, however, is used throughout OracleAS Portal.

Figure 14–7 Topology and Summary of Changes



14.4.6.1 What it Does

The SSL Configuration Tool does the following to enable HTTPS:

- Change the `opmn.xml` file to enable Oracle HTTP Server.
- Change the `ssl.conf` file to point to the load balancer's host and port.
- Run the `ssoreg` script to re-register `mod_osso` in the current `ORACLE_HOME` if SSO is enabled.
- Modify the `iasconfig.xml` file and then run the `ptlscript` script against it.

- Modify the `web.xml` file to refer to the load balancer's port.
- Modify the `webcache.xml` file to define the new site, origin server, listen port, do the proper site-to-server mappings, and point to any user-specified wallet locations if OracleAS Web Cache is installed.

If you have multiple OracleAS Portal instances, you must run `SSLConfigTool` for each Portal instance, then perform steps 2, 3, 4, and 6 in the list of manual steps in Section 14.5. If you are using OracleAS Wireless with OracleAS Portal, you must also perform step 5.

14.4.6.2 Running the SSL Configuration Tool

Run the following command to configure SSL for this scenario. The name of the input configuration file is `sslct_config.xml`, the Oracle administrator password is "welcome1," and the Portal invalidation password is also "welcome1."

```
SSLConfigTool -config_w_file sslct_config.xml -opwd welcome1 -ptl_inv_pwd welcome1
```

The following shows the contents of the `sslct_config.xml` input configuration file:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="on"
                       host="mylogin.us.oracle.com"
                       port="443"
                       inv_port="4001"
                       ssl_terminate="ohs" />
   <lbr loopback_port="7780" />
   <wc/>
   <ohs>
      <servers>
         <server host="machine_6.us.oracle.com" port="4443" />
      </servers>
   </ohs>
   </mid_tier>
</sslconfig>
```

To configure SSL interactively, use the `-config_w_prompt` option, as shown in the following example. The answers to the questions are in **bold**:

```
SSLConfigTool -config_w_prompt -opwd welcome1 -ptl_inv_pwd welcome1

Welcome to the OracleAS SSL Configuration Tool.
Below you will be guided with a series of questions.
If a question has the default answer,
the answer will be enclosed inside [square brackets].
Let's start now...

Do you want to configure your site to accept browser requests using SSL protocol?
[y]: y
What is the virtual host name for your site? [mylogin.us.oracle.com]:
mylogin.us.oracle.com
What is the virtual port number for your site? [4443]: 443
What is the invalidation port number your Portal uses? [4001]: 4001
Does your site have an external load balancer (LBR)?
Note: Do NOT include OracleAS Web Cache as LBR here. [y]: y
Does your Web Cache accept requests in SSL protocol? [y]: y
Does your Oracle HTTP Server (OHS) accept requests in SSL protocol? [y]: y
What is the Portal loop-back port in LBR or Web Cache? [7780]: 7780
Do you want to supply your own wallet locations for Web Cache? [n]: n
```

```
Do you want to supply your own wallet location for OHS? [n]: n
How many OHS instances does your Web Cache route traffic to? [1]: 1
What is the host name for OHS? [ohs6.us.oracle.com]: ohs6.us.oracle.com
What is the port number for OHS? [4443]: 4443
You have supplied all the information. Are you ready to continue? [y]: y
```

14.4.6.3 For More Information

For detailed information about configuring SSL in this scenario, see "Section 6.3.2.1, Configuring SSL for OracleAS Portal" in the *Oracle Application Server Portal Configuration Guide*.

14.4.7 Configuring an HTTP Instance

The SSL Configuration Tool can also be used to configure an HTTP-only instance. To accomplish this using a configuration input file, set the `ssl` attribute of the `<virtual_address>` element to "off," as shown in the following example:

```
<sslconfig>
   <mid_tier>
      <virtual_address ssl="off"
                       host="mylogin.us.oracle.com"
                       port="80" />
   </mid_tier>
</sslconfig>
```

To configure HTTP using the SSL Configuration Tool in interactive mode, answer "n" to the following question:

```
Do you want to configure your site to accept browser requests using SSL protocol?
[y]: n
```

14.4.8 Configuring SSL for Cluster Configurations

To configure SSL in a clustered environment (for example, clustered Identity Management or Oracle Application Server High Availability Solutions), perform the following steps:

1. Add the first node and HTTPS listener port to the new load balancer configuration. You should refer to your load balancer documentation for information on how to do this.
2. Run the SSL Configuration Tool on the first node to enable HTTPS.
3. Add the second node and HTTPS listener port to the load balancer configuration. You should refer to your load balancer documentation for information on how to do this.
4. Run the SSL Configuration Tool on the second node to enable HTTPS.

14.5 Manual Steps

After the SSL Configuration Tool has run, you will see the following message:

```
The tool has completed the configuration steps. But please keep in
mind that there are still some manual steps left for you to
perform before you can use the environment in SSL mode.
1) The tool has not done anything about the SSL certificates. You
need to make sure you use a real certificate inside the wallets.
2) The tool has done limited configuration about EM monitoring. E.g.
you need to import the root CA certificate into EM wallet, etc.
```

```
You need to follow the instructions in the documentation.
3) You may need to cluster Web Cache on your own.
4) You may need to manually upload the Portal Preference Store to
the database using Portal scripts.
5) You may need to configure Wireless following the instructions
in the documentation.
6) When you have multiple installations of the same type, you need
to manually copy osso.conf from one install to the rest as well as
run 'ssotransfer' command against them.
For more, please refer to the documentation.
```

The corresponding documentation for these tasks can be found as listed in the following:

1. See Chapter 15 for information about certificates and wallets.
2. See Section 16.3.7, "Configuring SSL for Oracle Enterprise Manager 10g" for details about how to enable SSL communication in Oracle Enterprise Manager 10*g*.

   For information about configuring a certificate in Oracle Enterprise Manager 10*g*, refer to the section titled "Enable Monitoring of the Load Balancing Router's OracleAS Portal Host and Port Settings" in the *Oracle Application Server Enterprise Deployment Guide*.
3. See "Chapter 3, Cache Clustering" in the *Oracle Application Server Web Cache Administrator's Guide* for information about OracleAS Web Cache clusters.
4. See *PDK-Java Release Notes*, available on Portal Center at `http://portalcenter.oracle.com`, for information about how to perform this procedure.
5. See *Oracle Application Server Wireless Administrator's Guide* for information about configuring Oracle Application Server Wireless. Additional wireless configuration information can be found in "Section C.8, Using the cfgiasw Script to Configure Mobile Settings" in *Oracle Application Server Portal Configuration Guide*.
6. See "Section 4.3.2.4, Reregistering mod_osso on the Partner Application Middle Tiers" in the *Oracle Application Server Single Sign-On Administrator's Guide* for an example of how to do this procedure.

14.6 Troubleshooting the SSL Configuration Tool

This section contains information that may help you in the event you encounter any problems while running the SSL Configuration Tool. The following topics are covered:

- General Troubleshooting Procedure
- Oracle Application Server Wireless Requires Manual Changes
- Configuring Seeded Providers for OracleAS Portal
- SSL Configuration Tool Does Not Support IASCONFIG_LOC Environment Variable
- SSL Configuration Tool Does Not Modify sso_apache.conf File
- SSL Configuration Tool Does Not Modify opmn.xml Parameters

14.6.1 General Troubleshooting Procedure

If the SSL Configuration Tool is able to detect a specific error, it will print some instructions on the screen and then exit. You should follow these instructions and take the recommended actions listed. For example:

```
Executing command:
/scratch/testuser/product/10.1.3/OracleAS/opmn/bin/opmnctl stopproc
ias-component=dcm-daemon

ERROR: failed to run '/scratch/testuser/product/10.1.3/OracleAS/opmn/bin/opmnctl'.
ACTION: Please refer to the log file for the previous command.
ACTION: You may try running it explicitly from the command line to get more
information about the error.
```

If the SSL Configuration Tool hangs while it is running, you should press CTRL-C to exit. On the screen, you should see a series of commands that have been executed up to the point where you had to exit. Start with the most recent command and work backwards, consulting the documentation as necessary to determine the nature of the problem.

In either case, whether the SSL Configuration Tool exists or hangs, you should make the changes and run the `SSLConfigTool` command again. If the problems persist, you should run the `SSLConfigTool -rollback` command to revert to the environment prior to running the `SSLConfigTool` command.

Note: you do not need to run the `SSLConfigTool -rollback` command if you do not see this line upon execution (the directories in this example may differ from your own):

```
Configuring HTTPS for your ORACLE_HOME at:
/private/iasinst/work/ssltool_A
Backing up file '/private/iasinst/work/ssltool_
A/opmn/conf/opmn.xml' to file '/p
rivate/iasinst/work/ssltool_A/opmn/conf/opmn.xml.orig_
SSLConfigTool'
```

This is because no files on your system are changed prior to this point in the execution of the `SSLConfigTool` command.

If you encounter failures during the execution of the `SSLConfigTool -rollback` command, you must resolve the issues and run the `SSLConfigTool -rollback` again. This procedure must be repeated until you have a successful rollback. If you are unable to rollback successfully, contact your Oracle support representative for assistance.

14.6.2 Oracle Application Server Wireless Requires Manual Changes

Modifying OracleAS Web Cache settings (for example, changing the Listener port) can affect the OracleAS Portal URL. Rather than pointing to `https`, the URL will begin with `http`. To fix this, you must manually update your OracleAS Wireless settings. Refer to the following sections in *Oracle Application Server Portal Configuration Guide* for details:

- "Section 6.3.2.1.3, SSL to OracleAS Web Cache"
- "Section C.8, Using the cfgiasw Script to Configure Mobile Settings"

14.6.3 Configuring Seeded Providers for OracleAS Portal

OracleAS Portal includes several default (seeded) user accounts and groups. The SSL Configuration Tool is not able to configure SSL for seeded user accounts and groups; you must enable SSL for them manually. For the manual steps, see "Section 5.6.4, Configure Seeded Providers and Locally Hosted Web Providers" in *Oracle Application Server Portal Configuration Guide*.

14.6.4 SSL Configuration Tool Does Not Support IASCONFIG_LOC Environment Variable

The `IASCONFIG_LOC` environment variable is used to override the default location of the `iasconfig.xml` file (the Portal Dependency Settings file). The SSL Configuration Tool does not modify the `IASCONFIG_LOC` environment variable, which means only the `iasconfig.xml` file in the default location (*`ORACLE_HOME`*`/portal/conf`) will be updated during SSL configuration.

For more information about the `IASCONFIG_LOC` environment variable, see "Appendix A, Using the Portal Dependency Settings Tool and File" in *Oracle Application Server Portal Configuration Guide*.

14.6.5 SSL Configuration Tool Does Not Modify sso_apache.conf File

The SSL Configuration Tool does not modify the `sso_apache.conf` file. This file needs to be modified in order for external applications to work in an SSL environment.

For information about modifying the `sso_apache.conf` file, see "Section 8.1.3, Protect Single Sign-On URLs" in *Oracle Application Server Single Sign-On Administrator's Guide*.

14.6.6 SSL Configuration Tool Does Not Modify opmn.xml Parameters

If you install Oracle Business Intelligence, Oracle Business Intelligence Discoverer adds the following parameters to the `opmn.xml` file:

- `oracle.discoverer.applications.host`
- `oracle.discoverer.applications.port`

By default, the `oracle.discoverer.applications.host` parameter is set to the local host, and the `oracle.discoverer.applications.port` parameter is set to the Apache non-SSL port.

The SSL Configuration Tool does not modify these parameters, so you have to modify them manually after running the SSL Configuration Tool:

- Change `oracle.discoverer.applications.host` to point to your LBR's virtual IP address (if you are using an LBR).
- Change `oracle.discoverer.applications.port` to point to your SSL port.

After making these changes, save `opmn.xml`, then run the following commands:

```
opmnctl reload
opmnctl restartproc type=oc4j instancename=OC4J_BI_FORMS
```

15

Managing Wallets and Certificates

This chapter explains how to obtain and manage security credentials for Oracle Application Server resources. Security administrators can use Oracle Wallet Manager and its command-line utility, `orapki`, to manage public key infrastructure (PKI) credentials on Oracle clients and servers. These tools create credentials that can be read by Oracle Database, Oracle Application Server 10g, and the Oracle Identity Management infrastructure.

This chapter contains the following topics:

- Using Oracle Wallet Manager
- Performing Certificate Validation and CRL Management with the orapki Utility
- Interoperability with X.509 Certificates

Note: If you already have certificates provisioned, the following sections may provide all of the information you need:

Section 15.1.2, "Starting Oracle Wallet Manager"

Section 15.3, "Interoperability with X.509 Certificates"

15.1 Using Oracle Wallet Manager

This section describes Oracle Wallet Manager, a GUI tool used to manage PKI certificates. It contains the following topics:

- Oracle Wallet Manager Overview
- Starting Oracle Wallet Manager
- How to Create a Complete Wallet: Process Overview
- Managing Wallets
- Managing Certificates

15.1.1 Oracle Wallet Manager Overview

Oracle Wallet Manager is an application used to manage and edit security credentials in Oracle wallets. A wallet is a password-protected container that stores authentication and signing credentials, including private keys, certificates, and trusted certificates, all of which are used by SSL for strong authentication. You can use Oracle Wallet Manager to perform the following tasks:

- Create wallets

- Generate certificate requests
- Open wallets to access PKI-based services
- Save credentials to hardware security modules by using APIs which comply to Public-Key Cryptography Standard #11 specification (see PKCS #11)
- Upload wallets to and download them from an LDAP directory
- Import third-party PKCS #12-format wallets to use in an Oracle environment
- Export Oracle wallets to third-party environments

The following topics describe Oracle Wallet Manager features:

- Wallet Password Management
- Strong Wallet Encryption
- Microsoft Windows Registry Wallet Storage
- Backward Compatibility
- Third-Party Wallet Support
- LDAP Directory Support

15.1.1.1 Wallet Password Management

Oracle wallets are password protected. Oracle Wallet Manager includes an enhanced wallet password management module that enforces the following password management policy guidelines:

- Minimum password length (8 characters)
- Maximum password length unlimited
- Alphanumeric character mix required

15.1.1.2 Strong Wallet Encryption

Oracle Wallet Manager stores private keys associated with X.509 certificates and uses Triple-DES encryption.

15.1.1.3 Microsoft Windows Registry Wallet Storage

As an option, Oracle Wallet Manager enables you to store multiple Oracle wallets in the user profile area of the Microsoft Windows system registry or in a Windows file management system. Storing your wallets in the registry provides the following benefits:

- **Better Access Control.** Wallets stored in the user profile area of the registry are only accessible by the associated user. User access controls for the system thus become, by extension, access controls for the wallets. In addition, when a user logs out of a system, access to that user's wallets is effectively precluded.
- **Easier Administration.** Since wallets are associated with specific user profiles, no file permissions need to be managed, and the wallets stored in the profile are automatically deleted when the user profile is deleted. Oracle Wallet Manager can be used to create and manage the wallets in the registry.

15.1.1.3.1 Options Supported:

- Open wallet from the registry
- Save wallet to the registry

- Save As to a different registry location
- Delete wallet from the registry
- Open wallet from the file system and save it to the registry
- Open wallet from the registry and save it to the file system

15.1.1.4 Backward Compatibility

Oracle Wallet Manager is backward-compatible to Release 8.1.7 of the database.

15.1.1.5 Third-Party Wallet Support

Oracle Wallet Manager can use PKI credentials from the following third-party applications:

- Microsoft Internet Explorer 5.0 and later
- Netscape Communicator 4.7.2 and later
- OpenSSL

Browser PKI credential stores (those from Microsoft Internet Explorer and Netscape) hold user certificates, which contain the subject's public key and identifying information, and their associated trusted certificates. To use these credentials, you must export them from the third-party environment and save them in PKCS #12 format. Then you can use Oracle Wallet Manager to open them for use with SSL.

> **See Also:** "Section 15.1.5.1.3, "Importing Certificates Created with a Third-Party Tool"

15.1.1.6 LDAP Directory Support

Oracle Wallet Manager can upload wallets to and retrieve them from an LDAP-compliant directory. Storing wallets in a centralized LDAP-compliant directory lets users access them from multiple locations or devices, ensuring consistent and reliable user authentication while providing centralized wallet management throughout the wallet life cycle. To prevent accidental over-write of functional wallets, only wallets containing an installed certificate can be uploaded.

Directory user entries must be defined and configured in the LDAP directory before Oracle Wallet Manager can be used to upload or download wallets for a user. If a directory contains Oracle8*i* (or prior) users, they are automatically upgraded to use the wallet upload and download feature on first use.

Oracle Wallet Manager downloads a user wallet by using a simple password-based connection to the LDAP directory. However, for uploads it uses an SSL connection if the open wallet contains a certificate with SSL Oracle PKI certificate usage. If an SSL certificate is not present in the wallet, password-based authentication is used.

Note: The directory password and the wallet password are independent, and can be different. Oracle Corporation recommends that these passwords be maintained to be consistently different, where neither one can logically be derived from the other.

See Also:

- Section 15.1.4.7, "Uploading a Wallet to an LDAP Directory"
- Section 15.1.4.8, "Downloading a Wallet from an LDAP Directory"
- Section 15.3.2, "Multiple Certificate Support"

15.1.2 Starting Oracle Wallet Manager

To start Oracle Wallet Manager:

- (Windows) Select **Start** > **Programs** > **Oracle-*Home_Name*** > **Network Administration** > **Wallet Manager**
- (UNIX) At the command line, enter `owm`.

15.1.3 How to Create a Complete Wallet: Process Overview

A wallet is a necessary repository in which to securely store user certificates and the trust points needed to validate the certificates of peers.

The following steps provide an overview of the complete wallet creation process:

1. Use Oracle Wallet Manager to create a new wallet:
   - See Section 15.1.4.1, "Required Guidelines for Creating Wallet Passwords" for information about creating a wallet password.
   - See Section 15.1.4.2, "Creating a New Wallet" for information about creating standard wallets (store credentials on your file system) and hardware security module wallets.
2. Generate a certificate request. Note that when you create a new wallet with Oracle Wallet Manager, the tool automatically prompts you to create a certificate request. See Section 15.1.5.1.1, "Adding a Certificate Request" for information about creating a certificate request.
3. Send the certificate request to the CA you want to use. You can copy and paste the certificate request text into an e-mail message, or you can export the certificate request to a file. See Section 15.1.5.1.7, "Exporting a User Certificate Request". Note that the certificate request becomes part of the wallet and must remain there until you remove its associated certificate.
4. When the CA sends your signed user certificate and its associated trusted certificate, then you can import these certificates in the following order. (Note that user certificates and trusted certificates in the PKCS #7 format can be imported at the same time.)
   - First import the CA's trusted certificate into the wallet. See Section 15.1.5.2.1, "Importing a Trusted Certificate". Note that this step may be optional if the new user certificate has been issued by one of the CAs whose trusted certificate is already present in Oracle Wallet Manager by default.
   - After you have successfully imported the trusted certificate, then import the user certificate that the CA sent to you into your wallet. See Section 15.1.5.1.2, "Importing the User Certificate into the Wallet".

Note: The BASE64 encoded PKCS#7 format used by most certificate authorities typically uses the following header and footer lines:

```
-----BEGIN PKCS7-----
-----END PKCS7-----
```

Regular certificates contain the following header & footer lines:

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

However, some certificate authorities use BEGIN CERTIFICATE and END CERTIFICATE header and footer lines in PKCS #7 format certificates as well. When certificates of PKCS #7 format are imported, the certificate authority certificates are imported as trusted certificates.

If you import the user certificate without its certificate authority certificate, Oracle Wallet Manager prompts you for the certificate authority certificate that issued the user certificate.

5. (Optional) Set the auto login feature for the wallet. See Section 15.1.4.14, "Using Auto Login".

   Typically, this feature, which enables PKI-based access to services without a password, is required for most wallets. It is required for database server and client wallets. It is only optional for products that take the wallet password at the time of startup.

After completing the preceding process, you have a wallet that contains a user certificate and its associated trust points.

15.1.4 Managing Wallets

This section describes how to create a new wallet and perform associated wallet management tasks in the following topics:

- Required Guidelines for Creating Wallet Passwords
- Creating a New Wallet
- Opening an Existing Wallet
- Closing a Wallet
- Exporting Oracle Wallets to Third-Party Environments
- Exporting Oracle Wallets to Tools That Do Not Support PKCS #12
- Uploading a Wallet to an LDAP Directory
- "Downloading a Wallet from an LDAP Directory"
- Saving Changes
- Saving the Open Wallet to a New Location
- Saving in System Default
- Deleting the Wallet
- Changing the Password
- Using Auto Login

15.1.4.1 Required Guidelines for Creating Wallet Passwords

Because an Oracle wallet contains user credentials that can be used to authenticate the user to multiple databases, it is especially important to choose a strong wallet password. A malicious user who guesses the wallet password can access all the databases to which the wallet owner has access.

Passwords must contain at least eight characters that consist of alphabetic characters combined with numbers or special characters.

Caution: It is strongly recommended that users avoid choosing easily guessed passwords based on user names, phone numbers, or government identification numbers, such as "admin0," "oracle1," or "2135551212A." This prevents a potential attacker from using personal information to deduce the users' passwords. It is also a prudent security practice for users to change their passwords periodically, such as once in each month or once in each quarter.

When you change passwords, you must regenerate auto login wallets.

See Also:

- Section 15.1.1.1, "Wallet Password Management"
- Section 15.1.4.14, "Using Auto Login"

15.1.4.2 Creating a New Wallet

You can use Oracle Wallet Manager to create PKCS #12 wallets (the standard default wallet type) that store credentials in a directory on your file system. It can also be used to create PKCS #11 wallets that store credentials on a hardware security module for servers, or private keys on tokens for clients. The following sections explain how to create both types of wallets by using Oracle Wallet Manager.

15.1.4.2.1 Creating a Standard Wallet Unless you have a hardware security module (a PKCS #11 device), then you should use a standard wallet that stores credentials in a directory on your file system.

To create a standard wallet, perform the following tasks:

1. Choose **Wallet** > **New** from the menu bar. The New Wallet dialog box appears.
2. Follow the "Required Guidelines for Creating Wallet Passwords" on page 1-9 and enter a password in the **Wallet Password** field. This password protects unauthorized use of your credentials.
3. Re-enter that password in the **Confirm Password** field.
4. Choose **Standard** from the **Wallet Type** list.
5. Click **OK** to continue. If the entered password does not conform to the required guidelines, then the following message appears:

```
Password must have a minimum length of eight characters,
and contain alphabetic characters combined with numbers
or special characters.
 Do you want to try again?
```

6. An alert is displayed, and informs you that a new empty wallet has been created. It prompts you to decide whether you want to add a certificate request. See Section 15.1.5.1.1, "Adding a Certificate Request".

If you choose **No**, you are returned to the Oracle Wallet Manager main window. The new wallet you just created appears in the left window pane. The certificate has a status of **[Empty]**, and the wallet displays its default trusted certificates.

7. Select **Wallet** > **Save In System Default** to save the new wallet.

   If you do not have permission to save the wallet in the system default, you can save it to another location. This location must be used in the SSL configuration for clients and servers.

   A message at the bottom of the window confirms that the wallet was successfully saved.

15.1.4.2.2 Creating a Wallet to Store Hardware Security Module Credentials To create a wallet to store credentials on a hardware security module that complies with PKCS #11, perform the following tasks:

1. Choose **Wallet** > **New** from the menu bar; the New Wallet dialog box appears.
2. Follow Section 15.1.4.1, "Required Guidelines for Creating Wallet Passwords" and enter a password in the **Wallet Password** field.
3. Re-enter that password in the **Confirm Password** field.
4. Choose **PKCS11** from the **Wallet Type** list, and click **OK** to continue. The New PKCS11 Wallet window appears.
5. Choose a vendor name from the **Select Hardware Vendor** list.

   Note: In the current release of Oracle Wallet Manager, only nCipher hardware has been certified to interoperate with Oracle wallets.

6. In the **PKCS11 library filename** field, enter the path to the directory in which the PKCS11 library is stored, or click **Browse** to find it by searching the file system.
7. Enter the **SmartCard password**, and choose **OK**.

   The smart card password, which is different from the wallet password, is stored in the wallet.

8. An alert is displayed, and informs you that a new empty wallet has been created. It prompts you to decide whether you want to add a certificate request. See Section 15.1.5.1.1, "Adding a Certificate Request".

   If you choose **No**, you are returned to the Oracle Wallet Manager main window. The new wallet you just created appears in the left window pane. The certificate has a status of **[Empty]**, and the wallet displays its default trusted certificates.

9. Select **Wallet** > **Save In System Default** to save the new wallet.

   If you do not have permission to save the wallet in the system default, you can save it to another location.

   A message at the bottom of the window confirms that the wallet was successfully saved.

   Note: If you change the SmartCard password or move the PKCS #11 library, an error message displays when you try to open the wallet. Then you are prompted to enter the new SmartCard password or the new path to the library.

15.1.4.3 Opening an Existing Wallet

Open a wallet that already exists in the file system directory as follows:

1. Choose **Wallet** > **Open** from the menu bar. The Select Directory dialog box appears.
2. Navigate to the directory location in which the wallet is located, and select the directory.
3. Choose **OK**. The Open Wallet dialog box appears.
4. Enter the wallet password in the **Wallet Password** field.
5. Choose **OK**.

   You are returned to the main window and a message appears at the bottom of the window indicating the wallet was opened successfully. The wallet's certificate and its trusted certificates are displayed in the left window pane.

15.1.4.4 Closing a Wallet

To close an open wallet in the currently selected directory:

Choose **Wallet** > **Close**.

A message appears at the bottom of the window to confirm that the wallet is closed.

15.1.4.5 Exporting Oracle Wallets to Third-Party Environments

Oracle Wallet Manager can export its own wallets to third party environments.

To export a wallet to third-party environments:

1. Use Oracle Wallet Manager to save the wallet file.
2. Follow the procedure specific to your third-party product to import an operating system PKCS #12 wallet file created by Oracle Wallet Manager (called `ewallet.p12` on UNIX and Windows platforms).

Note:

- Oracle Wallet Manager supports multiple certificates for each wallet, yet current browsers typically support import of single-certificate wallets only. For these browsers, you must export an Oracle wallet containing a single key-pair.
- Oracle Wallet Manager supports wallet export to only Netscape Communicator 4.7.2 and later, OpenSSL, and Microsoft Internet Explorer 5.0 and later.

15.1.4.6 Exporting Oracle Wallets to Tools That Do Not Support PKCS #12

You can export a wallet to a text-based PKI format if you want to put a wallet into a tool that does not support PKCS #12. Individual components are formatted according to the standards listed in Table 15–1. Within the wallet, only those certificates with SSL key usage are exported with the wallet.

To export a wallet to text-based PKI format:

1. Choose **Operations** > **Export Wallet...**. The Export Wallet dialog box appears.
2. Enter the destination file system directory for the wallet, or navigate to the directory structure under **Folders**.

3. Enter the destination file name for the wallet.
4. Choose **OK** to return to the main window.

Table 15–1 PKI Wallet Encoding Standards

Component	Encoding Standard
Certificate chains	X509v3
Trusted certificates	X509v3
Private keys	PKCS #8

15.1.4.7 Uploading a Wallet to an LDAP Directory

To upload a wallet to an LDAP directory, Oracle Wallet Manager uses SSL if the specified wallet contains an SSL certificate. Otherwise, it lets you enter the directory password.

To prevent accidental destruction of your wallet, Oracle Wallet Manager will not permit you to execute the upload option unless the target wallet is currently open and contains at least one user certificate.

To upload a wallet:

1. Choose **Wallet** > **Upload Into The Directory Service...**. If the currently open wallet has not been saved, a dialog box appears with the following message:

   Wallet needs to be saved before uploading.

   Choose **Yes** to proceed.

2. Wallet certificates are checked for `SSL` key usage. Depending on whether a certificate with `SSL` key usage is found in the wallet, one of the following results occur:

   - **If at least one certificate has SSL key usage:** When prompted, enter the LDAP directory server hostname and port information, then click **OK**. Oracle Wallet Manager attempts connection to the LDAP directory server using SSL. A message appears indicating whether the wallet was uploaded successfully or it failed.
   - **If no certificates have SSL key usage:** When prompted, enter the user's distinguished name (DN), the LDAP server hostname and port information, and click **OK**. Oracle Wallet Manager attempts connection to the LDAP directory server using simple password authentication mode, assuming that the wallet password is the same as the directory password.

     If the connection fails, a dialog box prompts for the directory password of the specified DN. Oracle Wallet Manager attempts connection to the LDAP directory server using this password and displays a warning message if the attempt fails. Otherwise, Oracle Wallet Manager displays a status message at the bottom of the window indicating that the upload was successful.

15.1.4.8 Downloading a Wallet from an LDAP Directory

When a wallet is downloaded from an LDAP directory, it is resident in working memory. It is not saved to the file system unless you expressly save it using any of the Save options described in the following sections.

See Also:

- Section 15.1.4.9, "Saving Changes"
- Section 15.1.4.10, "Saving the Open Wallet to a New Location"
- Section 15.1.4.11, "Saving in System Default"

To download a wallet from an LDAP directory:

1. Choose **Wallet** > **Download From The Directory Service...**.
2. A dialog box prompts for the user's distinguished name (DN), and the LDAP directory password, hostname, and port information. Oracle Wallet Manager uses simple password authentication to connect to the LDAP directory.

   Depending on whether the downloading operation succeeds or not, one of the following results occurs:

   - **If the download operation fails:** Check to make sure that you have correctly entered the user's DN, and the LDAP server hostname and port information.
   - **If the download is successful:** Choose **OK** to open the downloaded wallet. Oracle Wallet Manager attempts to open that wallet using the directory password. If the operation fails after using the directory password, then a dialog box prompts for the wallet password.

     If Oracle Wallet Manager cannot open the target wallet using the wallet password, then check to make sure you entered the correct password. Otherwise a message displays at the bottom of the window, indicating that the wallet was downloaded successfully.

15.1.4.9 Saving Changes

To save your changes to the current open wallet:

Choose **Wallet** > **Save**.

A message at the bottom of the window confirms that the wallet changes were successfully saved to the wallet in the selected directory location.

15.1.4.10 Saving the Open Wallet to a New Location

To save open wallets to a new location, use the **Save As...** menu option:

1. Choose **Wallet** > **Save As...**. The Select Directory dialog box appears.
2. Select a directory location in which to save the wallet.
3. Choose **OK**.

   The following message appears if a wallet already exists in the selected location:

   ```
   A wallet already exists in the selected path. Do you want to overwrite it?
   ```

   Choose **Yes** to overwrite the existing wallet, or **No** to save the wallet to another location.

   A message at the bottom of the window confirms that the wallet was successfully saved to the selected directory location.

15.1.4.11 Saving in System Default

To save wallets in the default directory location, use the **Save In System Default** menu option:

Choose **Wallet** > **Save In System Default**.

A message at the bottom of the window confirms that the wallet was successfully saved in the system default wallet location as follows for UNIX and Windows platforms:

- (UNIX) `/etc/ORACLE/WALLETS/$USER/`
- (Windows) `%USERPROFILE%\ORACLE\WALLETS\`

Note:

- SSL uses the wallet that is saved in the system default directory location.
- Some Oracle applications are not able to use the wallet if it is not in the system default location. Check the Oracle documentation for your specific application to determine whether wallets must be placed in the default wallet directory location.

15.1.4.12 Deleting the Wallet

To delete the current open wallet:

1. Choose **Wallet** > **Delete**. The Delete Wallet dialog box appears.
2. Review the displayed wallet location to verify you are deleting the correct wallet.
3. Enter the wallet password.
4. Choose **OK**. A dialog panel appears to inform you that the wallet was successfully deleted.

Note: Any open wallet in application memory will remain in memory until the application exits. Therefore, deleting a wallet that is currently in use does not immediately affect system operation.

15.1.4.13 Changing the Password

A password change is effective immediately. The wallet is saved to the currently selected directory, with the new encrypted password.

Note: If you are using a wallet with auto login enabled, you must regenerate the auto login wallet after changing the password. See Section 15.1.4.14, "Using Auto Login"

To change the password for the current open wallet:

1. Choose **Wallet** > **Change Password**. The Change Wallet Password dialog box appears.
2. Enter the existing wallet password.
3. Enter the new password.
4. Re-enter the new password.
5. Choose **OK**.

A message at the bottom of the window confirms that the password was successfully changed.

See Also:

- Section 15.1.4.1, "Required Guidelines for Creating Wallet Passwords"
- Section 15.1.1.1, "Wallet Password Management"

15.1.4.14 Using Auto Login

The Oracle Wallet Manager auto login feature creates an obfuscated copy of the wallet and enables PKI-based access to services without a password until the auto login feature is disabled for the wallet. File system permissions provide the necessary security for auto login wallets.

You must enable auto login if you want single sign-on access to multiple Oracle databases, which is disabled by default. Sometimes these are called "SSO wallets" because they provide single sign-on capability.

15.1.4.14.1 Enabling Auto Login To enable auto login:

1. Choose **Wallet** from the menu bar.
2. Check **Auto Login**. A message at the bottom of the window indicates that auto login is enabled.

15.1.4.14.2 Disabling Auto Login To disable auto login:

1. Choose **Wallet** from the menu bar.
2. Uncheck **Auto Login**. A message at the bottom of the window indicates that auto login is disabled.

15.1.5 Managing Certificates

Oracle Wallet Manager uses two kinds of certificates: user certificates and trusted certificates. All certificates are signed data structures that bind a network identity with a corresponding public key. User certificates are used by end entities, including server applications, to validate an end entity's identity in a public key/private key exchange. In comparison, trusted certificates are any certificates that you trust, such as those provided by CAs to validate the user certificates that they issue.

This section describes how to manage both certificate types, in the following subtopics:

- Managing User Certificates
- Managing Trusted Certificates

Note: Before a user certificate can be installed, the wallet must contain the trusted certificate representing the certificate authority who issued that user certificate. However, whenever you create a new wallet, several publicly trusted certificates are automatically installed, since they are so widely used. If the necessary certificate authority is not represented, you must install its certificate first.

Also, you can import using the PKCS#7 certificate chain format, which gives you the user certificate and the CA certificate at the same time.

15.1.5.1 Managing User Certificates

User certificates can be used by end users, smart cards, or applications, such as Web servers. Server certificates are a type of user certificate. For example, if a CA issues a certificate for a Web server, placing its distinguished name (DN) in the Subject field, then the Web server is the certificate owner, thus the "user" for this user certificate.

Managing user certificates involves the following tasks:

- Adding a Certificate Request
- Importing the User Certificate into the Wallet
- Importing Certificates Created with a Third-Party Tool
- Removing a User Certificate from a Wallet
- Removing a Certificate Request
- Exporting a User Certificate
- Exporting a User Certificate Request

15.1.5.1.1 Adding a Certificate Request You can add multiple certificate requests with Oracle Wallet Manager. When adding multiple requests, Oracle Wallet Manager automatically populates each subsequent request dialog box with the content of the initial request that you can then edit.

The actual certificate request becomes part of the wallet. You can reuse any certificate request to obtain a new certificate. However, you cannot edit an existing certificate request. Store only a correctly filled out certificate request in a wallet.

To create a PKCS #10 certificate request:

1. Choose **Operations** > **Add Certificate Request**. The Add Certificate Request dialog box appears.
2. Enter the information specified in Table 15–2.
3. Choose **OK**. A message informs you that a certificate request was successfully created. You can either copy the certificate request text from the body of this dialog panel and paste it into an e-mail message to send to a certificate authority, or you can export the certificate request to a file.
4. Choose **OK** to return to the Oracle Wallet Manager main window. The status of the certificate changes to **[Requested]**.

> **See Also:** "Section 15.1.5.1.7, "Exporting a User Certificate Request"

Table 15–2 Certificate Request: Fields and Descriptions

Field Name	Description
Common Name	Mandatory. Enter the name of the user's or service's identity. Enter a user's name in first name /last name format. Example: Eileen.Sanger
Organizational Unit	Optional. Enter the name of the identity's organizational unit. Example: Finance.
Organization	Optional. Enter the name of the identity's organization. Example: XYZ Corp.
Locality/City	Optional. Enter the name of the locality or city in which the identity resides.

Table 15–2 (Cont.) Certificate Request: Fields and Descriptions

Field Name	Description
State/Province	Optional. Enter the full name of the state or province in which the identity resides. Enter the full state name, because some certificate authorities do not accept two–letter abbreviations.
Country	Mandatory. Choose to view a list of country abbreviations. Select the country in which the organization is located.
Key Size	Mandatory. Choose to view a list of key sizes to use when creating the public/private key pair. See Table 15–3 to evaluate key size.
Advanced	Optional. Choose **Advanced** to view the Advanced Certificate Request dialog panel. Use this field to edit or customize the identity's distinguished name (DN). For example, you can edit the full state name and locality.

Table 15–3 lists the available key sizes and the relative security each size provides. Typically, CAs use key sizes of 1024 or 2048. When certificate owners wish to keep their keys for a longer duration, they choose 3072 or 4096 bit keys.

Table 15–3 Available Key Sizes

Key Size	Relative Security Level
512 or 768	Not regarded as secure.
1024 or 2048	Secure.
3072 or 4096	Very secure.

15.1.5.1.2 Importing the User Certificate into the Wallet When the Certificate Authority grants you a certificate, it may send you an e-mail that has your certificate in text (BASE64) form or attached as a binary file.

> **Note:** Certificate authorities may send your certificate in a PKCS #7 certificate chain or as an individual X.509 certificate. Oracle Wallet Manager can import both types.
>
> PKCS #7 certificate chains are a collection of certificates, including the user's certificate and all of the supporting trusted CA and subCA certificates.
>
> In contrast, an X.509 certificate file contains an individual certificate without the supporting certificate chain.
>
> However, before you can import any such individual certificate, the signer's certificate must be a Trusted Certificate in the wallet.

To import the user certificate from the text of the Certificate Authority's e-mail, copy the certificate, represented as text (BASE64), from the certificate authority's e-mail message. Include the lines `Begin Certificate` and `End Certificate`.

1. Choose **Operations** > **Import User Certificate...**. The Import Certificate dialog box appears.
2. Choose **Paste the certificate**, and then click **OK**. Another Import Certificate dialog box appears with the following message:

```
Please provide a base64 format certificate and paste it below.
```

3. Paste the certificate into the dialog box, and choose **OK**.
   a. If the certificate received is in PKCS#7 format, it is installed, and all the other certificates included with the PKCS#7 data are placed in the Trusted Certificate list.
   b. If the certificate received is *not* in PKCS#7 format, and the certificate of its CA is not already in the Trusted Certificates list, then more must be done. Oracle Wallet Manager will ask you to import the certificate of the CA that issued your certificate. This CA certificate will be placed in the Trusted Certificates list. (If the CA certificate was already in the Trusted Certificates list, your certificate is imported without additional steps.)

   After either (a) or (b) succeeds, a message at the bottom of the window confirms that the certificate was successfully installed. The Oracle Wallet Manager main window reappears, and the status of the corresponding entry in the left panel subtree changes to **[Ready]**.

Note:

The standard X.509 certificate includes the following start and end text:

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

A typical PKCS#7 certificate includes more, as described earlier, and includes the following start and end text:

```
-----BEGIN PKCS7-----
-----END PKCS7-----
```

You can use the standard Ctrl+c to copy, including all dashes, and Ctrl+v to paste.

To import the certificate from a file:

The user certificate in the file can be in either text (BASE64) or binary (`der`) format.

1. Choose **Operations** > **Import User Certificate...**. The Import Certificate dialog box appears.
2. Choose **Select a file that contains the certificate**, and click **OK**. Another Import Certificate dialog box appears.
3. Enter the path or folder name of the certificate file location.
4. Select the name of the certificate file (for example, `cert.txt`, `cert.der`).
5. Choose **OK**.
   a. If the certificate received is in PKCS#7 format, it is installed, and all the other certificates included with the PKCS#7 data are placed in the Trusted Certificate list.
   b. If the certificate received is *not* in PKCS#7 format, and the certificate of its CA is not already in the Trusted Certificates list, then more must be done. Oracle Wallet Manager will ask you to import the certificate of the CA that issued your certificate. This CA certificate will be placed in the Trusted Certificates list. (If the CA certificate was already in the Trusted Certificates list, your certificate is imported without additional steps.)

After either (a) or (b) succeeds, a message at the bottom of the window confirms that the certificate was successfully installed. You are returned to the Oracle Wallet Manager main panel, and the status of the corresponding entry in the left panel subtree changes to **[Ready]**.

15.1.5.1.3 Importing Certificates Created with a Third-Party Tool

Third-party certificates are those created from certificate requests that were not generated using Oracle Wallet Manager. These third-party certificates are actually wallets, in the Oracle sense, because they contain more than just the user certificate; they also contain the private key for that certificate. Furthermore, they include the chain of trusted certificates validating that the certificate was created by a trustworthy entity.

Oracle Wallet Manager makes these wallets available in a single step by importing them in PKCS#12 format, which includes all three elements described earlier: the user certificate, the private key, and the trusted certificates. It supports the following PKCS #12-format certificates:

- Netscape Communicator 4.x
- Microsoft Internet Explorer 5.x and later

Oracle Wallet Manager adheres to the PKCS#12 standard, so certificates exported by any PKCS#12-compliant tool should be usable with Oracle Wallet Manager.

Such third-party certificates cannot be stored into existing Oracle wallets because they would lack the private key and chain of trusted authorities. Therefore, each such certificate is exported and retrieved instead as an independent PKCS#12 file, that is, as its own wallet.

To import a certificate created with a third-party tool, you must first export it from the application you are using, and then save it as a wallet file that can be read by Oracle Wallet Manager. See Section 15.1.5.1.3, "Importing Certificates Created with a Third-Party Tool" for information about importing certificates that are created with third-party tools.

To import a certificate created with a third-party tool, perform the following steps:

1. Follow the procedures for your particular product to export the certificate. Take the actions indicated in the exporting product to include the private key in the export, and specify the new password to protect the exported certificate. Also include all associated trust points. (Under PKCS #12, browsers do not necessarily export **trusted certificates**, other than the signer's own certificate. You may need to add additional certificates to authenticate to your peers. You can use Oracle Wallet Manager to import trusted certificates.)

   The resulting file, containing the certificate, the private key, and the trust points, is the new wallet that enables the third-party certificate to be used.

2. Save the exported certificate to a file name appropriate for your operating system in a directory expected by Oracle Wallet Manager.

   For UNIX and Windows, the appropriate file name is `ewallet.p12`.

   For other operating systems, see the Oracle documentation for the applicable operating system.

3. Use Oracle Wallet Manager to navigate to the directory in which you saved the `ewallet.p12` file and open it to use the PKI credentials it contains.

Note: The password will be required whenever the associated application starts up or otherwise needs the certificate. To make such access automatic, see Section 15.1.4.14, "Using Auto Login".

However, if the private key for the desired certificate is held in a separate hardware security module, you will not be able to import that certificate.

If you exported the trusted certificate separately, then you must import the trusted certificate first before you open the `ewallet.p12` file that contains the imported third-party user certificate.

See Also: "Section 15.1.5.2.1, "Importing a Trusted Certificate"

15.1.5.1.4 Removing a User Certificate from a Wallet To remove a user certificate from a wallet:

1. In the left panel subtree, select the certificate that you want to remove.
2. Choose **Operations** > **Remove User Certificate...**. A dialog panel appears and prompts you to verify that you want to remove the user certificate from the wallet.
3. Choose **Yes** to return to the Oracle Wallet Manager main panel. The certificate displays a status of **[Requested]**.

15.1.5.1.5 Removing a Certificate Request You must remove a certificate before removing its associated request.

To remove a certificate request:

1. In the left panel subtree, select the certificate request that you want to remove.
2. Choose **Operations** > **Remove Certificate Request...**.
3. Click **Yes**. The certificate displays a status of **[Empty]**.

15.1.5.1.6 Exporting a User Certificate To save the certificate in a file system directory, export the certificate by using the following steps:

1. In the left panel subtree, select the certificate that you want to export.
2. Choose **Operations** > **Export User Certificate...** from the menu bar. The Export Certificate dialog box appears.
3. Enter the file system directory location in which you want to save your certificate, or navigate to the directory structure under **Folders**.
4. Enter a file name for your certificate in the **Enter File Name** field.
5. Choose **OK**. A message at the bottom of the window confirms that the certificate was successfully exported to the file. You are returned to the Oracle Wallet Manager main window.

See Also: "Section 15.1.4.5, "Exporting Oracle Wallets to Third-Party Environments" for information about exporting wallets. Note that Oracle Wallet Manager supports storing multiple certificates in a single wallet, yet current browsers typically support only single-certificate wallets. For these browsers, you must export an Oracle wallet that contains a single key-pair.

15.1.5.1.7 Exporting a User Certificate Request To save the certificate request in a file system directory, export the certificate request by using the following steps:

1. In the left panel subtree, select the certificate request that you want to export.
2. Choose **Operations** > **Export Certificate Request...**. The Export Certificate Request dialog box appears.
3. Enter the file system directory location in which you want to save your certificate request, or navigate to the directory structure under **Folders**.
4. Enter a file name for your certificate request, in the **Enter File Name** field.
5. Choose **OK**. A message at the bottom of the window confirms that the certificate request was successfully exported to the file. You are returned to the Oracle Wallet Manager main window.

15.1.5.2 Managing Trusted Certificates

Managing trusted certificates includes the following tasks:

- Importing a Trusted Certificate
- Removing a Trusted Certificate
- Exporting a Trusted Certificate
- Exporting All Trusted Certificates

15.1.5.2.1 Importing a Trusted Certificate You can import a trusted certificate into a wallet in either of two ways: paste the trusted certificate from an e-mail that you receive from the certificate authority, or import the trusted certificate from a file.

Oracle Wallet Manager automatically installs trusted certificates from VeriSign, RSA, Entrust, and GTE CyberTrust when you create a new wallet.

To copy and paste the text only (BASE64) trusted certificate:

Copy the trusted certificate from the body of the e-mail message you received that contained the user certificate. Include the lines `Begin Certificate` and `End Certificate.`

1. Choose **Operations** > **Import Trusted Certificate...** from the menu bar. The Import Trusted Certificate dialog panel appears.
2. Choose **Paste the Certificate**, and click **OK**. Another Import Trusted Certificate dialog panel appears with the following message:

   ```
   Please provide a base64 format certificate and paste it below.
   ```

3. Paste the certificate into the window, and click **OK**. A message at the bottom of the window informs you that the trusted certificate was successfully installed.
4. Choose **OK**. You are returned to the Oracle Wallet Manager main panel, and the trusted certificate appears at the bottom of the Trusted Certificates tree.

Keyboard shortcuts for copying and pasting certificates:

Use Ctrl+c to copy, and use Ctrl+v to paste.

To import a file that contains the trusted certificate:

The file containing the trusted certificate should have been saved in either text (BASE64) or binary (`der`) format.

1. Choose **Operations** > **Import Trusted Certificate...**. The Import Trusted Certificate dialog panel appears.
2. Enter the path or folder name of the trusted certificate location.
3. Select the name of the trusted certificate file (for example, `cert.txt`).
4. Choose **OK**. A message at the bottom of the window informs you that the trusted certificate was successfully imported into the wallet.
5. Choose **OK** to exit the dialog panel. You are returned to the Oracle Wallet Manager main panel, and the trusted certificate appears at the bottom of the Trusted Certificates tree.

15.1.5.2.2 Removing a Trusted Certificate You cannot remove a trusted certificate if it has been used to sign a user certificate still present in the wallet. To remove such trusted certificates, you must first remove the certificates it has signed. Also, you cannot verify a certificate after its trusted certificate has been removed from your wallet.

To remove a trusted certificate from a wallet:

1. Select the trusted certificate listed in the Trusted Certificates tree.
2. Choose **Operations** > **Remove Trusted Certificate...** from the menu bar.

   A dialog panel warns you that your user certificate will no longer be verifiable by its recipients if you remove the trusted certificate that was used to sign it.
3. Choose **Yes**. The selected trusted certificate is removed from the Trusted Certificates tree.

15.1.5.2.3 Exporting a Trusted Certificate To export a trusted certificate to another file system location:

1. In the left panel subtree, select the trusted certificate that you want to export.
2. Select **Operations** > **Export Trusted Certificate...**. The Export Trusted Certificate dialog box appears.
3. Enter a file system directory in which you want to save your trusted certificate, or navigate to the directory structure under **Folders**.
4. Enter a file name to save your trusted certificate.
5. Choose **OK**. You are returned to the Oracle Wallet Manager main window.

15.1.5.2.4 Exporting All Trusted Certificates To export all of your trusted certificates to another file system location:

1. Choose **Operations** > **Export All Trusted Certificates...**. The Export Trusted Certificate dialog box appears.
2. Enter a file system directory location in which you want to save your trusted certificates, or navigate to the directory structure under **Folders**.
3. Enter a file name to save your trusted certificates.
4. Choose **OK**. You are returned to the Oracle Wallet Manager main window.

15.2 Performing Certificate Validation and CRL Management with the orapki Utility

The `orapki` utility is a command-line tool that you can use to manage certificate revocation lists (CRLs), create and manage Oracle wallets, and to create signed certificates for testing purposes.

The following topics describe this tool and how to use it:

15.2.1 orapki Overview

The `orapki` utility is provided to manage public key infrastructure (PKI) elements, such as wallets and certificate revocation lists, on the command line so the tasks it performs can be incorporated into scripts. This enables you to automate many of the routine tasks of maintaining a PKI.

This command-line utility can be used to perform the following tasks:

- Creating signed certificates for testing purposes
- Manage Oracle wallets:
  - Create and display Oracle wallets
  - Add and remove certificate requests
  - Add and remove certificates
  - Add and remove trusted certificates
- Manage certificate revocation lists (CRLs):
  - Renaming CRLs with a hash value for certificate validation
  - Uploading, listing, viewing, and deleting CRLs in Oracle Internet Directory

15.2.1.1 orapki Utility Syntax

The basic syntax of the `orapki` command-line utility is as follows:

```
orapki module command -parameter value
```

In the preceding command, *`module`* can be `wallet` (Oracle wallet), `crl` (certificate revocation list), or `cert` (PKI digital certificate). The available commands depend on the `module` you are using. For example, if you are working with a `wallet`, then you can add a certificate or a key to the wallet with the `add` command. The following example adds the user certificate located at `/private/lhale/cert.txt` to the wallet located at `ORACLE_HOME/wallet/ewallet.p12`:

```
orapki wallet add -wallet ORACLE_HOME/wallet/ewallet.p12
-user_cert -cert /private/lhale/cert.txt
```

15.2.2 Displaying orapki Help

You can display all the `orapki` commands that are available for a specific mode by entering the following at the command line:

```
orapki mode help
```

For example, to display all available commands for managing certificate revocation lists (CRLs), enter the following at the command line:

```
orapki CRL help
```

> **Note:** Using the `-summary`, `-complete`, or `-wallet` command options is always optional. A command will still run if these command options are not specified.

15.2.3 Creating Signed Certificates for Testing Purposes

This command-line utility provides a convenient, lightweight way to create signed certificates for testing purposes. The following syntax can be used to create signed certificates and to view certificates:

To create a signed certificate for testing purposes:

```
orapki cert create [-wallet wallet_location] -request
 certificate_request_location
-cert certificate_location -validity number_of_days [-summary]
```

This command creates a signed certificate from the certificate request. The `-wallet` parameter specifies the wallet containing the user certificate and private key that will be used to sign the certificate request. The `-validity` parameter specifies the number of days, starting from the current date, that this certificate will be valid. Specifying a certificate and certificate request is mandatory for this command.

To view a certificate:

```
orapki cert display -cert certificate_location [-summary | -complete]
```

This command enables you to view a test certificate that you have created with `orapki`. You can choose either `-summary` or `-complete`, which determines how much detail the command will display. If you choose `-summary`, the command will display the certificate and its expiration date. If you choose `-complete`, it will display additional certificate information, including the serial number and public key.

15.2.4 Managing Oracle Wallets with the orapki Utility

The following sections describe the syntax used to create and manage Oracle wallets with the `orapki` command-line utility. You can use these `orapki` utility `wallet` module commands in scripts to automate the wallet creation process.

- Creating and Viewing Oracle Wallets with `orapki`
- Adding Certificates and Certificate Requests to Oracle Wallets with `orapki`
- Exporting Certificates and Certificate Requests from Oracle Wallets with `orapki`

> **Note:** The `-wallet` parameter is mandatory for all `wallet` module commands.

15.2.4.1 Creating and Viewing Oracle Wallets with orapki

To create an Oracle wallet:

```
orapki wallet create -wallet wallet_location
```

This command will prompt you to enter and re-enter a wallet password. It creates a wallet in the location specified for `-wallet`.

To create an Oracle wallet with auto login enabled:

```
orapki wallet create -wallet wallet_location -auto_login
```

This command creates a wallet with auto login enabled, or it can also be used to enable auto login on an existing wallet. If the `wallet_location` already contains a wallet, then auto login will be enabled for it. To turn the auto login feature off, use Oracle Wallet Manager. See Section 15.1.4.14, "Using Auto Login" for details.

> **Note:** For wallets with the auto login feature enabled, you are prompted for a password only for operations that modify the wallet, such as `add`.

To view an Oracle wallet:

```
orapki wallet display -wallet wallet_location
```

Displays the certificate requests, user certificates, and trusted certificates contained in the wallet.

15.2.4.2 Adding Certificates and Certificate Requests to Oracle Wallets with orapki

To add a certificate request to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -dn user_dn -keySize 512|1024|2048
```

This command adds a certificate request to a wallet for the user with the specified distinguished name (`user_dn`). The request also specifies the requested certificate's key size (512, 1024, or 2048 bits). To sign the request, export it with the export option. See Section 15.2.4.3, "Exporting Certificates and Certificate Requests from Oracle Wallets with orapki".

To add a trusted certificate to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -trusted_cert -cert
certificate_location
```

This command adds a trusted certificate, at the specified location (`-cert certificate_location`), to a wallet. You must add all trusted certificates in the certificate chain of a user certificate before adding a user certificate, or the command to add the user certificate will fail.

To add a root certificate to an Oracle wallet

```
orapki wallet add -wallet wallet_location -dn
certificate_dn -keySize 512|1024|2048 -self_signed -validity number_of_days
```

This command creates a new self-signed (root) certificate and adds it to the wallet. The `-validity` parameter (mandatory) specifies the number of days, starting from the

current date, that this certificate will be valid. You can specify a key size for this root certificate (`-keySize`) of 512, 1024, or 2048 bits.

To add a user certificate to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -user_cert -cert certificate_location
```

This command adds the user certificate at the location specified with the `-cert` parameter to the Oracle wallet at the *`wallet_location`*. Before you add a user certificate to a wallet, you must add all the trusted certificates that make up the certificate chain. If all trusted certificates are not installed in the wallet before you add the user certificate, then adding the user certificate will fail.

15.2.4.3 Exporting Certificates and Certificate Requests from Oracle Wallets with orapki

To export a certificate from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_dn -cert certificate_filename
```

This command exports a certificate with the subject's distinguished name (`-dn`) from a wallet to a file that is specified by `-cert`.

To export a certificate request from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_request_dn -request certificate_request_filename
```

This command exports a certificate request with the subject's distinguished name (`-dn`) from a wallet to a file that is specified by `-request`.

15.2.5 Managing Certificate Revocation Lists (CRLs) with the orapki Utility

CRLs must be managed with `orapki`. This utility creates a hashed value of the CRL issuer's name to identify the CRLs location in your system. If you do not use `orapki`, your Oracle server cannot locate CRLs to validate PKI digital certificates. The following sections describe CRLS, how you use them, and how to use orapki to manage them:

- Section 15.2.5.1, "About Certificate Validation with Certificate Revocation Lists"
- Section 15.2.5.2, "Certificate Revocation List Management"

15.2.5.1 About Certificate Validation with Certificate Revocation Lists

The process of determining whether a given certificate can be used in a given context is referred to as certificate validation. Certificate validation includes determining that

- A trusted certificate authority (CA) has digitally signed the certificate
- The certificate's digital signature corresponds to the independently-calculated hash value of the certificate itself and the certificate signer's (CA's) public key
- The certificate has not expired
- The certificate has not been revoked

The SSL network layer automatically performs the first three validation checks, but you must configure certificate revocation list (CRL) checking to ensure that certificates have not been revoked. CRLs are signed data structures that contain a list of revoked

certificates. They are usually issued and signed by the same entity who issued the original certificate.

15.2.5.1.1 What CRLs Should You Use? You should have CRLs for all of the trust points that you honor. The trust points are the trusted certificates from a third party identity that is qualified with a level of trust. Typically, the certificate authorities you trust are called trust points.

15.2.5.1.2 How CRL Checking Works Certificate revocation status is checked against CRLs which are located in file system directories, Oracle Internet Directory, or downloaded from the location specified in the CRL Distribution Point (CRL DP) extension on the certificate. If you store your CRLs on the local file system or in the directory, then you must update them regularly. If you use CRL DPs then CRLs are downloaded each time a certificate is used so there is no need to regularly refresh the CRLs.

The server searches for CRLs in the following locations in the order listed. When the system finds a CRL that matches the certificate CA's DN, it stops searching.

1. Local file system

   The system checks the `sqlnet.ora` file for the `SSL_CRL_FILE` parameter first, followed by the `SSL_CRL_PATH` parameter. If these two parameters are not specified, then the system checks the wallet location for any CRLs.

   Note: if you store CRLs on your local file system, then you must use the `orapki` utility to periodically update them. See "Renaming CRLs with a Hash Value for Certificate Validation" on page 1-28

2. Oracle Internet Directory

   If the server cannot locate the CRL on the local file system and directory connection information has been configured in the `ORACLE_HOME/ldap/admin/ldap.ora` file, then the server searches in the directory. It searches the CRL subtree by using the CA's distinguished name (DN) and the DN of the CRL subtree.

   The server must have a properly configured `ldap.ora` file to search for CRLs in the directory. It cannot use the Domain Name System (DNS) discovery feature of Oracle Internet Directory. Also note that if you store CRLs in the directory, then you must use the `orapki` utility to periodically update them. See "Uploading CRLs to Oracle Internet Directory" on page 1-28

3. CRL DP

   If the CA specifies a location in the CRL DP X.509, version 3, certificate extension when the certificate is issued, then the appropriate CRL that contains revocation information for that certificate is downloaded. Currently, Oracle Advanced Security supports downloading CRLs over HTTP and LDAP.

Notes:

- For performance reasons, only user certificates are checked.
- Oracle recommends that you store CRLs in the directory rather than the local file system.

15.2.5.2 Certificate Revocation List Management

Before you can enable certificate revocation status checking, you must ensure that the CRLs you receive from the CAs you use are in a form (renamed with a hash value) or in a location (uploaded to the directory) in which your system can use them. Oracle Advanced Security provides a command-line utility, `orapki`, that you can use to perform the following tasks:

- Renaming CRLs with a Hash Value for Certificate Validation
- Uploading CRLs to Oracle Internet Directory
- Listing CRLs Stored in Oracle Internet Directory
- Viewing CRLs in Oracle Internet Directory
- Deleting CRLs from Oracle Internet Directory

> **Note:** CRLs must be updated at regular intervals (before they expire) for successful validation. You can automate this task by using `orapki` commands in a script.

You can also use LDAP command-line tools to manage CRLs in Oracle Internet Directory.

> **See Also:** Appendix A, "Syntax for Command-Line Tools" in *Oracle Identity Management Application Developer's Guide* for information about LDAP command-line tools and their syntax.

15.2.5.2.1 Renaming CRLs with a Hash Value for Certificate Validation When the system validates a certificate, it must locate the CRL issued by the CA who created the certificate. The system locates the appropriate CRL by matching the issuer name in the certificate with the issuer name in the CRL.

When you specify a CRL storage location for the **Certificate Revocation Lists Path** field in Oracle Net Manager (sets the `SSL_CRL_PATH` parameter in the `sqlnet.ora` file), use the `orapki` utility to rename CRLs with a hash value that represents the issuer's name. Creating the hash value enables the server to load the CRLs.

On UNIX operating systems, `orapki` creates a symbolic link to the CRL. On Windows operating systems, it creates a copy of the CRL file. In either case, the symbolic link or the copy created by `orapki` are named with a hash value of the issuer's name. Then when the system validates a certificate, the same hash function is used to calculate the link (or copy) name so the appropriate CRL can be loaded.

Depending on your operating system, enter one of the following commands to rename CRLs stored in the file system.

To rename CRLs stored in UNIX file systems:

```
orapki crl hash -crl crl_filename [-wallet wallet_location]
-symlink crl_directory [-summary]
```

To rename CRLs stored in Windows file systems:

```
orapki crl hash -crl crl_filename
[-wallet wallet_location] -copy crl_directory [-summary]
```

In the preceding commands, `crl_filename` is the name of the CRL file, *`wallet_location`* is the location of a wallet that contains the certificate of the CA that issued the CRL, and *`crl_directory`* is the directory in which the CRL is located.

Using `-wallet` and `-summary` are optional. Specifying `-wallet` causes the tool to verify the validity of the CRL against the CA's certificate prior to renaming the CRL. Specifying the `-summary` option causes the tool to display the CRL issuer's name.

15.2.5.2.2 Uploading CRLs to Oracle Internet Directory Publishing CRLs in the directory enables CRL validation throughout your enterprise, eliminating the need for individual applications to configure their own CRLs. All applications can use the CRLs stored in the directory in which they can be centrally managed, greatly reducing the administrative overhead of CRL management and use.

The user who uploads CRLs to the directory by using `orapki` must be a member of the directory group `CRLAdmins` (`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`). This is a privileged operation because these CRLs are accessible to the entire enterprise. Contact your directory administrator to be added to this administrative directory group.

To upload CRLs to the directory, enter the following at the command line:

```
orapki crl upload -crl crl_location
-ldap hostname:ssl_port -user username [-wallet wallet_location] [-summary]
```

In the preceding command, *`crl_location`* is the file name or URL in which the CRL is located, *`hostname`* and *`ssl_port`* (SSL port with no authentication) are for the system on which your directory is installed, *`username`* is the directory user who has permission to add CRLs to the CRL subtree, and *`wallet_location`* is the location of a wallet that contains the certificate of the CA that issued the CRL.

Using `-wallet` and `-summary` are optional. Specifying `-wallet` causes the tool to verify the validity of the CRL against the CA's certificate prior to uploading it to the directory. Specifying the `-summary` option causes the tool to print the CRL issuer's name and the LDAP entry in which the CRL is stored in the directory.

Note:

- The `orapki` utility will prompt you for the directory password when you perform this operation.
- Ensure that you specify the directory SSL port on which the Diffie-Hellman-based SSL server is running. This is the SSL port that does not perform authentication. Neither the server authentication nor the mutual authentication SSL ports are supported by the `orapki` utility.

15.2.5.2.3 Listing CRLs Stored in Oracle Internet Directory You can display a list of all CRLs stored in the directory with `orapki`, which is useful for browsing to locate a particular CRL to view or download to your local system. This command displays the CA who issued the CRL (Issuer) and its location (DN) in the CRL subtree of your directory.

To list CRLs in Oracle Internet Directory, enter the following at the command line:

```
orapki crl list -ldap hostname:ssl_port
```

In the preceding command, the *hostname* and *ssl_port* are for the system on which your directory is installed. Note that this is the directory SSL port with no authentication as described in the preceding section.

15.2.5.2.4 Viewing CRLs in Oracle Internet Directory You can view specific CRLs that are stored in Oracle Internet Directory in a summarized format or you can request a complete listing of revoked certificates for the specified CRL. A summary listing provides the CRL issuer's name and its validity period. A complete listing provides a list of all revoked certificates contained in the CRL.

To view a summary listing of a CRL in Oracle Internet Directory, enter the following at the command line:

```
orapki crl display -crl crl_location [-wallet wallet_location] -summary
```

In the preceding command, *crl_location* is the location of the CRL in the directory. It is convenient to paste the CRL location from the list that displays when you use the `orapki crl list` command. See: "Listing CRLs Stored in Oracle Internet Directory" on page 1-29.

To view a list of all revoked certificates contained in a specified CRL, which is stored in Oracle Internet Directory, enter the following at the command line:

```
orapki crl display -crl crl_location [-wallet wallet_location] -complete
```

For example, the following `orapki` command:

```
orapki crl display -crl $T_WORK/pki/wlt_crl/nzcrl.txt -wallet $T_WORK/pki/wlt_crl
-complete
```

produces the following output, which lists the CRL issuer's DN, its publication date, date of its next update, and the revoked certificates it contains:

```
issuer = CN=root,C=us, thisUpdate = Sun Nov 16 10:56:58 PST 2003,
nextUpdate = Mon Sep 30 11:56:58 PDT 2013, revokedCertificates =
{(serialNo = 153328337133459399575438325845117876415,
revocationDate - Sun Nov 16 10:56:58 PST 2003)}
CRL is valid
```

Using the `-wallet` option causes the `orapki crl display` command to validate the CRL against the CA's certificate.

Depending on the size of your CRL, choosing the `-complete` option may take a long time to display.

You can also use Oracle Directory Manager, a graphical user interface tool that is provided with Oracle Internet Directory, to view CRLs in the directory. CRLs are stored in the following directory location:

```
cn=CRLValidation,cn=Validation,cn=PKI,cn=Products,cn=OracleContext
```

15.2.5.2.5 Deleting CRLs from Oracle Internet Directory The user who deletes CRLs from the directory by using `orapki` must be a member of the directory group `CRLAdmins`. See Section 15.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for information about this directory administrative group.

To delete CRLs from the directory, enter the following at the command line:

```
orapki crl delete -issuer issuer_name -ldap hostname:ssl_port
-user username [-summary]
```

In the preceding command, *issuer_name* is the name of the CA who issued the CRL, the *hostname* and *ssl_port* are for the system on which your directory is installed, and *username* is the directory user who has permission to delete CRLs from the CRL subtree. Note that this must be a directory SSL port with no authentication. See Section 15.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.

Using the `-summary` option causes the tool to print the CRL LDAP entry that was deleted.

For example, the following `orapki` command:

```
orapki crl delete -issuer "CN=root,C=us"
-ldap machine1:3500 -user cn=orcladmin -summary
```

produces the following output, which lists the location of the deleted CRL in the directory:

```
Deleted CRL at cn=root
cd45860c.rN,cn=CRLValidation,cn=Validation,cn=PKI,cn=Products,cn=OracleContext
```

15.2.6 orapki Utility Commands Summary

This section lists and describes the following `orapki` commands:

15.2.6.1 orapki cert create

The following sections describe this command.

15.2.6.1.1 Purpose Use this command to create a signed certificate for testing purposes.

15.2.6.1.2 Syntax `orapki cert create [-wallet` *wallet_location*`]`
`-request` *certificate_request_location*
`-cert` *certificate_location* `-validity` *number_of_days* `[-summary]`

- The `-wallet` parameter specifies the wallet containing the user certificate and private key that will be used to sign the certificate request.
- The `-request` parameter (mandatory) specifies the location of the certificate request for the certificate you are creating.

- The `-cert` parameter (mandatory) specifies the directory location in which the tool places the new signed certificate.
- The `-validity` parameter (mandatory) specifies the number of days, starting from the current date, that this certificate will be valid.

15.2.6.2 orapki cert display

The following sections describe this command.

15.2.6.2.1 Purpose Use this command to display details of a specific certificate.

15.2.6.2.2 Syntax `orapki cert display -cert` *`certificate_location`*
`[-summary|-complete]`

- The `-cert` parameter specifies the location of the certificate you want to display.
- You can use either the `-summary` or the `-complete` parameter to display the following information:
  - `-summary` displays the certificate and its expiration date
  - `-complete` displays additional certificate information, including the serial number and public key

15.2.6.3 orapki crl delete

The following sections describe this command.

15.2.6.3.1 Purpose Use this command to delete CRLs from Oracle Internet Directory. Note that the user who deletes CRLs from the directory by using `orapki` must be a member of the `CRLAdmins` (`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`) directory group.

15.2.6.3.2 Prerequisites None

15.2.6.3.3 Syntax `orapki crl delete -issuer` *`issuer_name`*
`-ldap` *`hostname`*`:`*`ssl_port`* `-user` *`username`* `[-summary]`

- The `-issuer` parameter specifies the name of the certificate authority (CA) who issued the CRL.
- The `-ldap` parameter specifies the hostname and SSL port for the directory in which the CRLs are to be deleted. Note that this must be a directory SSL port with no authentication. See "Uploading CRLs to Oracle Internet Directory" on page 7-29 for more information about this port.
- The `-user` parameter specifies the username of the directory user who has permission to delete CRLs from the CRL subtree in the directory.
- The `-summary` parameter is optional. Using it causes the tool to print the CRL LDAP entry that was deleted.

15.2.6.4 orapki crl display

The following sections describe this command.

15.2.6.4.1 Purpose Use this command to display specific CRLs that are stored in Oracle Internet Directory.

15.2.6.4.2 Syntax `orapki crl display -crl` *`crl_location`*
`[-wallet` *`wallet_location`*`] [-summary|-complete]`

- The `-crl` parameter specifies the location of the CRL in the directory. It is convenient to paste the CRL location from the list that displays when you use the `orapki crl list` command. See "orapki crl list" on page 1-33
- The `-wallet` parameter (optional) specifies the location of the wallet that contains the certificate of the certificate authority (CA) who issued the CRL. Using it causes the tool to verify the validity of the CRL against the CA's certificate prior to displaying it.
- Choosing either the -summary or the -complete parameters displays the following information:
  - `-summary` provides a listing that contains the CRL issuer's name and the CRL's validity period
  - `-complete` provides a list of all revoked certificates that the CRL contains. Note that this option may take a long time to display, depending on the size of the CRL.

15.2.6.5 orapki crl hash

The following sections describe this command.

15.2.6.5.1 Purpose Use this command to generate a hash value of the certificate revocation list (CRL) issuer to identify the location of the CRL in your file system for certificate validation.

15.2.6.5.2 Syntax `orapki crl hash -crl` *`crl_filename|URL`*
`[-wallet` *`wallet_location`*`] [-symlink|-copy]` *`crl_directory`* `[-summary]`

- The `-crl` parameter specifies the filename that contains the CRL or the URL in which it can be found.
- The `-wallet` parameter (optional) specifies the location of the wallet that contains the certificate of the certificate authority (CA) who issued the CRL. Using it causes the tool to verify the validity of the CRL against the CA's certificate prior to uploading it to the directory.
- Depending on your operating system, use either the `-symlink` or the `-copy` parameter:
  - (UNIX) use `-symlink` to create a symbolic link to the CRL at the *`crl_directory`* location
  - (Windows) use `-copy` to create a copy of the CRL at the *`crl_directory`* location
- The `-summary` parameter (optional) causes the tool to display the CRL issuer's name.

15.2.6.6 orapki crl list

The following sections describe this command.

15.2.6.6.1 Purpose Use this command to display a list of CRLs stored in Oracle Internet Directory. This is useful for browsing to locate a particular CRL to view or download to your local file system.

15.2.6.6.2 Syntax `orapki crl list -ldap hostname:ssl_port`

The `-ldap` parameter specifies the hostname and SSL port for the directory server from which you want to list CRLs. Note that this must be a directory SSL port with no authentication. See Section 15.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.

15.2.6.7 orapki crl upload

The following sections describe this command.

15.2.6.7.1 Purpose Use this command to upload certificate revocation lists (CRLs) to the CRL subtree in Oracle Internet Directory. Note that you must be a member of the directory administrative group `CRLAdmins` (`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`) to upload CRLs to the directory.

15.2.6.7.2 Syntax
```
orapki crl upload -crl crl_location
-ldap hostname:ssl_port -user username
[-wallet wallet_location] [-summary]
```

- The `-crl` parameter specifies the directory location or the URL of the CRL that you are uploading to the directory.
- The `-ldap` parameter specifies the hostname and SSL port for the directory to which you are uploading the CRLs. Note that this must be a directory SSL port with no authentication. See Section 15.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.
- The `-user` parameter specifies the username of the directory user who has permission to add CRLs to the CRL subtree in the directory.
- The `-wallet` parameter specifies the location of the wallet that contains the certificate of the certificate authority (CA) who issued the CRL. This is an optional parameter. Using it causes the tool to verify the validity of the CRL against the CA's certificate prior to uploading it to the directory.
- The `-summary` parameter is also optional. Using it causes the tool to display the CRL issuer's name and the LDAP entry in which the CRL is stored in the directory.

15.2.6.8 orapki wallet add

The following sections describe this command.

15.2.6.8.1 Purpose Use this command to add certificate requests and certificates to an Oracle wallet.

15.2.6.8.2 Syntax To add certificate requests:

```
orapki wallet add -wallet wallet_location -dn user_dn -keySize 512|1024|2048
```

- The `-wallet` parameter specifies the location of the wallet to which you want to add a certificate request.
- The `-dn` parameter specifies the distinguished name of the certificate owner.
- The `-keySize` parameter specifies the key size for the certificate.
- To sign the request, export it with the export option. See Section 15.2.6.11, "orapki wallet export".

To add trusted certificates:

```
orapki wallet add -wallet wallet_location -trusted_cert -cert certificate_location
```

- The `-trusted_cert` parameter causes the tool to add the trusted certificate, at the location specified with `-cert`, to the wallet.

To add root certificates:

```
orapki wallet add -wallet wallet_location -dn
certificate_dn -keySize 512|1024|2048 -self_signed -validity number_of_days
```

- The `-self_signed` parameter causes the tool to create a root certificate.
- The `-validity` parameter is mandatory. Use it to specify the number of days, starting from the current date, that this root certificate will be valid.

To add user certificates:

```
orapki wallet add -wallet wallet_location -user_cert -cert certificate_location
```

- The `-user_cert` parameter causes the tool to add the user certificate at the location specified with the `-cert` parameter to the wallet. Before you add a user certificate to a wallet, you must add all the trusted certificates that make up the certificate chain. If all trusted certificates are not installed in the wallet before you add the user certificate, then adding the user certificate will fail.

15.2.6.9 orapki wallet create

The following sections describe this command.

15.2.6.9.1 Purpose Use this command to create an Oracle wallet or to set auto login on for an Oracle wallet.

15.2.6.9.2 Syntax `orapki wallet create -wallet wallet_location [-auto_login]`

- The `-wallet` parameter specifies a location for the new wallet or the location of the wallet for which you want to turn on auto login.
- The `-auto_login` parameter creates an auto login wallet, or it turns on automatic login for the wallet specified with the `-wallet` option. See Section 15.1.4.14, "Using Auto Login" for details about auto login wallets.

15.2.6.10 orapki wallet display

The following sections describe this command.

15.2.6.10.1 Purpose Use this command to view the certificate requests, user certificates, and trusted certificates in an Oracle wallet.

15.2.6.10.2 Syntax `orapki wallet display -wallet wallet_location`

- The `-wallet` parameter specifies a location for the wallet you want to open if it is not located in the current working directory.

15.2.6.11 orapki wallet export

The following sections describe this command.

15.2.6.11.1 Purpose Use this command to export certificate requests and certificates from an Oracle wallet.

15.2.6.11.2 Syntax To export a certificate from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_dn -cert certificate_filename
```

- The `-wallet` parameter specifies the location of the wallet from which you want to export the certificate.
- The `-dn` parameter specifies the distinguished name of the certificate.
- The `-cert` parameter specifies the name of the file that contains the exported certificate.

To export a certificate request from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_request_dn -request certificate_request_filename
```

- The `-request` parameter specifies the name of the file that contains the exported certificate request.

15.3 Interoperability with X.509 Certificates

Oracle Wallet Manager functionality supports users who already have certificates provisioned. If you do not use Oracle Wallet Manager to create certificates, you can use it to manage and store certificates created previously.

15.3.1 Public-Key Cryptography Standards (PKCS) Support

Oracle Wallet Manager stores X.509 certificates and private keys in Public-Key Cryptography Standards (PKCS) #12 format, and generates certificate requests according to the PKCS #10 specification developed by RSA Laboratories. This makes the Oracle wallet structure interoperable with supported third party PKI applications, and provides wallet portability across operating systems.

Oracle Wallet Manager wallets can be enabled to store credentials on hardware security modules using APIs that conform to the PKCS #11 specification. When `PKCS11` wallet type is chosen at the time of wallet creation, then all keys stored in that wallet are saved to a hardware security module or token, such as smart cards, PCMCIA cards, smart diskettes, or other types of portable hardware devices that store private keys, perform cryptographic operations, or both.

See Also:

- Section 15.1.5.1.3, "Importing Certificates Created with a Third-Party Tool"
- Section 15.1.4.5, "Exporting Oracle Wallets to Third-Party Environments"
- Section 15.1.4.2.2, "Creating a Wallet to Store Hardware Security Module Credentials"
- To view PKCS standards documents, navigate to the following URL:

  `http://www.rsasecurity.com/rsalabs/`

15.3.2 Multiple Certificate Support

Oracle Wallet Manager enables you to store multiple certificates for each wallet, supporting the following Oracle PKI certificate usages:

- SSL
- S/MIME signature
- S/MIME encryption
- Code-Signing
- CA Certificate Signing

Oracle Wallet Manager supports multiple certificates for a single digital entity, where each certificate can be used for a set of Oracle PKI certificate usages, but the same certificate cannot be used for all such usages (See Table 15–4 and Table 15–5 for legal usage combinations). There must be a one-to-one mapping between certificate requests and certificates. The same certificate request can be used to obtain multiple certificates; however, more than one certificate for each certificate request cannot be installed in the same wallet at the same time.

Oracle Wallet Manager uses the X.509 Version 3 `KeyUsage` extension types to define Oracle PKI certificate usages. The key usage extension types are optional bits that can be set in certificates. Setting these bits defines what purpose the certificate's key can be used for. When certificates are issued, the certificate authority sets these bits according to the type of certificate that you have requested. Table 15–4 lists and describes these key usage types.

Table 15–4 X.509 Version 3 KeyUsage Extension Types, Values, and Descriptions

KeyUsage Extension Type	Value	Description
digitalSignature	0	Used for entity authentication and to authenticate data origin integrity.
nonRepudiation	1	Used to protects against the signing entity falsely denying some action.
keyEncipherment	2	Used when the subject public key is used for key transport.
dataEncipherment	3	Used when the subject public key is used for enciphering data, other than cryptographic keys.
keyAgreement	4	Used when the subject public key is used for key agreement during SSL connection negotiation.
keyCertSign	5	Used when the subject public key is used for verifying a signature on certificates. May only be used in CA certificates.
cRLSign	6	Used when the subject public key is used for verifying a signature on certificate revocation lists.
encipherOnly	7	When the encipherOnly bit is asserted, the keyAgreement bit must also be set. When these two bits are set the subject public key may be used only for enciphering data while performing key agreement.
decipherOnly	8	As with the encipherOnly bit, the keyAgreement bit must also be set when decipherOnly is set. When these two bits (decipherOnly and keyAgreement) are set the subject public key may be used only for deciphering data while performing key agreement.

See Also: The Internet Engineering Task Force RFC #2459, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, for a complete description of the `KeyUsage` extension types at the following URL:

`http://www.ietf.org/rfc.html/`

When installing a certificate (user certificate or trusted certificate), Oracle Wallet Manager maps the `KeyUsage` extension values to Oracle PKI certificate usages as specified in Table 15–4 and Table 15–5.

Table 15–5 Oracle Wallet Manager Import of Trusted Certificates to an Oracle Wallet

KeyUsage Value	Critical?[1]	Usage
none	na	Importable.
Any combination excluding 5	Yes	Not importable.
Any combination excluding 5	No	Importable.
5 alone, or any combination including 5	na	Importable.

[1] If the `KeyUsage` extension is *critical*, the certificate cannot be used for other purposes.

You should obtain certificates from the certificate authority with the correct `KeyUsage` value for the required Oracle PKI certificate usage. A single wallet can contain multiple key pairs for the same usage. Each certificate can support multiple Oracle PKI certificate usages, as indicated by Table 15–4 and Table 15–5. Oracle PKI applications use the first certificate containing the required PKI certificate usage.

For example: For SSL usage, the first certificate containing the SSL Oracle PKI certificate usage is used.

If you do not have a certificate with SSL usage, then an `ORA-28885` error (`No certificate with required key usage found`) is returned.

16

Enabling SSL in the Infrastructure

This chapter provides instructions for enabling SSL in Infrastructure installations.

It contains these topics:

16.1 SSL Communication Paths in the Infrastructure

This section identifies all SSL communication paths used in the Oracle Application Server Infrastructure, and provides cross-references to the configuration instructions in component guides in the Oracle Application Server documentation library.

> **Note:** When you install Identity Management, you are prompted to select a mode for Oracle Internet Directory. The default mode is dual mode, which allows some components to access Oracle Internet Directory using non-SSL connections. If SSL mode was chosen during installation, then all installed components must use SSL when connecting to the directory.
>
> Before you begin SSL configuration, determine the Oracle Internet Directory mode. Start the `oidadmin` tool and view the SSL mode in Oracle Directory Manager. Go to the Directory Server and select **View Properties > SSL Settings**.

The following are the communication paths through the Oracle Application Server Infrastructure, and their related SSL configuration instructions:

- **Oracle HTTP Server to the OC4J_SECURITY instance**

  To configure the AJP communication over SSL, you must configure mod_oc4j's communication with the `iaspt` daemon. To do this, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Configuring mod_oc4j to Use SSL."

- **Oracle HTTP Server to iaspt (Port Tunneling) and then to the OC4J_SECURITY instance**

  To configure this connection path for SSL, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Understanding Port Tunneling."

- **The OC4J_SECURITY instance to Oracle Internet Directory**

To configure this connection path for SSL, follow the instructions in the *Oracle Application Server Single Sign-On Administrator's Guide*. It explains how to configure SSL communication between the browser and the OracleAS Single Sign-On server (section titled "Enable SSL on the Single Sign-On Middle Tier".

Oracle Delegated Administration Services is SSL-enabled after you configure the Oracle HTTP Server for SSL. The Oracle Delegated Administration Services communication to Oracle Internet Directory is always SSL-enabled; you do not have to perform any configuration tasks to accomplish this. (OracleAS Single Sign-On, Oracle Application Server Certificate Authority, and Oracle Delegated Administration Services communicate with Oracle Internet Directory in SSL mode by default.)

- **Oracle Directory Integration and Provisioning to Oracle Internet Directory and Oracle Internet Directory replication server to Oracle Internet Directory**

  As shown in Figure 16–1, a variety of components and communication paths may be configured for SSL. The following lists references to the instructions for each:

  – Communication between the Oracle Internet Directory Replication server and the Oracle Internet Directory server: *Oracle Application Server High Availability Guide*, section titled "Secure Sockets Layer (SSL) and Oracle Internet Directory Replication"

  – Communication between Oracle Directory Integration and Provisioning and the Oracle Internet Directory server: *Oracle Identity Management Integration Guide*, chapter titled "Oracle Directory Integration and Provisioning Server Administration"

- **The OC4J_SECURITY instance to the Metadata Repository database and Oracle Internet Directory to the Metadata Repository database**

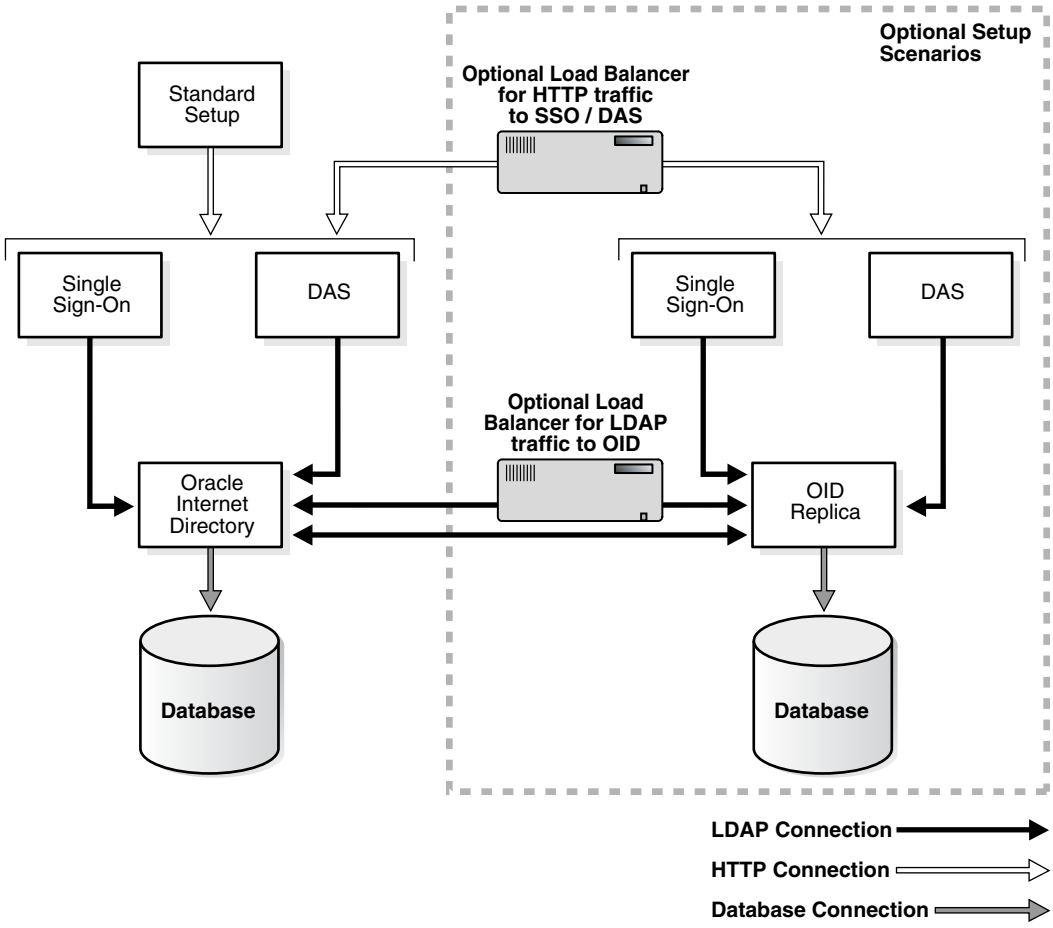
  If Oracle Internet Directory configured to accept SSL connections on the SSL port specified, you need only specify the SSL protocol and SSL port in the JDBC URL requesting an application, as follows:

  ldaps://host:sslport/...

  Note that when you are using a secure connection, you must add an s to the name of the protocol. For example, use ldaps instead of ldap.

  If Oracle Internet Directory is not configured to accept SSL connections on the SSL port, you must modify the configuration. See *Oracle Internet Directory Administrator's Guide*, section titled "Secure Sockets Layer (SSL) and the Directory."

Figure 16–1 Identity Management Components and SSL Connection Paths



16.2 Recommended SSL Configurations

The *Oracle Application Server Security Guide* discusses security concepts in detail and provides recommendations for configuring security in various configurations. The "Recommended Deployment Topologies" chapter presents sample architectures for Oracle Application Server 10*g* Release 2 (10.1.2) installation types. After you have identified the components on which you need to enable SSL, use the instructions in this chapter and Chapter 17, "Enabling SSL in the Middle Tier" to configure the components.

Configuring SSL in OracleAS Single Sign-On and Oracle Delegated Administration Services is typical in the recommended deployment topologies (as described in Section 16.3.1, "Configuring SSL for OracleAS Single Sign-On and Oracle Delegated Administration Services"). Configuring SSL in all Infrastructure communication paths is described in Section 16.1, "SSL Communication Paths in the Infrastructure".

16.3 Common SSL Configuration Tasks

This section provides references to the component guides in the Oracle Application Server documentation library that provide instructions for configuring SSL in individual components.

16.3.1 Configuring SSL for OracleAS Single Sign-On and Oracle Delegated Administration Services

Follow the instructions in the *Oracle Application Server Single Sign-On Administrator's Guide* to configure SSL communication between:

- The browser and the OracleAS Single Sign-On server (section titled "Enable SSL on the Single Sign-On Middle Tier")
- The OracleAS Single Sign-On server and the Oracle Internet Directory server (section titled "Configuring SSL Between the Single Sign-On Server and Oracle Internet Directory")

Oracle Delegated Administration Services is SSL-enabled after you configure the Oracle HTTP Server for SSL (as described in "Enable SSL on the Single Sign-On Middle Tier"). The Oracle Delegated Administration Services communication to Oracle Internet Directory is always SSL-enabled; you do not have to perform any configuration tasks to accomplish this.

16.3.2 Configuring SSL for Oracle Internet Directory

Instructions for configuring SSL communication in Oracle Internet Directory are provided in the following:

- *Oracle Internet Directory Administrator's Guide*, section titled "Secure Sockets Layer (SSL) and the Directory"
- *Oracle Internet Directory Administrator's Guide*, section titled "Configuring SSL Parameters"
- *Oracle Internet Directory Administrator's Guide*, section titled "Limitations of the Use of SSL in 10*g* (10.1.2)"

16.3.3 Configuring SSL for Oracle Internet Directory Replication Server and Oracle Directory Integration and Provisioning

As shown in Figure 16–1, a variety of components and communication paths may be configured for SSL. The following lists references to the instructions for each:

- Communication between the Oracle Internet Directory Replication server and the Oracle Internet Directory server: *Oracle Application Server High Availability Guide*, section titled "Secure Sockets Layer (SSL) and Oracle Internet Directory Replication"
- Communication between Oracle Directory Integration and Provisioning and the Oracle Internet Directory server: *Oracle Identity Management Integration Guide*, chapter titled "Oracle Directory Integration and Provisioning Server Administration"

16.3.4 Configuring SSL in the Identity Management Database

Follow the instructions in the *Oracle Application Server Single Sign-On Administrator's Guide*, section titled "Reconfigure the Identity Management Infrastructure Database" to configure SSL communication to the Identity Management database.

16.3.5 Additional SSL Configuration in the OC4J_SECURITY Instance

This section provides references to SSL configuration information for mod_oc4j and OC4J.

16.3.5.1 Configuring SSL from mod_oc4j to OC4J_SECURITY

To configure the AJP communication over SSL, you must configure mod_oc4j's communication with the `iaspt` daemon. To do this, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Enabling SSL between mod_oc4j and OC4J."

16.3.5.2 Using Port Tunneling from mod_oc4j to the OC4J_SECURITY Instance

To configure this connection path for SSL, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Understanding Port Tunneling."

16.3.5.3 Configuring JDBC/SSL (ASO support)

If Oracle Internet Directory configured to accept SSL connections on the SSL port specified, you need only specify the SSL protocol and SSL port in the JDBC URL requesting an application, as follows:

ldaps://host:sslport/...

Note that when you are using a secure connection, you must add an s to the name of the protocol. For example, use ldaps instead of ldap.

If Oracle Internet Directory is not configured to accept SSL connections on the SSL port, you must modify the configuration. See *Oracle Internet Directory Administrator's Guide*, section titled "Secure Sockets Layer (SSL) and the Directory."

16.3.6 SSL in Oracle Application Server Certificate Authority

Oracle Application Server Certificate Authority is SSL-enabled by default, so there are no configuration tasks associated with this component.

> **Tip:** OracleAS Certificate Authority simplifies the task of certificate provisioning for Identity Management users (certificate are automatically provisioned to OracleAS Single Sign-On-authenticated users).
>
> To enable certificate based authentication using OCA OracleAS Single Sign-On, see the *Oracle Application Server Certificate Authority Administrator's Guide*. To enable certificate-based authentication to OracleAS Single Sign-On, see the *Oracle Application Server Single Sign-On Administrator's Guide*.

16.3.7 Configuring SSL for Oracle Enterprise Manager 10*g*

Oracle Enterprise Manager 10*g* comprises two components, each of which can be configured for SSL communication: Grid Control Console and Application Server Control Console.

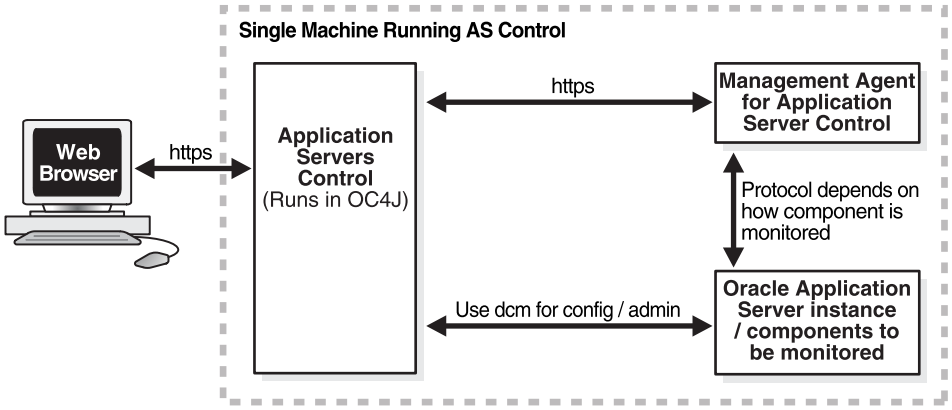
16.3.7.1 Configuring Security for the Grid Control

Follow the steps in the "Configuring Security (SSL and HTTPS) for Grid Control" in the *Oracle Enterprise Manager Advanced Configuration Guide*.

16.3.7.2 Configuring Security for the Application Server Control Console

The communication paths of interest in the SSL configuration of Application Server Control Console are shown in Figure 16–2, and are labeled **https**.

Figure 16–2 SSL Connection Paths in Oracle Enterprise Manager 10g



To secure the communications between the Web browser and the Application Server Control Console, and between the Application Server Control Console and the Management Agent, you can use the `emctl secure iasconsole` command-line utility. See Section A.4, "Configuring Security for Application Server Control Console" for instructions.

The communication (that is, obtaining monitoring information and configuration and administration tasks) between the Management Agent and the application server being monitored, and the Application Server Control and the application server being administered is not affected in any way when you use the `emctl secure iasconsole` utility. Those communication paths are not secured until you perform the application server security configuration steps for the particular path. Instructions on configuring SSL for application server communication paths are provided in Section 16.1, "SSL Communication Paths in the Infrastructure" and Section 17.1, "SSL Communication Paths in the Middle Tier". In addition to their SSL configuration, some components also require that you perform configuration changes to the application Server Control's Management Agent. The instructions for these changes are included with the instructions for enabling the components for SSL.

For information and instructions on configuring SSL in the Application Server Control Console, see Section A.4, "Configuring Security for Application Server Control Console".

17

Enabling SSL in the Middle Tier

This chapter provides instructions for enabling SSL in Oracle Application Server middle-tier installations.

It contains these topics:

17.1 SSL Communication Paths in the Middle Tier

This section identifies all SSL communication paths used in the Oracle Application Server middle-tier installation types, and provides cross-references to the configuration instructions in component guides in the Oracle Application Server documentation library.

The following are communication paths through the Oracle Application Server middle tier, and their related SSL configuration instructions:

> **Note:** In most cases, SSL can be configured with the SSL Configuration Tool. For more information, see Chapter 14, "Using the SSL Configuration Tool".

- **External Clients or Load Balancer to Oracle HTTP Server**

  To configure the Oracle HTTP Server for SSL, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Enabling SSL."

- **External Clients or Load Balancer to OracleAS Web Cache**

  To configure OracleAS Web Cache for SSL, follow the instructions in "Configuring OracleAS Web Cache for HTTPS Requests" in the *Oracle Application Server Web Cache Administrator's Guide*.

- **OracleAS Web Cache to Oracle HTTP Server**

  To configure OracleAS Web Cache for SSL, follow the instructions in "Configuring OracleAS Web Cache for HTTPS Requests" in the *Oracle Application Server Web Cache Administrator's Guide*.

- **Oracle HTTP Server to OC4J Applications (AJP)**

  To configure the AJP communication over SSL, you must configure mod_oc4j's communication with the `iaspt` daemon. To do this, follow the instructions in the

Oracle HTTP Server Administrator's Guide, section titled "Configuring mod_oc4j to Use SSL."

- **Oracle HTTP Server to iaspt and then to OC4J**

  To configure this connection path for SSL, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Understanding Port Tunneling."

- **OC4J (the JAAS provider) to Oracle Internet Directory**

  To configure the provider, follow the instructions in the *Oracle Application Server Containers for J2EE Security Guide*. To configure the provider for SSL, set the `SSL_ONLY_FLAG` to `true`.

- **OC4J to the database (ASO)**

  If Oracle Internet Directory configured to accept SSL connections on the SSL port specified, you need only specify the SSL protocol and SSL port in the JDBC URL requesting an application, as follows:

  ```
  ldaps://host.sslport/...
  ```

  Note that when you are using a secure connection, you must add an s to the name of the protocol. For example, use ldaps instead of ldap.

  If Oracle Internet Directory is not configured to accept SSL connections on the SSL port, you must modify the configuration. See *Oracle Internet Directory Administrator's Guide*, section titled "Secure Sockets Layer (SSL) and the Directory."

- **ORMI (Oracle Remote Method Invocation, a custom wire protocol) over HTTP and HTTP over SSL**

  ORMI over SSL is not supported. To configure similar functionality, you can configure ORMI over HTTP, and then configure HTTP for SSL.

  See the *Oracle Application Server Containers for J2EE Services Guide*, section titled "Configuring ORMI Tunnelling Through HTTP" for instructions on how to configure ORMI/HTTP.

- **SSL into standalone OC4J (HTTPS)**

  To configure this connection path for SSL, follow the instructions in the *Oracle Application Server Containers for J2EE Security Guide*, section titled "Configuring SSL in OC4J" explains how to use SSL to secure communication between clients and an OC4J instance.

- **OracleAS Portal Parallel Page Engine (the servlet in the OC4J_PORTAL instance) to OracleAS Web Cache (HTTPS)**

  To configure this connection path for SSL, follow the instructions in the *Oracle Application Server Containers for J2EE Security Guide*, section titled "Configuring SSL in OC4J."

17.2 Recommended SSL Configurations

The *Oracle Application Server Security Guide* discusses security concepts in detail and provides recommendations for configuring security in various configurations. The "Recommended Deployment Topologies" chapter presents sample architectures for Oracle Application Server 10*g* Release 2 (10.1.2) installation types. After you have identified the components on which you need to enable SSL, use the instructions in this chapter and Chapter 16, "Enabling SSL in the Infrastructure" to configure the components.

17.3 Common SSL Configuration Tasks for the Middle Tier

This section identifies some commonly used SSL configurations in the Oracle Application Server middle-tier installation types, and provides cross-references to the configuration instructions in component guides in the Oracle Application Server documentation library.

17.3.1 Enabling SSL in OracleAS Web Cache

OracleAS Web Cache is part of all Oracle Application Server middle-tier installations. To configure it for SSL, follow the instructions in chapter "Configuring OracleAS Web Cache for HTTPS Requests" in the *Oracle Application Server Web Cache Administrator's Guide*.

A script, `SSLConfigTool`, automates the SSL configuration of the following:

- HTTPS listening ports and wallet location for the cache
- HTTPS operations ports for the cache
- Site for HTTPS requests
- HTTPS port and wallet location for the origin server
- Site-to-server mapping

For instructions on using this script, seeChapter 14, "Using the SSL Configuration Tool".

17.3.2 Enabling SSL in the Oracle HTTP Server

Oracle HTTP Server is part of all Oracle Application Server middle-tier installations. To configure it for SSL, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Enabling SSL."

A script, `SSLConfigTool`, automates the setting of the SSL parameters in the `httpd.conf` file. For more information about this script, see Chapter 14, "Using the SSL Configuration Tool".

17.3.3 Enabling SSL in OC4J

To configure SSL connections to OC4J clients, follow the instructions in the *Oracle Application Server Containers for J2EE Security Guide* section titled "Oracle HTTPS for Client Connections."

17.3.3.1 Configuring SSL from Oracle HTTP Server to OC4J

To configure the AJP communication over SSL, you must configure `mod_oc4j`'s communication with the `iaspt` daemon. To do this, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Enabling SSL between mod_oc4j and OC4J."

17.3.3.2 Using Port Tunneling (iaspt) from Oracle HTTP Server to OC4J

To configure this connection path for SSL, follow the instructions in the *Oracle HTTP Server Administrator's Guide*, section titled "Understanding Port Tunneling."

17.3.3.3 Configuring ORMI/HTTP SSL

ORMI over SSL is not supported. To configure similar functionality, you can configure ORMI over HTTP, and then configure HTTP for SSL.

See the *Oracle Application Server Containers for J2EE Services Guide*, section titled "Configuring ORMI Tunnelling Through HTTP" for instructions on how to configure ORMI/HTTP.

17.3.3.4 Configuring Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider for SSL with Oracle Internet Directory

To configure the provider, follow the instructions in the *Oracle Application Server Enterprise Deployment Guide*, section titled "Configuring Application Authentication and Authorization." To configure the provider for SSL, set the `SSL_ONLY_FLAG` to `true`.

17.3.3.5 Configuring Oracle HTTP Server for SSL

The *Oracle Application Server Containers for J2EE Security Guide*, section titled "Enabling SSL in OC4J" explains how to configure Oracle HTTP Server for SSL.

17.3.3.6 Configuring SSL in Standalone OC4J Installations

The *Oracle Application Server Containers for J2EE Security Guide*, section titled "Enabling SSL in OC4J" explains how to use SSL to secure communication between clients and an OC4J instance.

17.3.4 Enabling SSL in J2EE and Web Cache Installations

Depending on your security needs and the configuration of the Oracle Application Server J2EE and Web Cache installation, you may implement secure communication in one or more of the installed components. Configuring the first listener (whether it is OracleAS Web Cache or the Oracle HTTP Server) may be sufficient.

To configure the Oracle HTTP Server for SSL, follow the steps in "Enabling SSL for Oracle HTTP Server" in the *Oracle HTTP Server Administrator's Guide*.

To configure OracleAS Web Cache for SSL, follow the instructions in "Configuring OracleAS Web Cache for HTTPS Requests" in the *Oracle Application Server Web Cache Administrator's Guide*.

A script called `SSLConfigTool` is provided to automate some of the configuration tasks. For instructions on using this script, see Chapter 14.

17.3.5 Enabling SSL in Virtual Hosts

You can use virtual hosts to deploy multiple Web sites on a single Oracle HTTP Server (for example, to make an application available over the HTTP protocol and the HTTPS protocol).

The *Oracle Application Server Single Sign-On Administrator's Guide*, section titled "Configuring mod_osso with Virtual Hosts" contains instructions on configuring an SSL virtual host to be protected by mod_osso. You cannot use name-based virtual hosting. You must use IP-based or port-based virtual hosting.

The scenario presented assumes that the following conditions are in effect:

- The host name of the application middle tier is app.mydomain.com (replace this name with the host name of your application middle tier).
- The middle tier is already configured as a non-SSL partner application (this is typically done during installation).
- The default SSL port number of the application middle tier is 4443.

17.3.6 Enabling SSL in OracleBI Discoverer

The *Oracle Business Intelligence Discoverer Configuration Guide* explains how to configure OracleBI Discoverer for SSL.

For a discussion of Oracle Application Server Framework Security, including the SSL protocols for Oracle Business Intelligence, see the *Oracle Business Intelligence Discoverer Configuration Guide*, section titled "Using Discoverer with OracleAS Framework Security."

For information on implementing SSL in OracleBI Discoverer, see the *Oracle Business Intelligence Discoverer Configuration Guide*, section titled "What is HTTPS and why should I use it?"

For instructions on enabling OracleBI Discoverer for SSL, see the *Oracle Business Intelligence Discoverer Configuration Guide*, section titled "About running Discoverer over HTTPS."

17.3.7 Enabling SSL in OracleAS Wireless

For instructions on configuring SSL in OracleAS Wireless, see the Wireless Security chapter in the *Oracle Application Server Wireless Administrator's Guide*. The section titled "Site Administration" explains how to use the System Manager HTTP, HTTPS configuration page in Oracle Enterprise Manager 10*g* to configure the Wireless site's proxy server settings, URLs, and SSL certificates.

17.3.8 Enabling SSL in OracleAS Portal

OracleAS Portal uses a number of different components for HTTP communication (such as the Parallel Page Engine, Oracle HTTP Server, and OracleAS Web Cache), each of which may act as a client or server. As a result, each component in the Oracle Application Server middle tier may be configured individually to use the HTTPS protocol instead of HTTP.

These components' interaction with OracleAS Portal involves a number of distinct network hops. These include:

- Between the client browser and the entry point of the OracleAS Portal environment; the entry point can be OracleAS Web Cache or a network edge hardware device such as a reverse proxy or SSL accelerator
- Between OracleAS Web Cache and the Oracle HTTP Server of the Oracle Application Server middle tier
- Between the client browser and the Oracle HTTP Server of the OracleAS Single Sign-On/Oracle Internet Directory (or Infrastructure) tier
- A loop back connection between the Parallel Page Engine (PPE) on the middle tier and OracleAS Web Cache or the front-end reverse proxy
- Between the Parallel Page Engine (PPE) and the Remote Web Provider producing Portlet content
- Between the OracleAS Portal infrastructure and the Oracle Internet Directory server

The following sections in the *Oracle Application Server Portal Configuration Guide* provide an overview of the most common SSL configurations for OracleAS Portal and instructions for implementing them:

- SSL to OracleAS Single Sign-On: Follow the instructions in the *Oracle Application Server Portal Configuration Guide* to configure a secure connection to OracleAS Single Sign-On.
- SSL to OracleAS Web Cache: Follow the instructions in the *Oracle Application Server Portal Configuration Guide* to configure a secure connection to OracleAS Web Cache.
- SSL throughout OracleAS Portal: Follow the instructions in the *Oracle Application Server Portal Configuration Guide* to configure secure connections throughout OracleAS Portal.
- External SSL with non- SSL within Oracle Application Server: Follow the instructions in the *Oracle Application Server Portal Configuration Guide* to configure OracleAS Portal such that the site is externally accessible through SSL URLs, with the Oracle Application Server running in non-SSL mode.

Note: For general information on securing OracleAS Portal, see the *Oracle Application Server Portal Configuration Guide* (Chapter 6, Securing OracleAS Portal).

17.3.9 Configuring SSL for Oracle Enterprise Manager 10*g*

See Section 16.3.7, "Configuring SSL for Oracle Enterprise Manager 10g" on page 16-5 in Chapter 16, "Enabling SSL in the Infrastructure".

18

Troubleshooting SSL

This chapter lists common questions and errors related to SSL.

It contains these topics:

18.1 Name-Based Virtual Hosting and SSL

You cannot use name-based virtual hosting with SSL. This is a limitation of SSL.

If you need to configure multiple virtual hosts with SSL, here are some possible workarounds:

- Use IP-based virtual hosting. To do this, you configure multiple IP addresses for your computer, and map each IP address to a different virtual name.
- If you are willing to use non-standard port numbers, you can associate the same IP with different names, but you must configure each name with a different port number (for example, `name1:443`, `name2:553`). This enables you to use the same IP, but you have to use non-standard port numbers. Only one name can use the standard 443 port; other names must use other port numbers.

18.2 Common ORA Errors Related to SSL

You may need to enable Oracle Net tracing to determine the cause of an error. For information about setting tracing parameters for Oracle Net, see *Oracle Database Net Services Administrator's Guide*.

ORA-28759: Failure to Open File

Cause: The system could not open the specified file. Typically, this error occurs because the Oracle wallet cannot be found.

Action: Check the following:

- Ensure that the Oracle wallet is located either in the default location (`ORACLE_HOME/Apache/Apache/conf/ssl.wlt/default`) or in the location specified by the `SSLWallet` directive in the `ORACLE_HOME/Apache/Apache/conf/ssl.conf` file. This should be the same directory location where you saved the wallet.
- Enable Oracle Net tracing to determine the name of the file that cannot be opened and the reason.

- Ensure that auto login was enabled when you saved the Oracle wallet. See Section 15.1.4.14, "Using Auto Login" for details.

ORA-28786: Decryption of Encrypted Private Key Failure

Cause: An incorrect password was used to decrypt an encrypted private key. Frequently, this happens because an auto login wallet is not being used.

Action: Use Oracle Wallet Manager to turn the auto login feature on for the wallet. Then re-save the wallet. See Section 15.1.4.14, "Using Auto Login".

ORA-28858: SSL Protocol Error

Cause: This is a generic error that can occur during SSL handshake negotiation between two processes.

Action: Enable Oracle Net tracing and attempt the connection again to produce trace output. Then contact Oracle customer support with the trace output.

ORA-28859 SSL Negotiation Failure

Cause: An error occurred during the negotiation between two processes as part of the SSL protocol. This error can occur when two sides of the connection do not support a common cipher suite.

Action: Ensure that the cipher suites configured on Oracle HTTP Server and on the client (which is the browser) are compatible for both client and server.

To check the cipher suites configured on Oracle HTTP Server, check the `SSLCipherSuite` directive in the `ORACLE_HOME/Apache/Apache/conf/ssl.conf` file.

To check the cipher suites configured on your browser, see the documentation for your browser. Each type of browser has its own way of setting the cipher suite.

You should also ensure that the SSL versions on both the client and the server match, or are compatible. For example, if the server accepts only SSL 3.0 and the client accepts only TLS 1.0, then the SSL connection will fail.

ORA-28862: SSL Connection Failed

Cause: This error occurred because the peer closed the connection.

Action: Check the following:

- Ensure that the Oracle wallet is located either in the default location (`ORACLE_HOME/Apache/Apache/conf/ssl.wlt/default`) or in the location specified by the `SSLWallet` directive in the `ORACLE_HOME/Apache/Apache/conf/ssl.conf` file. This should be the same directory location where you saved the wallet.
- Check that the cipher suites are compatible for both client and server. See "ORA-28859 SSL Negotiation Failure" for details on how to check the cipher suite.
- Check that the names of the cipher suites are spelled correctly.
- Ensure that the SSL versions on both the client and the server match, or are compatible. Sometimes this error occurs because the SSL version specified on the server and client do not match. For example, if the server accepts only SSL 3.0 and the client accepts only TLS 1.0, then the SSL connection will fail.
- For more diagnostic information, enable Oracle Net tracing on the peer.

ORA-28865: SSL Connection Closed

Cause: The SSL connection closed because of an error in the underlying transport layer, or because the peer process quit unexpectedly.

Action: Check the following:

- Ensure that the SSL versions on both the client and the server match, or are compatible. Sometimes this error occurs because the SSL version specified on the server and client do not match. For example, if the server accepts only SSL 3.0 and the client accepts only TLS 1.0, then the SSL connection will fail.
- If you are using a Diffie-Hellman anonymous cipher suite and the `SSLVerifyClient` directive is set to `require` in the `ssl.conf` file, then the client does not pass its certificate to the server. When the server does not receive the client's certificate, the server cannot authenticate the client so the connection is closed. To resolve this, use a different cipher suite, or set the `SSLVerifyClient` directive to `none` or `optional`.

  See "ORA-28859 SSL Negotiation Failure" for details on how to check the cipher suite.
- Enable Oracle Net tracing and check the trace output for network errors.

ORA-28868: Peer Certificate Chain Check Failed

Cause: When the peer presented the certificate chain, it was checked and that check failed. This failure can be caused by a number of problems, including:

- One of the certificates in the chain is expired.
- A certificate authority for one of the certificates in the chain is not recognized as a trust point.
- The signature in one of the certificates cannot be verified.

Action: Follow the instructions in Section 15.1.4.3, "Opening an Existing Wallet" to use Oracle Wallet Manager to open your wallet, and check the following:

- Ensure that all of the certificates installed in your wallet are current (not expired).
- Ensure that a certificate authority's certificate from your peer's certificate chain is added as a trusted certificate in your wallet. See Section 15.1.5.2.1, "Importing a Trusted Certificate" to use Oracle Wallet Manager to import a trusted certificate.

ORA-28885: No certificate with the required key usage found.

Cause: Your certificate was not created with the appropriate X.509 Version 3 key usage extension.

Action: Use Oracle Wallet Manager to check the certificate's key usage. See Table 15–4, " X.509 Version 3 KeyUsage Extension Types, Values, and Descriptions".

ORA-29024: Certificate Validation Failure

Cause: The certificate sent by the other side could not be validated. This may occur if the certificate has expired, has been revoked, or is invalid for another reason.

Action: Check the following:

- Check the certificate to determine whether it is valid. If necessary, get a new certificate, inform the sender that her certificate has failed, or resend.

- Check to ensure that the server's wallet has the appropriate trust points to validate the client's certificate. If it does not, then use Oracle Wallet Manager to import the appropriate trust point into the wallet. See Section 15.1.5.2.1, "Importing a Trusted Certificate" for details.
- Ensure that the certificate has not been revoked and that certificate revocation list (CRL) checking is enabled. See Section 15.2.5, "Managing Certificate Revocation Lists (CRLs) with the orapki Utility".

ORA-29223: Cannot Create Certificate Chain

Cause: A certificate chain cannot be created with the existing trust points for the certificate being installed. Typically, this error is returned when the peer does not give the complete chain and you do not have the appropriate trust points to complete it.

Action: Use Oracle Wallet Manager to install the trust points that are required to complete the chain. See Section 15.1.5.2.1, "Importing a Trusted Certificate".

Part V

Backup and Recovery

Backup and recovery refers to the various strategies and procedures involved in guarding against hardware failures and data loss, and reconstructing data should loss occur. This part describes how to back up and recover Oracle Application Server.

This part contains the following chapters:

19

Introduction to Backup and Recovery

This chapter provides information on getting started with Oracle Application Server backup and recovery.

It contains the following topics:

19.1 Philosophy of Oracle Application Server Backup and Recovery

This section introduces the philosophy for backing up and recovering your Oracle Application Server environment. An Oracle Application Server environment can consist of different components and configurations. To determine which components and configurations best meet your requirements, refer to the *Oracle Application Server Installation Guide* and *Oracle Application Server Concepts*.

A typical Oracle Application Server environment contains:

- An **Infrastructure installation** that contains Identity Management and a Metadata Repository
- One or more **middle-tier installations** (J2EE and Web Cache, Portal and Wireless, or Business Intelligence and Forms) that may use the Infrastructure

The installations in an Oracle Application Server environment are interdependent in that they contain configuration information, applications, and data that are kept in sync. For example, when you perform a configuration change, you might update configuration files in the middle-tier installation and Infrastructure; when you deploy an application, you might deploy it to all middle-tier installations; and when you perform an administrative change on a middle-tier installation, you might update data in the Metadata Repository.

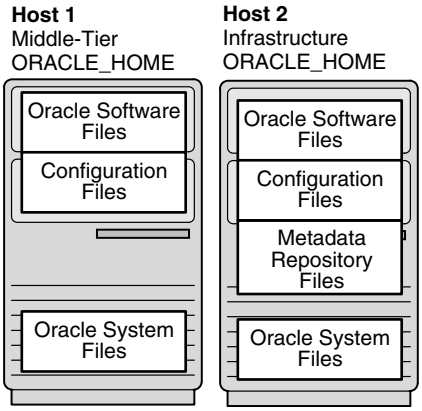
It is, therefore, important to consider your entire Oracle Application Server environment when performing backup and recovery. For example, you should not back up your middle-tier installation on Monday and your Infrastructure on Tuesday. If you lose files in your middle-tier installation, you could restore it to Monday's state. However, your Infrastructure would be in its current state—out of sync with the middle tier. And, because you backed up the Infrastructure on Tuesday, you would

have no means of restoring it to a state in sync with Monday's middle-tier installation. You would not be able to restore your environment to a consistent state.

Instead, you should back up your entire Oracle Application Server environment at once. Then, if a loss occurs, you can restore your entire environment to a consistent state.

For the purposes of backup and recovery, you can divide your Oracle Application Server environment into different types of files, as shown in Figure 19–1.

Figure 19–1 Types of Files for Oracle Application Server Backup and Recovery



The types of files for backup and recovery are:

- **Oracle software files**

  These are static files such as binaries and libraries. They reside in the middle-tier and Infrastructure Oracle homes. They are created at installation time.

- **Configuration files**

  These files contain configuration information and deployed applications. They reside in the middle-tier and Infrastructure Oracle homes. They are created at installation or runtime and are updated during the normal operation of your application server.

  There are two types of configuration files: configuration files managed by Distributed Configuration Management (DCM) and configuration files not managed by DCM. The files managed by DCM contain configuration information for OHS, OC4J, OPMN, Logloader, and JAZN. Components not managed by DCM include Portal and Wireless. The Backup and Recovery Tool creates an archive for each group of these components. The archives are stored in the same directory. In order to maintain synchronicity, the configuration files archive and the DCM-managed configuration files archive are paired by a unique timestamp. During restores, you specify the timestamp and the tool uses the timestamp to identify and restore both archives.

- **Metadata Repository files**

  These are the datafiles and control files that make up your Metadata Repository. They reside in the Infrastructure Oracle home. They are created at installation time and are updated during the normal operation of your application server.

- **Oracle system files**

  These files may be in the `/var/opt/oracle` or `/etc` directory, and the `oraInventory` directory. They exist on each host in your Oracle Application

Server environment. They usually reside outside of your Oracle Application Server installations, although the `oraInventory` directory may be in an Oracle home. They are created and updated by Oracle Universal Installer at installation time and contain information about your installations. On Windows, some registries are created by the installer.

The strategies and procedures in this book involve backing up and recovering these different types of files in a manner that maintains your Oracle Application Server environment in a consistent state.

Note: Your Oracle Application Server environment contains additional files to those mentioned in this section, such as log files; database configuration files, including `orapwd`, and `spfile/pfile`; and additional files you may deploy in the Oracle home, such as static HTML files and CGI scripts. You can add any of these files to the backup list.

19.2 Overview of the Backup Strategy

This section describes the backup strategy used in this book. It contains the following topics:

- Types of Backups
- Recommended Backup Strategy

19.2.1 Types of Backups

The Oracle Application Server backup strategy involves two types of backups:

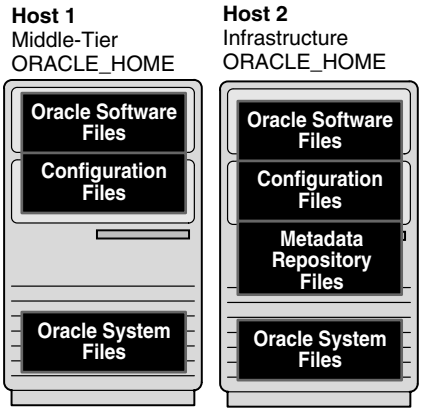
- Image Backup
- Instance Backup

Image Backup

An image backup of an Oracle Application Server instance includes the Oracle Home directory of that instance, the OraInventory directory, the oratab file, and Windows registries on that node and finally a cold instance backup of that Oracle Application Server instance. The Oracle Home directory contains all the binary files, executables, initialization files, configuration files, log files, and so forth of the OracleAS instance and of all components and deployed applications in that instance. The OraInventory directory contains the installation information for the instance.

In Figure 19–2, the files that are backed up during an image backup of an Oracle Application Server environment are shaded. An image backup includes everything necessary to restore the initial installation of your Oracle Application Server environment including the Metadata Repository if the instance is an Infrastructure. To avoid an inconsistent backup, do not make any configuration changes until the backup is completed for all Oracle Application Server instances.

Figure 19–2 Files Backed Up in an Image Backup of an Oracle Application Server Environment



Instance Backup

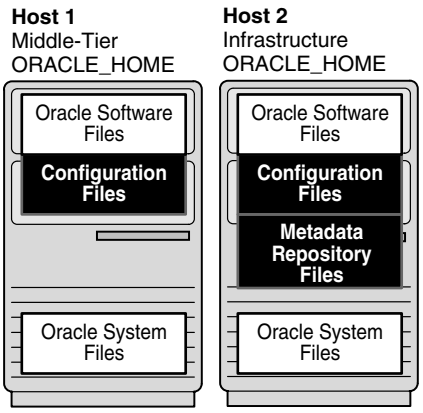
The contents of an Instance Backup depends on the type of Oracle Application Server instance that is being backed up. There are three types of instance backups:

- **Oracle Application Server Infrastructure Instance Backup**—The Backup and Recovery Tool first backs up the entire Oracle Metadata Repository database. The database contains the product metadata schemas for Oracle Application Server mid-tier components such as Portal and Wireless. If Identity Management is also installed in this instance, then the database may also contain the configuration information of the Oracle Internet Directory (OID) and Oracle Application Server Single Sign-On (SSO) components of the Identity Management Service. Next, the tool requests Distributed Configuration Management (DCM) to create and export a consistent archive (jar file) of the configuration schemas from the DCM repository for DCM-managed components like Oracle HTTP Server (OHS) and OracleAS Containers for J2EE (OC4J). Then, the tool adds the archive file to the backup. Finally, for each configured component, the tool backs up all the local copies of the configuration files specified for that component in its backup input file. For a list of component backup files, see Table 19–1, " Oracle Application Server Component Backup Input Files".
- **Oracle Application Server Middle-tier Instance Backup**—Contains the configuration information of all its Oracle Application Server components and deployed applications. Some of these components, like Portal, Wireless, Integration B2B, and Oracle Business Intelligence Discoverer are not managed by DCM. They have their product metadata in the Oracle Application Server Metadata Repository database which is backed up by the Backup and Recovery Tool in the Oracle Application Server Infrastructure instance. Other components like the OHS, OC4J, Oracle Process Management and Notification Server (OPMN) and Java Authentication and Authorization Service (JAZN) are managed by DCM. The configuration information, for these components and the deployed J2EE applications, is stored in the DCM repository which can be a file-based repository or a database repository. As in an Infrastructure instance backup, the tool requests DCM to create and export a consistent archive of the configuration schemas from the DCM repository and adds the archive to the mid-tier instance backup. The tool also backs up all the local copies of the configuration files specified for each configured mid-tier component in its backup input file.
- **Oracle Application Server Metadata Repository Creation Assistant**—Instead of creating a new Oracle Application Server Infrastructure instance, you can install the Oracle Application Server Metadata Repository in an existing Oracle database

using the OracleAS Metadata Repository Creation Assistant (MRCA, previously called RepCA). The existing database may also have Oracle Internet Directory and SSO installed for Identity Management. Since there are no other OracleAS components in an OracleAS MRCA instance, the Backup and Recovery Tool backs up only the existing database and not any other local configuration files.

In Figure 19–3, the files that are backed up during an instance backup are shaded. This type of backup involves saving the configuration information, and metadata across your entire Oracle Application Server environment at the same point in time. To avoid an inconsistent backup, do not make any configuration changes until the backup completes for all Oracle Application Server instances.

Figure 19–3 Files Backed Up in an Instance Backup



19.2.2 Oracle Application Server Component Backup Input Files

Each Oracle Application Server component has a backup input file which contains a list of all the local configuration files that should be backed up for that component. In a backup operation, if a component is installed and configured, the Backup and Recovery Tool invokes the component's backup input file to determine what files to backup. A component backup input file has the file extension `.inp` and resides in the *`Oracle_Home`*`/backup_restore/config` directory. The following is a list of all the component backup input files that can reside in the directory:

Table 19–1 Oracle Application Server Component Backup Input Files

Component Name	Backup Input File
Content Management SDK	config_cmsdk_files.inp
Delegated Administration Services	config_das_files.inp
Distributed Configuration Management Service	config_dcm_files.inp
Directory Integration and Provisioning	config_dip_files.inp
OracleBI Discoverer	config_discoverer_files.inp
OracleAS Guard for Disaster Recovery	config_dsa_files.inp
Oracle Enterprise Manager	config_em_files.inp
List of files to be excluded during backup	config_exclude_files.inp
Oracle Forms	config_forms_files.inp
OracleAS installation information	config_install_files.inp
Oracle BPEL Process Manager	config_ip_files.inp

Table 19–1 (Cont.) Oracle Application Server Component Backup Input Files

Component Name	Backup Input File
Business Integration Application Adapters	config_IPadapters_files.inp
Business Integration B2B	config_IPb2b_files.inp
BPEL Process Analytics	config_IPbam_files.inp
Business Integration Process Manager	config_IPbpm_files.inp
Business Integration Interconnect	config_IPinterconnect_files.inp
Java Object Cache	config_javaobjcache_files.inp
Oracle Enterprise Manager Log Loader	config_logloader_files.inp
Extra miscellaneous files to be backed up	config_misc_files.inp
OracleAS Containers for J2EE applications	config_oc4j_files.inp
OracleAS Certificate Authority	config_oca_files.inp
OracleAS HTTP Server	config_ohs_files.inp
Oracle Internet Directory	config_oid_files.inp
Oracle Process Management and Notification Server	config_opmn_files.inp
OracleAS Personalization	config_personalization_files.inp
OracleAS Portal	config_reports_files.inp
OracleAS Single Sign-ON	config_sso_files.inp
TopLink	config_toplink_files.inp
Oracle Ultra Search for OracleAS Infrastructure	config_ultrasearch_infra_files.inp
Oracle Ultra Search for OracleAS Mid-tier	config_ultrasearch_mid_files.inp
OracleAS Web Cache	config_webcache_files.inp
OracleAS Wireless	config_wireless_files.inp

19.2.3 Recommended Backup Strategy

This section outlines the recommended strategy for performing backups. Using this strategy ensures that you will be able to perform the recovery procedures in this book.

- **Perform a complete image backup.**

  Immediately after you install Oracle Application Server, you should perform a complete image backup for each node in your Oracle Application Server environment. This backup contains everything you need in order to restore each node to its initial state. It serves as a baseline for all subsequent online backups.

- **Perform instance backups on a regular basis.**

  After every administrative change, or, if this is not possible, on a regular basis, perform an instance backup of your Oracle Application Server environment. This enables you to restore your environment to a consistent state as of the time of your most recent configuration and metadata backup. To avoid an inconsistent backup, do not make any configuration changes until backup completes for all Oracle Application Server instances.

  See Also: Appendix G, "Examples of Administrative Changes" to learn more about administrative changes

- **After a major change, perform a new complete image backup.**

  If you make a major change to your Oracle Application Server environment, perform a new complete image backup. This backup will serve as the basis for subsequent online backups.

  Perform a new complete image backup after:

  - An operating system software upgrade
  - An Oracle Application Server software upgrade or patch application

  If you decide to back out an upgrade or patch, revert back to your last complete image backup. You can then apply any instance backups that occurred between the software upgrade or patch and the last complete image backup of your Oracle Application Server environment. Restoring an instance backup without restoring the last complete image backup might mix old configuration files with newly upgraded software that might not be compatible.

- **Perform instance backups on a regular basis.**

  After you establish a new complete image backup of your Oracle Application Server environment, continue to perform instance backups on a regular basis.

19.3 Overview of Recovery Strategies

There are two types of Oracle Application Server recovery strategies used in this book:

- Recovery Strategies for Data Loss, Host Failure, or Media Failure (Critical)
- Recovery Strategies for Process Crashes or System Outages (Non-Critical)

Recovery Strategies for Data Loss, Host Failure, or Media Failure (Critical)

These strategies enable you to recover from critical failures that involve actual data loss. Depending on the type of loss, they can involve recovering any combination of the following types of files:

- Oracle software files
- Configuration files
- Metadata Repository files
- Oracle system files

In all cases, these strategies involve making sure your state is consistent across all installations.

Recovery Strategies for Process Crashes or System Outages (Non-Critical)

These strategies involve restarting processes that have stopped or failed. They do not involve restoring data. They are included in this book for completeness.

19.4 What Is the Oracle Application Server Backup and Recovery Tool?

The Oracle Application Server Backup and Recovery Tool (OracleAS Backup and Recovery Tool) is an application. You can use the tool to backup and recover the following types of files:

- Configuration files in the middle-tier and Infrastructure Oracle home
- Identity Management/Metadata Repository files

The OracleAS Backup and Recovery Tool is installed by default whenever you install Oracle Application Server. The tool is installed in the *Oracle_Home*/backup_restore directory. For instructions on how to install the tool manually, see Section 20.1.1, "Manually Installing the OracleAS Backup and Recovery Tool"

19.5 Assumptions and Restrictions

The following assumptions and restrictions apply to the backup and recovery procedures in this book:

- The Backup and Recovery Tool is not backward compatible with previous releases of the Backup and Recovery Tool. Archives, created using previous versions of the tool, are not recoverable by the current version of the tool.
- If you convert a database repository to a file-based repository, perform a full backup of the file-based repository and configuration files immediately after the conversion because the Backup and Recovery Tool cannot recover any database repository backups after the conversion.
- The following installation types are supported:
  - J2EE and Web Cache
  - J2EE Standalone
  - Portal and Wireless
  - Business Intelligence and Forms
  - Infrastructure (Identity Management and Metadata Repository)
  - Infrastructure (Identity Management only)
  - Infrastructure (Metadata Repository only)
  - OracleAS TopLink (Standalone or installed into a middle-tier Oracle home)
  - Oracle BPEL Process Analytics
  - Oracle Content Management Software Development Kit
  - Integration B2B
  - Business Intelligence CD
  - Cold Failover Cluster (Infrastructure)
  - Cold Failover Cluster (Middle Tier)
  - Identity Management (Oracle Internet Directory + Single Sign-On)
  - Identity Management (Oracle Internet Directory)
  - Identity Management (Single Sign-On)
  - Identity Management High Availability
  - MRCA 10.1.0.x (Real Application Clusters Database)
  - MRCA 9.2.0.x
- **Alert:** When a Metadata Repository is created by running the Oracle Application Server Metadata Repository Creation Assistant (OracleAS Metadata Repository Creation Assistant) on an existing database, the OracleAS Backup and Recovery Tool performs backup and restore on the entire database not only on the Metadata Repository.

- For a component upgrade from 9.0.4 to 10.1.2, you must run the `configure` command from Oracle Application Server Control Console or the command line to incorporate the new components.
- If you are using OracleAS Cold Failover Cluster or Disaster Recovery, refer to the High Availability Guide for special considerations.
- On Windows, if you want to store backups on a remote file system, you must create a local mapped drive and specify it as the backup storage directory. For example, if `Z:\ASbackups` is the mapped drive for backups, then configuration files and repository backups should have `Z:\ASbackups` as their backup directory.

19.6 Roadmap for Getting Started with Backup and Recovery

This section provides a roadmap for getting started with Oracle Application Server backup and recovery.

1. **Learn About Database Backup and Recovery.**

   The Oracle Application Server environment includes the Metadata Repository—an Oracle Database 10*g* database. Performing backup and recovery on Oracle Application Server includes performing backup and recovery of a database. It is, therefore, important for application server administrators to understand database backup and recovery.

   If you are not experienced with database backup and recovery, Oracle recommends you read *"Oracle Backup and Recovery Basics"*, which is available in the Oracle Database 10*g* document library.

   In particular, the following topics apply to Oracle Application Server backup and recovery:

   - Using `ARCHIVELOG` mode
   - Performing cold database backups
   - Performing online database backups
   - Using the RMAN backup and recovery utility

2. **Configure the OracleAS Backup and Recovery Tool.**

   Oracle recommends you configure the tool and familiarize yourself with its features.

3. **Implement the Backup Strategy.**

   Chapter 21, "Backup Strategy and Procedures" outlines the Oracle-recommended backup strategy and backup procedures. Following this backup strategy ensures that you will be able to perform the recovery procedures in this book.

4. **Recover as Necessary.**

   In the event of system failure or data loss, refer to Chapter 22, "Recovery Strategies and Procedures". It outlines different types of failures and describes the procedures you can follow to recover.

20

Oracle Application Server Backup and Recovery Tool

This chapter describes how to install, configure, and use the Oracle Application Server Backup and Recovery Tool.

It contains the following topics:

20.1 How to Obtain the Oracle Application Server Backup and Recovery Tool

The Oracle Application Server Backup and Recovery Tool is installed as part of an Oracle Application Server installation. The tool is located in the *Oracle_Home*/backup_restore directory. Table 20–1 lists the files that may reside in the backup_restore directory.

Table 20–1 OracleAS Backup and Recovery Tool Files

File[1]	Description
bkp_restore.pl	If you have installed TopLink or MRCA, run this Perl script.
bkp_restore.sh	A shell script used to run the Perl script on UNIX.
bkp_restore.bat	A batch command file used to run the Perl script on Windows.
config/config.inp	The main configuration file that contains parameters for customizing the tool for your environment. The orainst_loc_path field must be changed only if the instance is installed with the –invPtrLoc option installer command-line option. It must be changed to reflect the nonstandard location of oraInst.loc.

Table 20–1 (Cont.) OracleAS Backup and Recovery Tool Files

File[1]	Description
config/config_*component*_files.inp	Component configuration files—each contains a list of configuration files for a particular component. These specify which files to back up when performing a configuration file backup. See Section 19.2.2, "Oracle Application Server Component Backup Input Files" for a list component configuration files.

[1] Paths are relative to the root of the OracleAS Backup and Recovery Tool directory.

See Also: *"Oracle Application Server Installation Guide"* for information about installing the Oracle Application Server.

20.1.1 Manually Installing the OracleAS Backup and Recovery Tool

If you are running TopLink, in standalone mode, or RepCA, in an existing database, you must install the OracleAS Backup and Recovery Tool manually. Before you install the OracleAS Backup and Recovery Tool, review the following notes:

- You must install the tool on the same host as its corresponding installation. You can install the tool in the Oracle home of its corresponding installation, or you can install it into a directory outside of the Oracle home.
- The tool requires a Perl 5.6.1 interpreter, or later. You can obtain the interpreter from the Perl site: http://www.perl.org, or you can use the Perl interpreter that ships with Oracle Application Server:
  - On UNIX systems:

    ORACLE_HOME/perl/bin/perl

  - On Windows systems:

    ORACLE_HOME\perl\5.6.1\bin\MSWin32-x86\perl.exe

- The tool requires that J2SE Development Kit (JDK) be in the execution path. You can obtain the JDK at: http://java.sun.com/j2se.

To install the OracleAS Backup and Recovery Tool:

1. Log in as the user who installed Oracle Application Server.
2. Extract the backup_restore.jar from the MRUA + Utilities cd which is located in the directory: CD_ROM/utilities/backup/backup.jar, for example:

   ```
   cd ORACLE_HOME
   jar xvf
   CD_ROM/utilities/backup/backup.jar
   ```

   If you install the Oracle Application Server Metadata Repository Upgrade Assistant, then the file backup_restore.jar is automatically extracted for you and put in directory *ORACLE_HOME*/utilities/backup

   Once you have obtained the backup_restore.jar, extract its contents into the Oracle home of the Toplink or RepCA installation. For example:

   ```
   cd ORACLE_HOME
   jar xvf utilities/backup/backup_restore.jar
   ```

3. On UNIX, make sure the `bkp_restore.sh` file has execute permission, for example:

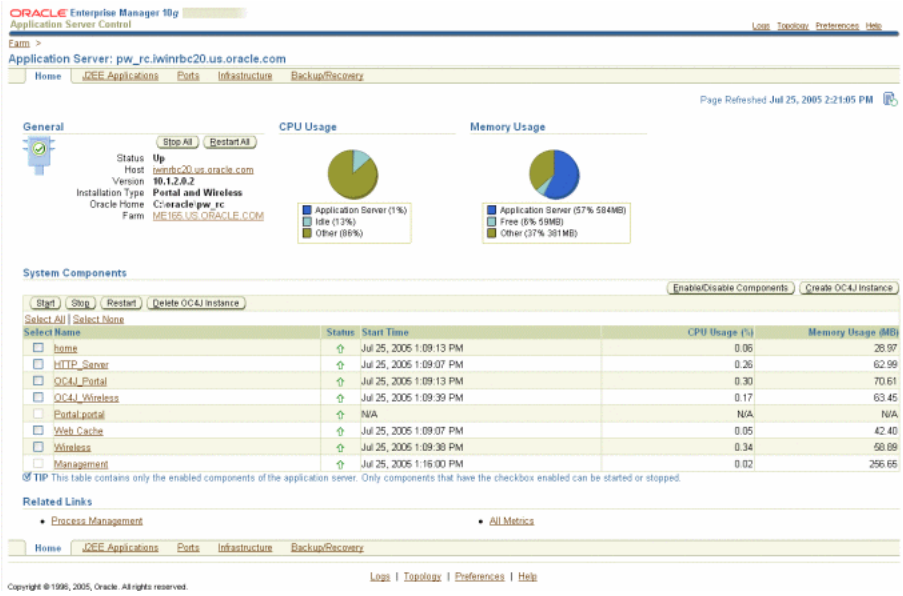
```
chmod 755 ORACLE_HOME/backup_restore/bkp_restore.sh
```

4. Familiarize yourself with the OracleAS Backup and Recovery Tool files, which are described in the Table 20–1. Instructions for editing the configuration files are in subsequent steps.

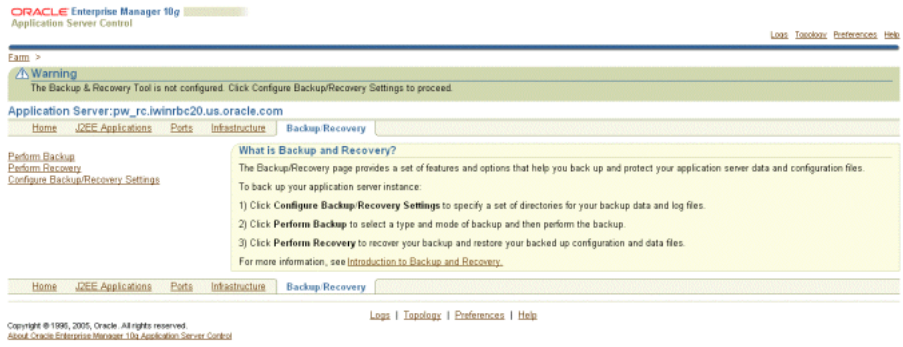
20.2 Using Oracle Application Server Control to Configure the Backup and Recovery Tool

You can use Oracle Application Server Control to configure the Backup and Recovery Tool by performing the following steps:

1. Login into **Oracle Application Server Control**. The Application Server Control Console displays:



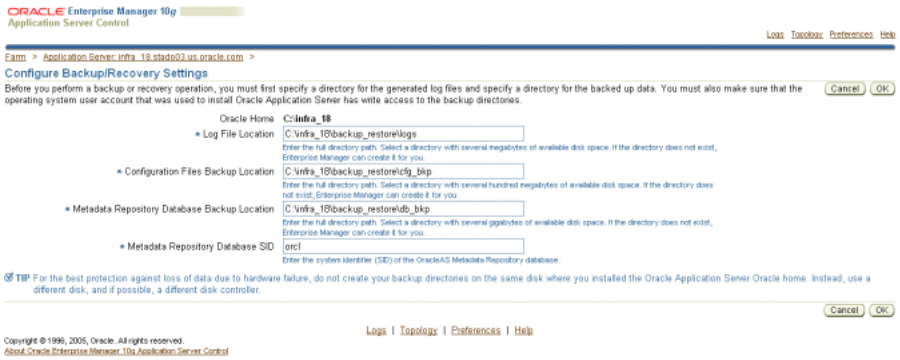
2. Click **BackupRecovery**. The Backup and Recovery screen displays:



3. If the Backup and Recovery Tool has not been configured, a warning screen displays stating that the tool is not configured. Click **Configure BackupRecovery Settings**. Depending on the type of installation, the midtier configuration screen or the Infrastructure configuration screen displays:

For the midtier configuration screen, enter the following information for each field:

- **Log File Location**—Enter the directory where you want the log file for backup and recovery stored. You should allow for several megabytes of disk space. In the preceding example screen the default location is shown.
- **Configuration Files Backup Location**—Enter the directory where you want the backups of configuration files stored. You should allow for several hundred megabytes of disk space. In the preceding example screen, a location is filled in as an example. If the Backup and Recovery Tool is not configured, this field is blank.

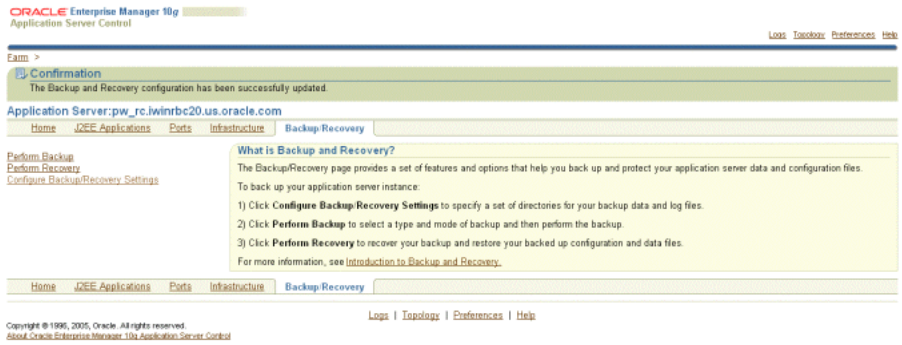


For the Infrastructure configuration screen, enter the following information for each field:

- **Log File Location**—Enter the directory where you want the log file for backup and recovery stored. You should allow for several megabytes of disk space.
- **Configuration Files Backup Location**—Enter the directory where you want the backups of configuration files stored. You should allow for several hundred megabytes of disk space.
- **Metadata Repository Database Backup Location**—Enter the directory where you want the backups of the metadata repository database stored. You should allow for several hundred gigabytes of disk space.
- **Metadata Repository Database SID**—This field is automatically filled in. Change it only in the case where the database SID has changed since installation.

4. Click **OK**. If any of the specified directories do not exist, a confirmation screen displays:

5. Click **Yes** to have the directories created, or click **No** to create the directories manually. Once a successful configuration completes, a confirmation screen displays a message stating that configuration was successful, or the screen displays a message stating that the configuration was unsuccessful.



20.3 How to Configure the OracleAS Backup and Recovery Tool Manually

This section describes how to configure the OracleAS Backup and Recovery Tool manually. You must follow these steps for each installation in your environment.

> **Note for Windows Users:** Do not use a rich text editor, such as WordPad, when editing files in the Backup and Recovery Tool directory. It inserts a return character at the end of each line that may cause the tool to fail. Oracle recommends that you use a basic text formatter, such as Notepad, instead.

1. Prior to running the Backup and Recovery Tool, set `ORACLE_HOME` for your environment. If the instance is an Infrastructure installation, set `ORACLE_SID` to the Metadata Repository SID.
2. If the installation is an Infrastructure or metadata repository, ensure that the database and the listener are up.
3. The tool writes out log files and backup files, and you must specify the following directories to hold these. The default log file directory is *`ORACLE_HOME`*`/backup_restore/logs`. Edit config.inp to create the following directories:
   a. **Log file directory**: (Middle tier and Infrastructure) This directory holds log files created by the tool. This directory should have several megabytes of space.
   b. **Configuration file backup directory**: (Middle tier and Infrastructure) This directory holds configuration file backups. This directory should have several hundred megabytes of space.

c. **Database backup directory**: (Infrastructure only) This directory holds datafile and control files backups of the Metadata Repository, as well as archived redo logs. This directory should have several gigabytes of space.

Recommendations for creating these directories are as follows:

- Create your backup directories on a file system on a separate disk and, if possible, a separate disk controller, than your Oracle Application Server Oracle home. This gives you the best chance of recovering data in the event of a hardware failure.
- Make sure your backup directories are writable by the user that installed Oracle Application Server.

  For example, to create a log file directory, configuration file backup directory, and database backup directory on `/disk1`:

  On Unix:

```
mkdir -p /disk1/backups/log_files
mkdir -p /disk1/backups/config_files
mkdir -p /disk1/backups/db_files
cd /disk1/backups
chmod 755 log_files config_files db_files
chown OracleAS_user log_files config_files db_files
```

  On Windows:

```
mkdir C:\backups\log_files
mkdir C:\backups\db_files
mkdir C:\backups\config_files
```

4. Edit `config.inp` and modify the parameters as described in Table 20–2. Notice that some of the instructions are different depending on whether this is a middle-tier or Infrastructure installation.

Table 20–2 Parameters in config.inp

Parameter	Value
`oracle_home`	Do not insert a value for this. If you invoke the Backup and Recovery Tool through Oracle Application Server Control, it will pass the oracle_home value corresponding to the instance. If you are using the command-line interface, set ORACLE_HOME in the shell environment first.
`log_path`	Specify the full path of the log file directory. If the full path is not specified, the default log directory *ORACLE_HOME*`/backup_restore/logs` is automatically created when the `-m configure` command is executed. If a `log_path` is specified in the `config.inp` file, but the specified directory does not exist, the Backup and Recovery Tool automatically creates the specified log directory whether or not the `-f` (force) option is used in the `-m configure` command. However, the configuration file backup directory and the database backup directory are not automatically created unless the `-f` option is specified.
`config_files_list`	Do not insert a value for this; leave it as `config_files_list=DO_NOT_SET`. This parameter will be updated with the appropriate list of configuration files for your installation when you run `bkp_restore.pl -m configure`.
`config_backup_path`	Specify the full path of the configuration file backup directory.

Table 20–2 (Cont.) Parameters in config.inp

Parameter	Value
install_type	Do not insert a value for this; leave it as install_type=DO_NOT_SET. This parameter is updated with the appropriate value for your installation when you run bkp_restore.pl -m configure.
dbid	Do not insert a value for this; leave it as dbid=DO_NOT_SET. For Infrastructure installations, this value is updated when you run bkp_restore.pl -m configure. By default, the tool obtains the dbid from the Metadata Repository. Or, you can supply a dbid in special cases involving migrating a Metadata Repository from one host to another, such as for Disaster Recovery. For middle-tier installations, this value is untouched.
pfile	**Middle-tier Installation:** Leave this line commented out. **Infrastructure:** If desired, specify an alternate pfile to use when starting up the database. Otherwise, leave the line commented out and the default spfile will be used: ■ For UNIX systems: *ORACLE_HOME*/dbs/spfileSID.ora ■ For Windows systems *ORACLE_HOME*\database\spfileSID.ora Be sure to leave the pfile entry commented out if you want to use the default because blank values are not allowed in this file. If the spfileorcl.ora file is not present at the default location, the following file will be used as pfile: For UNIX: *ORACLE_HOME*/dbs/initSID.ora For Windows: *ORACLE_HOME*\database\initSID.ora If you want to use a different pfile, specify an alternate pfile name for starting up the database.
database_backup_path	**Middle-tier Installation:** Do not insert a value for this; leave it as database_backup_path=VALUE_NOT_SET. **Infrastructure:** Specify the full path of the database backup directory.
orainst_loc_path	This parameter is used for UNIX platforms only. If the default path is overridden during installation, specify the full path of the directory where the oraInst.loc file exits. Otherwise, leave the parameter with the default value.
infra_with_portal	Do not insert a value for this; leave it as infra_with_portal=VALUE_NOT_SET. This parameter indicates whether the instance is an Infrastructure installation with registered portal middle-tiers.

Configure the tool by running it with the -m configure option, for example:

- For UNIX systems:

```
./bkp_restore.sh -m configure
```

- For Windows systems:

```
bkp_restore.bat -m configure
```

- For TopLink or MRCA installations on UNIX or Windows, after specifying the correct version of perl.exe, run:

```
bkp_restore.pl -m configure
```

This updates parameters in `config.inp` and, in the case of an Infrastructure, creates customized `.dat` files, which are used to backup, restore, and recover the Metadata Repository.

You are now ready to use the OracleAS Backup and Recovery Tool.

20.4 Running the Portal Validation/Cleanup Utility

If your Oracle Application Server installation has a Metadata Repository Database (MRDB) with a registered Portal middle-tier, by default, the Portal Schema Validation/Cleanup Utility (SVU) runs during a backup of the MRDB. Running SVU provides a way to ensure the integrity of your data whenever you perform a backup of your database. If you do not want to run SVU, uncheck the Validate Portal Schema checkbox in the Oracle Application Server Control Backup screen. If you are using the command line, you can disable SVU by using the `-z` option:

For UNIX systems:

```
bkp_restore.sh -z -m backup_instance_cold
```

For Windows systems:

```
bkp_restore.bat -z -m backup_instance_cold
```

If you want to run SVU and your installation has Identity Management running in a different instance than the instance where the MRDB is installed, ensure that Oracle Internet Directory is running in the instance where Identity Management runs before performing a backup of the MRDB.

The Backup and Recovery tool runs SVU in "reporting" mode to flag data inconsistencies after performing a backup. Output from the SVU is saved in a file in the same directory as the backup log: *`log_path`*`/YYYY-MM-DD_HH-MM-SS_portal_validation.log`. The *`log_path`* is the value of the `log_path` parameter in the `config.inp` file. You can clean up the inconsistencies by running the SVU in "cleanup" mode. After that, you can delete the SVU output files (`*_portal_validation.log`) to conserve space.

When you run the Backup and Recovery tool in configure or backup mode and you are not disabling SVU, it checks to see if the Oracle Application Server instance has a MRDB with a registered Portal middle-tier. If it does and the infra_with_portal parameter is not set to **yes**, the tool changes the value to **yes**.

With the `portal_validation` parameter set to **yes**, the Backup and Recovery tool runs SVU after the database backup is taken as a result of running the tool in one of the following modes:

- `backup_cold`
- `backup_cold_incr`
- `backup_online`

- `backup_online_incr`
- `backup_instance_cold`
- `backup_instance_online`
- `backup_instance_cold_incr -l incr_backup_level`
- `backup_instance_online_incr -l incr_backup_level`

20.5 Customizing the Tool for Your Configuration Files

As shipped, the OracleAS Backup and Recovery Tool backs up all of the Oracle Application Server configuration files that are necessary to reconstruct an Oracle Application Server installation. You can customize the tool to include any additional files that you would like to back up regularly, or to exclude any configuration files you do not want to back up.

20.5.1 How the Tool Works When Backing Up Configuration Files

Before you customize the tool, you should understand how it works. When you use the tool to back up your configuration files, it:

1. Opens `config.inp` (unless another environment file was specified with the `-e` option) and retrieves `config_files_list`.
2. Attempts to open each file in `config_files_list` and exits with an error if it cannot open all of the files.
3. Examines the contents of `config_exclude_files.inp`. The tool will not attempt to back up the files listed in this file.
4. Walks through each file in `config_files_list` and examines the first entry in each file. This entry is the *key file*. The key file is used to determine if the component exists in this installation.
   - If the tool finds the key file, it knows the component is installed, and attempts to back up all of the entries in the file. It logs an error whenever it cannot find a key file. For all other files that the tool does not find, a warning is issued and the backup continues.
   - If the key file does not exist, the tool does not attempt to back up any entries in the configuration file. It logs an error to the log file and skips to the next configuration file.
5. The configuration files are stored in jar files located in the directory specified by the `config_backup_path` parameter in the config.inp file. Two jar files are created, one for DCM-managed components and one for all the other components. The jar files are paired by the timestamp incorporated in each jar file name, for example:

   ```
   config_bkp_2004-05-10_18-33-10.jar
   dcm_archive_2004-05-10_18-33-10.jar
   ```

20.5.2 How to Customize the Tool

Since the tool knows how to determine which configuration files exist in your installation, it is not necessary to customize the tool. However, you may want to customize the tool by:

- Adding Files to a Backup

You may want to add your own local configuration files or any other files you would like to back up regularly, such as log files.

- Excluding Files from a Backup

  You may want to exclude files from being backed up.

Adding Files to a Backup

To add files, such as Oracle Application Server component specific log files, to a backup, add entries to the `config_misc_files.inp` file as follows:

- To specify a particular file:

  `${OH}/directorypath/file`

- To specify an entire directory:

  `${OH}/directorypath/`

- To use wildcards:

  `${OH}/directorypath/*.html`

You can add as many entries as you like. The `config_misc_files.inp` file is always included in the `config_files_list` parameter in `config.inp`, so there is no need to edit `config.inp`.

In some cases Oracle Backup and Recovery Tool might not be aware of additional configuration or content files stored outside a typical directory structure. For example, in following cases you must edit `config_misc_files.inp` to ensure proper backup of the following configuration files:

- Virtual paths defined in the Oracle HTTP configuration file -- httpd.conf. The web server configuration is pointing to a set of static files located in specific directory. These files should be considered a part of the runtime and metadata information.
- An application deployed to a OC4J container that uses files located outside the container directory. The Backup and Recovery Tool automatically backs up all the files located in the container directory. If your application uses any additional directories, you should consider them as part of configuration backups.
- Java Messaging Service (JMS) with the file-based persistence. The JMS runtime data (messages) are stored in physical files and should be a part of the backup process.

Note that you do not need to specify a key file in `config_misc_files.inp`.

Excluding Files from a Backup

You can exclude files from a backup in either of the following ways:

- You can simply remove the file entry from its `config_component.inp` file.
- If you have a situation where a `config_component.inp` file specifies an entire directory to back up, and you would like to exclude a specific file from that directory, you can add an entry for that file to `config_exclude_files.inp`. The tool will back up the entire directory except for the file you specify. You cannot specify directories or use wildcards in `config_exclude_files.inp`. Only single file entries are allowed.

  Note that you do not need to specify a key file in `config_exclude_files.inp`.

20.6 OracleAS Backup and Recovery Tool Usage Summary

This section summarizes usage for the OracleAS Backup and Recovery Tool.

It contains the following topics:

20.6.1 Prerequisites for Running the Tool

Before running the OracleAS Backup and Recovery Tool:

- Log in as the user that installed Oracle Application Server.
- Make sure the `ORACLE_HOME` environment variable is set.
- If you are performing a database backup, make sure the `ORACLE_SID` environment variable is set.
- All remote database instances must be shutdown before performing any of the following operations on a RAC database:
  - backup_cold
  - backup_cold_incr
  - backup_instance_cold
  - backup_instance_cold_incr
  - restore_repos
  - restore_instance

20.6.2 Syntax

The following commands and syntax for the commands is provided for instances of MRCA, TopLink, and standalone J2EE. While they can also be used with other components, Oracle highly recommends that you use Oracle Application Server Control to manage and run the Backup and Recovery Tool.

The syntax for the Oracle Application Server Backup and Recovery Tool is:

On UNIX:

```
bkp_restore.sh [-defsv] -m mode [args]
```

On Windows:

```
bkp_restore.bat [-defsv] -m mode [args]
```

It accepts the following options:

`-c` Restore control file as part of the database restore
`-d` Print a trace without executing.
`-e` Specify an environment file (default is `config.inp`).
`-f` Force log file, database backup, and configuration file directories to be created if they are required by the current command and do not exist.
`-n` Suppress prompts so the tool can run in batch mode.
`-o` Loss of Host Automation (LOHA) operation

`-s` Run in silent mode.
`-v` Run in verbose mode.
`-z` Suppress Portal Validation This option applies to Infrastructure installations only.

Use the -m option to specify which mode to run. Some modes take arguments. Table 20–3 describes the OracleAS Backup and Recovery Tool modes and their arguments. All modes and arguments are case-sensitive.

Some of the modes in the following table are included for use with OracleAS TopLink, OracleAS Metadata Repository Creation Assistant (MRCA), and custom database installations. The modes are:

- `backup_cold`
- `backup_cold_incr -1 incr_backup level`
- `backup_online`
- `backup_online_incr -1 incr_backup level`
- `restore_repos`
- `flashback_repos`

Table 20–3 Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description

`backup_cold`	Performs a complete cold backup of the Metadata Repository. The command performs the following operations:
	■ Opens `config.inp` (or the alternate file specified with the `-e` option) and retrieves `log_path`.
	■ Shuts down the database, starts it in mounted mode, but does not open it.
	■ Performs a backup of the datafiles and control files using RMAN. The commands are in `backup_cold.dat`.
	■ Stores the backup in the directory specified in `backup_cold.dat`. (This is usually set to the `database_backup_path` in `config.inp`.)
	■ Stores a log file in `log_path`.
	■ Opens the database.
	For a DCM file-based Metadata Repository:
	■ Executes the `dcmctl exportrepository` command to perform a backup of the file-based repository.
	■ Stores the backup in the directory, specified by `config_backup_path` parameter in `config.inp`.
	If both a metadata repository and a file-based repository coexist in an application server instance, the `backup_cold` option backs up both of them as a set. This would be the case where a file-based repository exists in an Infrastructure install.
	To check whether a particular OracleAS instance hosts a file-based repository or a database repository, use the following command:
	ORACLE_HOME`/dcm/bin/dcmctl whichfarm`
	`Repository Type: Database (host) => Hosts a database repository`
	`Repository Type: Distributed File Based (host) => Hosts a file based repository`

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description
`backup_cold_incr`	Performs an incremental backup of the Metadata Repository.
`-l` *`incr_backup_level`*	Works the same as `backup_cold`, except: ■ The `-l` option specifies the increment level (`0` - `4`). ■ Uses the `backup_cold_incr`*`level`*`.dat file` There are two types of incremental backups, cumulative and differential. The tool uses the default type, which is differential. For more information, refer to *Oracle Database Backup and Recovery Basics* in the Oracle Database 10*g* Documentation Library.

`backup_config`	Performs a full configuration backup. The backup includes the configurations for DCM managed components and non-DCM managed components. The command performs the following operations:
	■ Opens `config.inp` (or the alternate file specified with the `-e` option) and retrieves `config_files_list`, `config_backup_path`, and `log_path`.
	■ Creates an archive for configuration of DCM managed components:
	dcmctl createarchive -archive *archive_name*
	dcmctl exportarchive -archive *archive_name* -f *unique name in config_backup_path*
	dcmctl removearchive -archive *archive_name*
	■ Attempts to open each file in `config_files_list`. Exits with an error if it cannot open all of the files.
	■ For each file in `config_files_list`, checks if the first entry (the key file) exists. If the key file does not exist, it is treated as a fatal error. Otherwise, backs up all files in the list. If any other files do not exist, logs an error and continues.
	■ Excludes files listed in `config_exclude_files.inp`.
	■ When finished, stores the backup in `config_backup_path/config_bkp_`*`timestamp`*`.jar` and `config_backup_path/dcm_archive_`*`timestamp`*`.jar` for DCM-managed components.
	■ If any errors are encountered, creates a log file in *`log_path`*`/config_bkp_`*`timestamp`*.
	Process Prerequisites:
	If the DCM repository type is a database, the following processes should be up:
	■ The Oracle Internet Directory process must to be up. The command `opmnctl startproc ias-component=OID` can be used to start this process. The Oracle Internet Directory process exists on Infrastructure (IM + MR) or IM installation. Before starting the Oracle Internet Directory process, the OPMN process must be up. The command `opmnctl start` can be used to bring it up.
	■ The database must be up and running.
	■ The listener process must be up.
	To check whether a particular Oracle Application Server instance hosts a file based repository or a database repository, use the following command:
	`ORACLE_HOME``/dcm/bin/dcmctl whichfarm`
	`Repository Type: Database (host) => Hosts a database repository`
	`Repository Type: Distributed File Based (host) => Hosts a file based repository`

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description
`backup_config_incr`	Performs an incremental configuration file backup. Works the same as `backup_config`, except: ■ Backs up all configuration files that have changed since the last full or incremental configuration file backup. For process prerequisites, refer to the `backup_config` option.
`backup_instance_cold`	Performs a complete cold backup of the Oracle Application Server instance. The command performs the following operations: ■ Stops all OPMN managed processes. ■ Starts the OPMN administrative process. ■ Checks all of the OPMN managed processes to ensure that the processes are stopped. If not, tries to stop them one more time. If the processes still cannot be stopped, issues a fatal error. ■ Performs repository backup (database and file-based). For a database repository, shuts down the database for the duration of the backup. ■ Starts Oracle Internet Directory and DCM-daemon processes for database repositories. ■ Performs configuration backup. ■ Starts all OPMN managed processes. ■ Checks to ensure that all OPMN processes are running. If not, issues a warning message.
`backup_instance_cold_incr -1` *`level number`*	Performs an incremental cold backup of the Oracle Application Server instance. The command performs the following operations: ■ Stops all OPMN managed processes. ■ Starts the OPMN administrative process. ■ Checks all of the OPMN managed processes to ensure that the processes are stopped. If not, tries to stop them one more time. If the processes still cannot be stopped, issues a fatal error. ■ Performs repository backup (database or file-based). For a database repository, shuts down the database for the duration of the backup. The level option applies to database repositories only. Backup is performed to the specified level. The default level is 1. ■ Starts Oracle Internet Directory and DCM-daemon processes for database repositories. ■ Performs configuration backup. ■ Starts all OPMN managed processes. ■ Checks to ensure that all OPMN processes are running. If not, issues a warning message.
`backup_instance_online`	Performs an online backup of the Oracle Application Server instance. The Metadata Repository database must have ARCHIVELOG mode enabled. The command performs the following operations: ■ Performs repository backup (database or file based). For a database repository, the database remains up while being backed up. ■ Performs configuration backup.

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description
`backup_instance_online_incr -1` *`level number`*	Performs an incremental online backup of the Oracle Application Server instance. The command performs the following operations: ■ Performs an incremental repository backup (database or file-based). For a database repository, the database remains up while being backed up. The level option applies to database repositories only. Backup is performed to the specified level. The default level is 1. ■ Performs incremental configuration backup.

`backup_online`	Performs an online backup of the Metadata Repository. If you are running this command on an Infrastructure, ensure that the Metadata Repository is up before running this command. The Metadata Repository database must have ARCHIVELOG mode enabled. The command performs the following operations:
	■ Opens `config.inp` (or the alternate file specified with the `-e` option) and retrieves `log_path`.
	■ Assumes the database is open.
	■ Performs a backup of the datafiles and control files using RMAN. The commands are in `backup_online.dat`.
	■ Stores the backup in the directory specified in `backup_online.dat`. (This is usually set to the `database_backup_path` in `config.inp`.)
	■ Stores a log file in `log_path`.
	■ Leaves the database open.
	For a DCM file-based Metadata Repository:
	■ Executes the `dcmctl exportrepository` command to perform a backup of the file-based repository.
	■ Stores the backup in the directory, specified by `config_backup_path` parameter in the `config.inp` file.
	If both a metadata repository and a file-based repository coexist in an application server instance, the `backup_online` option backs both of them up as a set. This would be the case where a file-based repository exists in an infrastructure install.
	To check whether a particular OracleAS instance hosts a file-based repository or a database repository, use the following command:
	`ORACLE_HOME``/dcm/bin/dcmctl whichfarm`
	`Repository Type: Database (host) => Hosts a database repository`
	`Repository Type: Distributed File Based (host) => Hosts a file based repository`

`backup_online_incr` `-l` *`incr_backup_level`*	Performs an incremental online backup of the Metadata Repository. Works the same as `backup_online`, except: ■ The `-l` option specifies the increment level (`0 - 4`). ■ Uses the `backup_online_incr`*`level`*`.dat file` There are two types of incremental backups, cumulative and differential. The tool uses the default type, which is differential. For more information, refer to *Oracle Database Backup and Recovery Basics* in the Oracle Database 10*g* Documentation Library.

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description
`configure` `[-i dbid]`	Configures the tool. When using this command on an Infrastructure, make sure the Metadata Repository is up before you run this command. The command performs the following operations: ■ Updates `config_files_list` and `install_type` in `config.inp` with the appropriate information for your installation. ■ If using this on an Infrastructure, updates the configuration file with the database id (`dbid`) and creates customized `*.dat` files from the database backup `*.tmpl` files. By default, it queries the Metadata Repository for the `dbid`. If you use the `-i` option, you can supply the `dbid` (this is used for migrating the Metadata Repository from one node to another, such as for Disaster Recovery).
`configure_nodb`	Same as `configure` but does not perform the Infrastructure configuration. Note: You should use `configure` for all middle-tier and Infrastructure installations. The `configure_nodb` applies to disaster recovery strategies described in *Oracle Application Server High Availability Guide*.
`help`	Prints a usage message.
`list_backups`	Lists the metadata repository and configuration backups taken for the instance.
`list_instance_backups`	Lists instance level backups taken for the instance.
`list_changed_config`	Lists any configuration files that have changed since the last full or incremental backup. This command checks the modification date of each file; it does not check the actual contents of the file. It writes the list of files to a log file and prints the name of the log file. Deleted files or deleted directories are not listed in `list_changed_config`. Only modified files or directories containing modified files are listed.
`node_backup -o image_backup -P` *`directory for the image archive`*	Creates an image archive of the original host. The image includes the original Oracle home, oratab, central inventory and so forth depending on the installation. On UNIX, this operation must be run as root.
`node_backup -o prepare`	Prepares the node for backup. Preparation includes discovering the operating system type, host name/ip, user/group id, install type, the location of the central inventory, oracle home locations if there are multiple of them, Windows registry, Windows service database scanning to find all services created for Oracle homes. The information is placed in a file to be used in node restoration. This mode also creates a config backup and a cold database backup.
`node_restore -o inst_reconfigure -t config_`*`bkp_timestamp`*	Reconfigures the instance on the new host including ip changing, database restore, database tempfile setup, config backup restore and so forth depending in the installation type.
`node_restore -o inst_register`	Registers the instance with the oratab and the central inventory. It also sets up the daemon start and stop script and so forth by running root.sh, or on Windows, Windows services are created. It must be run as root on UNIX systems.
`node_restore -o sys_init`	Restores Oracle Universal Installer related metadata such as oratab(Unix), Windows registries (Windows) and central inventor. It should be run once only on the new host. It must be run as root on UNIX systems.

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description

`restore_config`
`[-t config_bkp_`*`timestamp`*`]`
`[-n]`
	Restores configuration files. The command performs the following operations:
	■ Opens `config.inp` (or the alternate file specified with the `-e` option) and retrieves `config_backup_path` and `log_path`.
	■ If the `-t` option is supplied and it is the timestamp from a full backup, it restores that full backup.
	■ If the `-t` option is supplied and it is the timestamp from an incremental backup, it restores the full backup and all incremental backups up to and including the specified incremental backup.
	■ If the `-t` option is not supplied, displays a list of configuration file backups in `config_backup_path` and exits. You can then rerun the command and supply one of these files with the `-t` option.
	■ Restores all files from the configuration file backup to the Oracle home, preserving owner, group, permissions, and timestamp.
	■ If any errors are encountered, creates a log file in *`log_path`*`/config_rst_`*`timestamp`*.
	■ Restore configuration for DCM managed components
	`dcmctl importarchive -f` *`location in config_backup_path that contains dcm archive`*
	`dcmctl applyarchiveto -archive` *`archive name`* `[-cluster` *`cluster_name`*`]`
	`dcmctl removearchive -archive` *`archive name`*
	The `-n` option suppresses prompts so you can use the tool in batch mode.
	For the process prerequisites, refer to the `backup_config` option.
	Do not run `restore_config` on multiple nodes in a J2EE cluster in parallel. Doing so will cause `restore_config` failures. Run `restore_config` on one node at a time.

`restore_db`	This command is deprecated. Use `restore_repos` instead.
`restore_instance -t` *`timestamp`*`-c`	Restores an instance of Oracle Application Server. If the timestamp argument is not specified, then a list of backup timestamps is displayed to the user. The command performs the following operations: ■ Stops all OPMN managed processes. ■ Checks to verify that the OPMN processes have stopped. If OPMN processes cannot be stopped (maybe an `opmn.xml` file is missing), a file system restore is performed. Then tries to stop the OPMN processes again. If the OPMN processes still cannot be stopped, issues a fatal error. ■ Starts the OPMN administration process. ■ Performs repository restore. The `-c` option is applicable for database repositories only. If the `-c` option is specified, the control file is restored also. ■ Starts Oracle Internet Directory and DCM-Daemon processes (applicable to database repositories only). ■ Performs configuration restore. ■ Starts all OPMN managed processes. ■ Checks to ensure that all OPMN managed processes are up. If not, issues a warning message.

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description

`restore_repos`
`[-u` *`timestamp`*`]`
`[-c] [-n]`
	Restores and recovers the Metadata Repository and the DCM file-based repository from the available cold and online backups. To perform `restore_repos`, the Metadata Repository database must be started and open. The command performs the following operations:
	■ Opens `config.inp` (or the alternate file specified with the `-e` option) and retrieves `log_path`.
	■ Restores the control files and datafiles, and performs recovery using RMAN. The commands are in `restore_repos.dat`.
	■ Stores a log file in `log_path`.
	■ Leaves the database open.
	By default, this command restores and recovers the database to its most recent state. You can use the `-u` option to restore and recover the database to its state at a particular point in time. The format for the timestamp is MM/DD/YYYY_HR24:MIN:SEC, for example:
	On UNIX:
	`bkp_restore.sh -m restore_repos -u 07/26/2003_13:45:06`
	On Windows:
	`bkp_restore.bat -m restore_repos -u 07/26/2003_13:45:06`
	By default, this command does not restore the control file. You can use the `-c` option to restore the control file.
	If you use the `-u` or `-c` option, be sure to do a full backup right away because all past backups are invalidated.
	The `-n` option suppresses prompts so you can use the tool in batch mode.
	Refer to Section 22.2.5, "Restoring and Recovering the Metadata Repository" for more information.
	This command performs the following operations to restore a file-based repository:
	■ Checks for timestamp input. If not provided, lists the available backup timestamps corresponding to the file-based repository.
	■ Executes `dcmctl importrepository -file` *`location in config_backup_path that stores the repository backup`*
	On UNIX:
	`bkp_restore.sh -m restore_repos -t 2004-05-10_18-33-12`
	On Windows:
	`bkp_restore.bat -m restore_repos -t 2004-05-10_18-33-12`
	If both the metadata repository and a file-based repository coexist in an application server instance, the `restore_repos` option restores both of them as a set. This would be the case where DCM uses a file-based repository in an infrastructure install.

Table 20–3 (Cont.) Oracle Application Server Backup and Recovery Tool Modes and Arguments

Mode and Arguments	Description

`flashback_repos`
`-u timestamp |`
`-b timestamp`
`[-n]`
	Rewinds the Metadata Repository to a specified time by using the before images of changed data blocks to back out changes made to the database since the specified time. To perform Flashback, the Metadata Repository database must be started and open. The command performs the following operations:
	■ Opens the `config.inp` file (or an alternate file specified with the `-e` option) and retrieves `log_path`.
	■ Recovers the database to or before a specified time by performing Flashback. The commands are located in:
	`flashback_repos_to.tmpl`
	`flashback_repos_before.tmpl`
	■ Stores a log file in `log_path`.
	■ Leaves the database open.
	Flashback requires a database repository. Flashback is not supported on file-based repositories.
	Flashback supports recovery of a Metadata Repository back to the point in time where resetlogs occur. Once resetlogs occur, Flashback cannot recover any change blocks that occurred before the resetlogs.
	You do not need to perform a cold backup before running Flashback. Flashback does not require restoring previous backups in order to recover the database. This means the `flashback_repos` operation is faster than the `restore_repos` operation. Flashback can undo any logical data corruption or user error, such as deleting an Oracle Application Server schema or undeploying an application by mistake.
	To perform Flashback, the database must be configured with a Flash Recovery Area, and ARCHIVELOG mode and Flashback must be enabled. Use the following SQL statements to configure and enable Flashback:
	`ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE = size SCOPE=BOTH SID='*';`
	`ALTER SYSTEM SET DB_RECOVERY_FILE_DEST = directory_path SCOPE=BOTH SID='*';`
	`ALTER DATABASE ARCHIVELOG;`
	`ALTER DATABASE FLASHBACK ON;`
	Refer to the Oracle Database Backup and Recovery Basics manual, Chapter 3, the section on "Setting up a Flash Recovery Area for RMAN" for more detail. Also, refer to Section 21.2.2, "Enabling ARCHIVELOG Mode" in this manual for information on enabling ARCHIVELOG mode.
	Either the -u or -b option must be specified. The -u option returns the database to its state at the specified time. The -b option returns the database to its state prior to the specified time. The format for the timestamp is `MM/DD/YYYY_HR24:MIN:SEC`.
	■ On UNIX systems:
	`bkp_restore.sh -m flashback_repos -u 07/26/2003_13:45:06`
	■ On Windows systems:
	`bkp_restore.bat -m flashback_repos -u 07/26/2003_13:45:06`
	The -n option suppresses prompts so the tool can be run in batch mode.
	For more information on Flashback technology, refer to the *Oracle Database Backup and Recovery Advanced User's Guide*.
	After running `flashback_repos`, do a full backup immediately because all past backups are invalidated. See Section 22.2.5, "Restoring and Recovering the Metadata Repository" for more information.

20.6.3 Usage Examples

This section contains usage examples for the OracleAS Backup and Recovery Tool. The Unix command is listed first and then the Windows command.

- Configure the tool using the default `config.inp` file:

```
bkp_restore.sh -m configure
bkp_restore.bat -m configure
```

- Configure the tool using a configuration file called `myconfig.inp`:

```
bkp_restore.sh -m configure -e myconfig.inp
bkp_restore.bat -m configure -e myconfig.inp
```

- Perform a full configuration file backup:

```
bkp_restore.sh -v -m backup_config
bkp_restore.bat -v -m backup_config
```

- Perform a full configuration file backup using an environment file called `myconfig.inp`:

```
bkp_restore.sh -v -m backup_config -e myconfig.inp
bkp_restore.bat -v -m backup_config -e myconfig.inp
```

- Perform an incremental configuration file backup:

```
bkp_restore.sh -v -m backup_config_incr
bkp_restore.bat -v -m backup_config_incr
```

- Restore configuration files.

```
bkp_restore.sh -m restore_config -t 2004-09-21_06-12-45
bkp_restore.bat -m restore_config -t 2004-09-21_06-12-45
```

- Perform a full cold backup of the Metadata Repository:

```
bkp_restore.sh -m backup_cold
bkp_restore.bat -m backup_cold
```

- Perform a level 2 incremental cold backup of the Metadata Repository:

```
bkp_restore.sh -m backup_cold_incr -l 2
bkp_restore.bat -m backup_cold_incr -l 2
```

- Perform an full online backup of the Metadata Repository:

```
bkp_restore.sh -m backup_online
bkp_restore.bat -m backup_online
```

- Perform a level 0 incremental online backup of the Metadata Repository:

```
bkp_restore.sh -m backup_online_incr -l 0
bkp_restore.bat -m backup_online_incr -l 0
```

- Restore the Metadata Repository to its most recent state:

```
bkp_restore.sh -m restore_repos
bkp_restore.bat -m restore_repos
```

- Restore the Metadata Repository to its state at a particular time:

```
bkp_restore.sh -m restore_repos -u 07/26/2003_13:45:06
```

```
bkp_restore.bat -m restore_repos -u 07/26/2003_13:45:06
```

- Flashback the Metadata Repository to its state at a particular point in time:

```
bkp_restore.sh -m flashback_repos -u 07/26/2003_13:45:06
bkp_restore.bat -m flashback_repos -u 07/26/2003_13:45:06
```

- Restores the file based repository to its state at a particular time:

```
bkp_restore.sh -m restore_repos -t  2004-05-10_18-33-12
bkp_restore.bat -m restore_repos -t  2004-05-10_18-33-12
```

- Perform an cold backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_cold
bkp_restore.bat -m backup_instance_cold
```

- Perform an incremental cold backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_cold_incr -l level
bkp_restore.bat -m backup_instance_cold_incr -l level
```

- Perform an online backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_online
bkp_restore.bat -m backup_instance_online
```

- Perform an online incremental backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_online_incr -l level
bkp_restore.bat -m backup_instance_online_incr -l level
```

- Restore an Oracle Application Server instance to its state at a particular time and include the control file in the restore:

```
bkp_restore.sh -m restore_instance -t 2004-09-21_06-12-45 -c
bkp_restore.bat -m restore_instance -t 2004-09-21_06-12-45 -c
```

- Node backup preparation using Loss of Host Automation (LOHA)

```
bkp_restore.sh -m node_backup -o prepare
bkp_restore.bat -m node_backup -o prepare
```

- Create an image backup of the original host using LOHA

```
bkp_restore.sh -m node_backup -o image_backup -P directory for image archive
bkp_restore.bat -m node_backup -o image_backup -P directory for image archive
```

- Restore OUI related metadata on the new host using LOHA

```
bkp_restore.sh -m node_restore -o sys_init
bkp_restore.bat -m node_restore -o sys_init
```

- Register the instance on the new host using LOHA

```
bkp_restore.sh -m node_restore -o inst_register
bkp_restore.bat -m node_restore -o inst_register
```

- Configure the instance on the new host using LOHA

```
bkp_restore.sh -m node_restore -o inst_reconfigure -t config_bkp_timestamp
bkp_restore.bat -m node_restore -o inst_reconfigure -t config_bkp_timestamp
```

20.6.4 Purging Backups and Moving Them to Tertiary Storage

The Backup and Restore Tool saves records of successful backups in a catalog file (`data/catalog.txt`) in the `backup_restore` directory. Each backup is identified by a timestamp, which is also embedded in the filenames of jar files saved in the configuration file backup directory in the case of a instance or configuration only backup. If you delete all the `.jar` files corresponding to a timestamp or move them somewhere else, for example offline storage, although the catalog still contains a record of the timestamp, you will not see this record when you run `-m list_backups`, nor will you be able to restore using this timestamp as the `-t` value. This is the expected behavior.

In the case of a repository only backup, a jar file is not created in the configuration backup directory. To delete obsolete database backups or move them to tape, you should use rman. When the backup files corresponding to a repository only backup are purged or moved to tertiary storage, the Backup and Restore Tool still lists the corresponding timestamp when you run -m list_backups although the database backup is not available for restore.

21

Backup Strategy and Procedures

This chapter describes the Oracle Application Server backup strategy and procedures.

It contains the following topics:

21.1 Recommended Backup Strategy

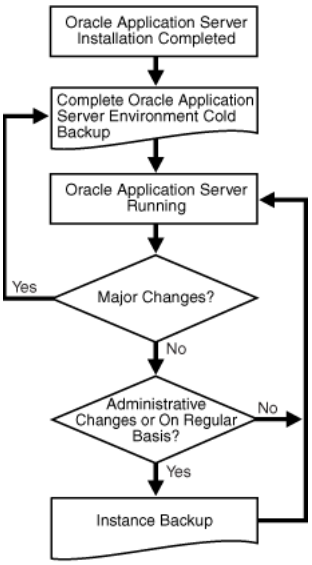
This section describes the recommended backup strategy for Oracle Application Server. Using this strategy ensures that you can perform the recovery procedures described in this book.

The backup strategy is as follows:

- Task 1: Perform a Complete Cold Backup of Your Oracle Application Server Environment
- Task 2: Perform Instance Backups on a Regular Basis
- Task 3: Perform a New Complete Environment Backup After a Major Change
- Task 4: Perform Instance Backups on a Regular Basis (Return to Task 2)

The flow chart in Figure 21–1 provides an overview of how to decide which type of backup is appropriate for a given circumstance.

Figure 21–1 Decision Flow Chart for Type of Backup



Task 1: Perform a Complete Cold Backup of Your Oracle Application Server Environment

The first backup you perform should be an image backup, which includes all of the files in your environment. Before you perform your first backup, make sure `ARCHIVELOG` mode is enabled in the Metadata Repository. You should also create a record of your environment.

1. Enable `ARCHIVELOG` mode in the Metadata Repository.

   By default, the Metadata Repository does not have `ARCHIVELOG` mode enabled. You should enable it immediately so your online redo logs are archived. You should enable `ARCHIVELOG` mode before you perform your first image backup. Otherwise, your backup control files will contain the `NOARCHIVELOG` mode setting. You cannot use the Backup and Recovery Tool in the `NOARCHIVELOG` mode.

   Refer to Section 21.2.2, "Enabling ARCHIVELOG Mode".

2. Perform a complete Oracle Application Server environment backup.

   This will serve as the baseline for all subsequent instance backups.

   Refer to Section 21.2.6, "Performing a Complete Oracle Application Server Environment Backup".

3. Create a record of your Oracle Application Server environment.

   In the event you need to reconstruct your environment, you can refer to this record.

   Refer to Section 21.2.3, "Creating a Record of Your Oracle Application Server Configuration".

Task 2: Perform Instance Backups on a Regular Basis

After every administrative change, or, if this is not possible, on a regular basis, perform an instance backup of your Oracle Application Server environment.

> **See Also:** Appendix G, "Examples of Administrative Changes" to learn more about administrative changes

Refer to Section 21.2.4, "Performing an Instance Backup of Oracle Application Server Using Application Server Control Console" or Section 21.2.5, "Performing an Oracle Application Server Instance Backup from the Command Line".

Task 3: Perform a New Complete Environment Backup After a Major Change

If you make a major change to your Oracle Application Server environment, you must perform a new image backup of your Oracle Application Server environment. This backup will serve as the basis for subsequent instance backups. You should also update the record of your environment with the new configuration information.

Perform a new image backup after:

- An operating system software upgrade
- An Oracle Application Server software upgrade or patch application

To do so:

1. Update the record of your Oracle Application Server environment.

   Refer to Section 21.2.3, "Creating a Record of Your Oracle Application Server Configuration".

2. Perform a complete Oracle Application Server environment backup.

   Refer to Section 21.2.6, "Performing a Complete Oracle Application Server Environment Backup".

Task 4: Perform Instance Backups on a Regular Basis (Return to Task 2)

After you establish a new complete Oracle Application Server environment backup, return to Task 2 and continue to perform instance backups on a regular basis.

Additional Tips:

- Create a backup of the JRE/JDK on your system. This is not an Oracle product, but it is utilized by Oracle Application Server and, if accidentally lost or corrupted, would need to be restored in order for Oracle Application Server to function. This issue only applies to HP-UX, HP Tru64, and IBM AIX systems.
- Ensure that your backups are valid by routinely verifying that they can be restored.

21.2 Backup Procedures

This section describes the backup procedures in detail. There is some data interdependency between the configuration files in your Oracle Application Server middle-tier installations, the Distributed Management Repository, the Identity Management metadata, and the Oracle Application Server Metadata Repository in the Infrastructure. In order to maintain configuration data consistency, you should take a backup of each of your Oracle Application Server instances (middle-tier and Infrastructure) at the same time. While taking a backup of one Oracle Application Server instance, ensure that no configuration changes are made in any of the other instances.

This section contains the following topics:

- Enabling Block Change Tracking
- Enabling ARCHIVELOG Mode
- Creating a Record of Your Oracle Application Server Configuration

- Performing an Instance Backup of Oracle Application Server Using Application Server Control Console
- Performing an Oracle Application Server Instance Backup from the Command Line
- Performing a Complete Oracle Application Server Environment Backup

21.2.1 Enabling Block Change Tracking

To increase performance on incremental database backups, enable block change tracking using the following command:

```
alter database enable block change tracking using file file_name;
```

If the db_create_file_dest parameter is set in the spfile or init.ora file of the database, the following command can be used:

```
alter database enable block change tracking;
```

Once you enable block change tracking, incremental database backup will use block change tracking.

For more information on block change tracking, refer to *"Backup and Recovery Basics"* in the *"Oracle Database 10g Release 1 (10.1) Documentation Library"*.

21.2.2 Enabling ARCHIVELOG Mode

By default, the Metadata Repository does not have ARCHIVELOG mode enabled. You must enable ARCHIVELOG mode, which enables the archiving of online redo logs. This will allow you to perform the recovery strategies in this book.

> **See Also:** You can find more detailed information on the parameters in this section, and setting up archive logging in general, in *"Oracle Database Administrator's Guide 10g Release 1 (10.1)"*.

To enable ARCHIVELOG mode:

1. Run the following sql query to check if the flashback_recovery_area is setup:

   ```
   SQL> show parameters db_recovery
   ```

   If the flashback_recovery_area is setup, the query returns:

   ```
   Name                           Type        Value
   db_recovery_file_dest          string      /private2/AS1012Installs/AS1012Infra/
                                               flash_recovery_area
   db_recovery_file_dest_size     big integer 2G
   ```

   If the flashback_recovery_area is setup, then the destination specified by the db_recovery_file_dest parameter is used as the archivelog destination, and you do not need to specify the destination directory for your archives in the following step.

2. Specify the destination directory for your archives by including the initialization parameter LOG_ARCHIVE_DEST_n in the initialization file. If spfile is used, then the following command can be issued:

   ```
   alter system set log_archive_dest_n="LOCATION=<backup directory>" scope=spfile;
   ```

In the log_archive_dest_n parameter, n is a number of 1 through 10.

If pfile is used, the following initialization file must be edited:

For UNIX systems:

INFRA_ORACLE_HOME/dbs/init*SID*.ora

For Windows systems:

INFRA_ORACLE_HOME\database\init*SID*.ora

Change the LOG_ARCHIVE_DEST_n parameter to:

LOG_ARCHIVE_DEST_n="LOCATION=<*backup directory*>"

(Optional) The default filename format for archive logs is:

- For UNIX systems:

  %t_%s_%r.dbf

- For Windows systems:

  ARC%S_%R.%T

If you would like to use a different format, include the initialization parameter LOG_ARCHIVE_FORMAT in the initialization file, for example:

```
LOG_ARCHIVE_FORMAT = 'log%t_%r_%s.arc'
```

In the preceding example, t represents the thread number, r represents the reset log ID, and s represents the log sequence number.

3. Make sure that the ORACLE_HOME and ORACLE_SID (the default is orcl) environment variables are properly set.
4. Make sure that no one is using the database.
5. Perform a clean, normal shutdown of the database instance.

```
INFRA_ORACLE_HOME/bin/sqlplus /nolog
SQL> connect sys/password as sysdba
SQL> shutdown
```

6. Start up the instance and mount, but do not open the database.

```
SQL> startup mount;
```

7. Enable database ARCHIVELOG mode.

```
SQL> alter database archivelog;
```

8. Shut down and restart the database instance.

```
SQL> shutdown
SQL> startup
```

9. Verify the database is now in ARCHIVELOG mode.

   Execute the following command and verify that Database log mode is Archive Mode and Automatic archival is Enabled.

```
SQL> archive log list;
Database log mode            Archive Mode
```

```
Automatic archival           Enabled
Archive destination          /disk1/oraHome/archive
Oldest on-line log sequence  997
Next log sequence to archive 999
Current log sequence         999
```

21.2.3 Creating a Record of Your Oracle Application Server Configuration

In the event you need to restore and recover your Oracle Application Server environment, it is important to have all the necessary information at your disposal. This is especially true in the event of a hardware loss that requires you to reconstruct all or part of your Oracle Application Server environment on a new disk or host.

You should maintain an up-to-date record of your Oracle Application Server environment that includes the information listed in this section. You should keep this information both in hardcopy and electronic form. The electronic form should be stored on a host or e-mail system that is completely separate from your Oracle Application Server environment.

Your Oracle Application Server hardware and software configuration record should include:

- The following information for each host in your environment:
  - Hostname
  - Virtual hostname (if any)
  - Domain name
  - IP address
  - Hardware platform
  - Operating system release level and patch information
- The following information for each Oracle Application Server installation in your environment:
  - Installation type (For example: Infrastructure or J2EE and Web Cache)
  - Host on which the installation resides
  - User name, userid number, group name, groupid number, environment profile, and type of shell for the operating system user that owns the Oracle home (`/etc/passwd` and `/etc/group` entries)
  - Directory structure, mount points, and full path for *`ORACLE_HOME`*
  - Amount of disk space used by the installation
  - Port numbers used by the installation

    Note: *`ORACLE_HOME`*`/install/portlist.ini` contains the port numbers assigned during installation. However, this file is not updated if you change port numbers after installation, so you need to keep track of those changes manually.

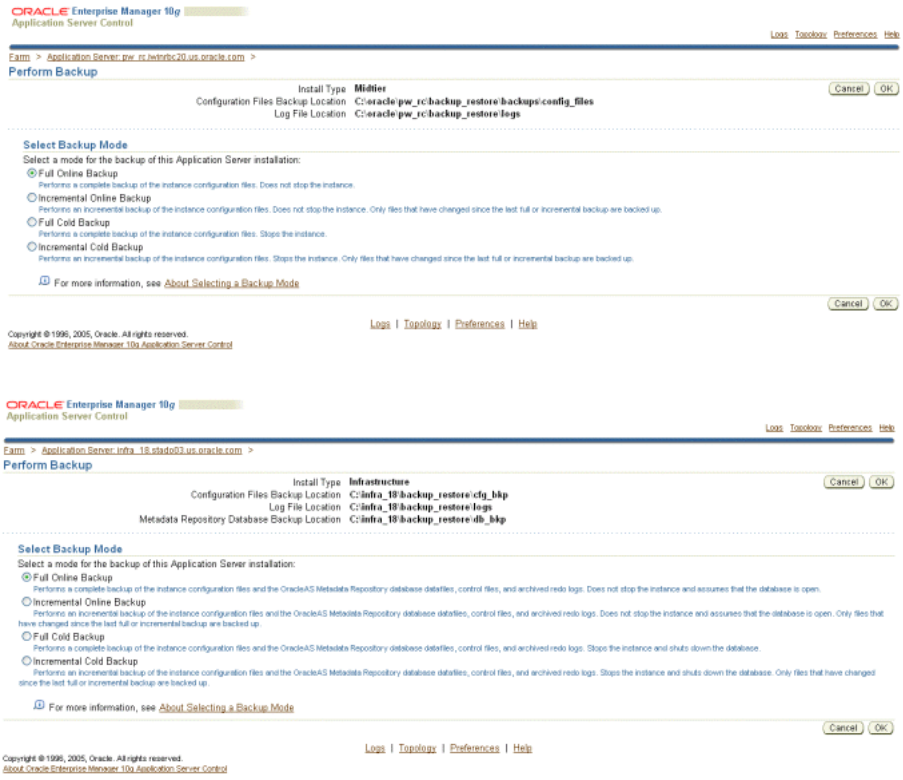
- The following information for the Metadata Repository:
  - Database version and patch level
  - Base language

- Character set
- Global database name
- SID

21.2.4 Performing an Instance Backup of Oracle Application Server Using Application Server Control Console

You can use the Oracle Enterprise Manager 10*g* Application Server Control Console to manage backup and recovery of an Oracle Application Server instance. Once you have performed a complete Oracle Application Server environment backup, you should perform subsequent instance backups after every administrative change, or, if this is not possible, on a regular basis. Perform the following steps to take a backup:

1. From the Home page for an application server instance, click **Backup/Recovery** to display the Backup/Recovery page.
2. Click **Perform Backup**. Depending on the install type, the middle tier backup screen or the Infrastructure backup screen displays:



3. Select the type of backup you want performed by clicking the radio button next to the type of backup. After the backup completes, a confirmation screen displays the results of the backup:

21.2.5 Performing an Oracle Application Server Instance Backup from the Command Line

This section describes how to perform various Oracle Application Server instance backups from the command line. An instance level backup backs up all the required components in an application server instance: configuration files, repositories (database or file-based) for the infrastructure and mid-tier.

Once you have performed a complete Oracle Application Server environment backup, you should perform subsequent instance backups after every administrative change, or, if this is not possible, on a regular basis.

Performing a Cold Backup of an Oracle Application Server Instance

Use the following command to perform a cold backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_cold
bkp_restore.bat -m backup_instance_cold
```

Performing an Incremental Cold Backup of an Oracle Application Server Instance

Use the following command to perform an incremental cold backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_cold_incr -l <level>
```

```
bkp_restore.bat -m backup_instance_cold_incr -l <level>
```

Performing an Online Backup of an Oracle Application Server Instance

Use the following command to perform an online backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_online
bkp_restore.bat -m backup_instance_online
```

Performing an Incremental Online Backup of an Oracle Application Server Instance

Use the following command to perform an incremental online backup of an Oracle Application Server instance:

```
bkp_restore.sh -m backup_instance_online_incr -l <level>
bkp_restore.bat -m backup_instance_online_incr -l <level>
```

21.2.6 Performing a Complete Oracle Application Server Environment Backup

This section describes how to perform a complete Oracle Application Server environment backup. A complete Oracle Application Server environment backup includes Identity Management metadata and Oracle Application Server Metadata Repository, which can be stored in the same database or different databases. You should backup the node after installation or after an upgrade. Perform the following tasks for each instance on the host:

Configuration Backup of the Node

Run the following command to create a backup of the node configuration:

On UNIX:

```
bkp_restore.sh -m configure
```

On Windows:

```
bkp_restore.bat -m configure
```

Node Backup Preparation

Run the following command to prepare a node for backup:

On UNIX:

```
bkp_restore.sh -m node_backup -o prepare
```

On Windows:

```
bkp_restore.bat -m node_backup -o prepare
```

Creating an Image Backup of the Instance

This task creates an archive of an instance that includes the Oracle home, oratab, central inventory, Windows registries and so forth. On UNIX, the command must be run from root. If you are performing a metadata repository or Infrastructure image backup, the database listener must be running. Run the following command to create an image backup of the instance:

On UNIX:

```
bkp_restore.sh -m node_backup  -o image_backup -P <archive path>
```

On Windows:

```
bkp_restore.bat -m node_backup  -o image_backup -P <archive path>
```

After the command completes, the backup is placed in the directory specified in *<archive path>*.

21.3 Recovering a Loss of Host Automatically

Oracle Application Server Backup and Recovery Tool provides an automated procedure to take a full backup of the instances on one host and restore them to a new host after loosing the original operating environment.

Loss of Host Automation (LOHA) automates the tasks necessary for the Oracle Application Server user to migrate Oracle Application Server instances from one host to another. The new host can be a different host running the same operating system or the same host after system re-imaging. LOHA provides a solution for a loss of host when you want to restore the original instances to a new environment without having to reinstall the instances and preserve the application data.

LOHA supports all middle-tier installations, and the new host's name can be the same or different from the original host. For metadata repositories and Infrastructure installations, only the target host name must be the same as the original host. For an Oracle Identity Management installation, full automation is supported if the new host name is the same as the original. For different host names, some manual work is required. LOHA does not support the Toplink standalone install type.

LOHA can move all the Oracle Application Server instances from one host to a new host provided that the new host does not have any other Oracle Application Server instances already running. You can restore a subset of the instances to the new host if the subset does not have any dependencies on the instances remaining on the old host. You cannot restore instances from multiple hosts to a single host.

LOHA can also be used to recover a corrupted instance on a host without affecting other instances on the same host.

This section contains the following topics:

- Preparing to Use Loss of Host Automation
- Enabling Loss of Host Automation
- Restoring a Node on a New Host
- Restoring a Host with Identity Management to a Host with a Different Name
- Recovering an Instance on the Same Host

21.3.1 Preparing to Use Loss of Host Automation

The Loss of Host Automation service is installed as part of the Backup and Recovery Tool. It is installed into the following directory:

On UNIX:

```
ORACLE_HOME/backup_restore/loha
```

On Windows:

```
ORACLE_HOME\backup_restore\loha
```

To use the Loss of Host Automation service, you must install and configure the Backup and Recovery Tool as described in Chapter 20, "Oracle Application Server

Backup and Recovery Tool". You must set ORACLE_HOME. If the installation is an Infrastructure, you must also set ORACLE_SID.

The Loss of Host service has the following prerequisites:

- The new host must have the same version of operating system and the same level of patches as required by Oracle Application Server.
- In the `config.inp` file, the `orainst_loc_path` field must be changed only if the instance is installed with the –invPtrLoc option installer command line option. It must be changed to reflect the nonstandard location of oraInst.loc.
- For Windows platforms, Windows Support Files (WSF) must be installed. You can obtain WSF from the Oracle Application Server installation cd.
- For Windows platforms, the Microsoft service utility sc.exe must be installed on both the original host and the new host. According to Microsoft, it is part of the NT ResourceKit. For Windows XP, the utility is part of the installation. For Windows 2000 platforms, it must be installed. Ensure that it is in the execution path.
- On the new host, jar (Windows) or tar (Unix) must be available to unpack the node archive. If your system has its own tar program, use it instead of GNU tar.
- The user must have administrative privileges on the system such that system or root level tasks can be performed.
- There should not be any other Oracle products installed on the new host. For example, if there are some Oracle Application Server instances on this new host, they must be shutdown and uninstalled cleanly.
- The user/group id on the new host must match that on the original host.
- Check port usage on the new host. Make sure there are not any processes using the same ports as any of the Oracle Application Server instances you are restoring. If any processes are using the same ports, reconfigure the processes to use different ports before restoring any Oracle Application Server instance.
- After completing the restore, the same mount point and full path, as the original middle-tier Oracle home, are preserved. Ensure that the Oracle home parent directory is on a file system with enough space to hold the middle-tier installation, and that the directory is owned by the same user and group as on the original host.

For a host with Oracle Identity Management and the new host has a different name, see "Restoring a Host with Identity Management to a Host with a Different Name" for instructions on restoring the host.

21.3.2 Enabling Loss of Host Automation

The following tasks must be performed, for each instance on the original host, to enable the Loss of Host Automation service:

Configuration Backup of the Node

You should backup the node after installation or after an upgrade. Run the following command to create a backup of the node configuration:

On UNIX:

```
bkp_restore.sh -m configure
```

On Windows:

```
bkp_restore.bat -m configure
```

Node Backup Preparation

During node backup preparation, the Loss of Host Automation service determines the following information about the current host:

- operating system
- host name
- ip address
- user/group id
- install type
- central inventory location
- Oracle home locations
- Windows registry and all Windows services created for all Oracle homes

The service also creates an instance backup with this operation.

Run the following command to prepare a node for backup:

On UNIX:

```
bkp_restore.sh -m node_backup -o prepare
```

On Windows:

```
bkp_restore.bat -m node_backup -o prepare
```

Creating an Image Backup of the Original Host

This task creates an archive of an instance that includes the original Oracle home, oratab, central inventory, Windows registries and so forth. On UNIX, the command must be run from root. If you are performing a metadata repository or Infrastructure image backup, the database listener must be running. Run the following command to create an image backup of the original instance:

On UNIX:

```
bkp_restore.sh -m node_backup  -o image_backup -P <archive path>
```

On Windows:

```
bkp_restore.bat -m node_backup  -o image_backup -P <archive path>
```

After the command completes, the backup is placed in the directory specified in *<archive path>*.

21.3.3 Restoring a Node on a New Host

The commands in this section restore a node on a new host after a loss of host. Before performing the following steps, ensure that all the prerequisites in Section 21.3.1, "Preparing to Use Loss of Host Automation" are fulfilled.

After unpacking the archive, ensure that the database `flashback_recovery_area` is the same as the original if it is located outside the `ORACLE_HOME` for the instance.

For an instance with a file-based repository, the dcm-daemon process should be up on the member nodes while restoring to the new host. After restoring the instance, for all

member nodes edit the following line in *ORACLE_HOME*/dcm/config/dcmCache.xml:

```
<discoverer ip="host-name" discovery-port="repository-id"original="false"
xmlns=""/>
```

In the preceding example, *host-name* is the name of the new host and *repository-id* is the id of the new host.

The following commands must be run in order.

1. Unpack the backup archive of the old node:

   On UNIX, login as root:

   ```
   cd /
   tar -xvpf <archive_name>
   ```

   On Windows:

   ```
   jar -xvf <archive_name>
   ```

2. The following command restores Oracle Universal Installer related metadata such as oratab (UNIX), Windows registries, and central inventory on the new host. If multiple instances are to be restored, this operation should be performed only for the first instance. The command must be run as root on UNIX.

   On UNIX:

   ```
   bkp_restore.sh -m node_restore -o sys_init
   ```

   On Windows:

   ```
   bkp_restore.bat -m node_restore -o sys_init
   ```

3. The following command registers the instance with oratab and the central inventory, it also sets up daemon start/stop script by running root.sh on UNIX, or, on Windows, Windows services are created. The command must be run as root on UNIX.

   On UNIX:

   ```
   bkp_restore.sh -m node_restore -o inst_register
   ```

   On Windows:

   ```
   bkp_restore.bat -m node_restore -o inst_register
   ```

4. This command reconfigures the instance on the new host. This includes IP changing, database restore, database tempfile setup, config backup restore and so forth depending on the install type. Prior to running the command, run `opmnctl shutdown` and `emctl stop iasconsole` to ensure that opmn and Enterprise Manager processes are not using ports required by the reconfigure process. The command must be run as the owner of the instance. The path to the instance backups must be valid. If database RMAN logs error RMAN-06054 in the `restore_repos` log file, it should be treated as innocuous.

   On UNIX:

   ```
   bkp_restore.sh -m node_restore -o inst_reconfigure -t config_bkp_timestamp
   ```

   On Windows:

   ```
   bkp_restore.bat -m node_restore -o inst_reconfigure -t config_bkp_timestamp
   ```

Without a timestamp argument, this command shows all the available instance backups. For a successful completion of this operation, ensure that all the other required services are up and running if they do not belong to this instance. Those required services can include Oracle Identity Management, Oracle Application Server Metadata Repository and Infrastructure. If these services must be restored, they must be done in the proper order.

LOHA will not detect port conflicts on the new host. It is recommended that you do not run other applications using the same TCP ports that are to be used by the restored instance. Any port conflict will cause this operation to fail.

Business Intelligence Forms Installations

For Oracle Business Intelligence Forms installations, after the instance is restored to the new host, perform the following steps:

1. In the backup archive directory, find the archive `config_bkp_<timestamp>.jar`, where <*timestamp*> is the timestamp used in LOHA `inst_reconfigure` operation.
2. Unpack the archive into a temporary directory.
3. Copy *temp_dir*`/j2ee/OC4J_BI_Forms/config/oc4j.properties` to *$ORACLE_HOME*`/j2ee/OC4J_BI_Forms/config/oc4j.properties`.
4. Remove the temporary directory and restart the instance.

21.3.4 Restoring a Host with Identity Management to a Host with a Different Name

To restore a host with Identity Management to a new host with a different name, perform the following procedures:

1. Perform the steps in "Preparing to Use Loss of Host Automation".
2. Perform the steps in "Enabling Loss of Host Automation".
3. Perform steps 1 through 3 in "Restoring a Node on a New Host".
4. Create a backup copy of the `configtool.xml.tmpl` file. The file directory is *ORACLE_HOME*`/chgip/config/`. Save the backup copy to another directory. Edit the original `configtool.xml.tmpl` file and remove the following lines pertaining to the `updateConfig` parameter:

```
<ConfigTool Name="DCM" Desc="Distributed Configuration Manager"
InstallType="Core">
   <Command>%ORACLE_HOME%\dcm\bin\dcmctl.bat</Command>
   <Parameter Name="" Value="updateConfig"/>
</ConfigTool>

<ConfigTool Name="DCM" Desc="Distributed Configuration Manager"
InstallType="Portals">
   <Command>%ORACLE_HOME%\dcm\bin\dcmctl.bat</Command>
   <Parameter Name="" Value="updateConfig"/>
</ConfigTool>

<ConfigTool Name="DCM" Desc="Distributed Configuration Manager"
InstallType="BIServices">
   <Command>%ORACLE_HOME%\dcm\bin\dcmctl.bat</Command>
   <Parameter Name="" Value="updateConfig"/>
</ConfigTool>

<ConfigTool Name="DCM" Desc="Distributed Configuration Manager"
```

```
InstallType="Infrastructure_ID">
   <Command>%ORACLE_HOME%\dcm\bin\dcmctl.bat</Command>
   <Parameter Name="" Value="updateConfig"/>
</ConfigTool>

<ConfigTool Name="DCM" Desc="Distributed Configuration Manager"
InstallType="Infrastructure">
   <Command>%ORACLE_HOME%\dcm\bin\dcmctl.bat</Command>
   <Parameter Name="" Value="updateConfig"/>
</ConfigTool>

<ConfigTool Name="DCM" Desc="Distributed Configuration Manager"
InstallType="OCS">
   <Command>%ORACLE_HOME%\dcm\bin\dcmctl.bat</Command>
   <Parameter Name="" Value="updateConfig"/>
</ConfigTool>
```

Save the file, and then run the chgiphost script:

On UNIX:

```
ORACLE_HOME/chgip/scripts/chgiphost.sh -mid
```

On Windows:

```
ORACLE_HOME/chgip/scripts/chgiphost.bat -mid
```

5. Use the following commands to restore DCM managed components:
   - On UNIX:

     ```
     bkp_restore.sh -m restore_config -F dcm-resyncforce
     ```

   - On Windows:

     ```
     bkp_restore.bat -m restore_config -F dcm-resyncforce
     ```

6. Perform the following steps to start the Middle-tier instance.
   a. Start OPMN and OPMN-managed processes:

      ```
      ORACLE_HOME/opmn/bin/opmnctl startall
      ```

   b. Start the Application Server Control Console:

      ```
      ORACLE_HOME/bin/emctl start iasconsole
      ```

7. The chgiphost script must be run with the updateConfig parameter in *ORACLE_HOME*/chgip/config/configtool.xml.tmpl file. Use the original version of configtool.xml.tmpl file. Replace the modified version of this file (from step 4) with the backed-up copy.

   Run the chgiphost script to update the host information for DCM-managed components:

   ```
   chgiphost.sh -mid
   ```

21.3.5 Recovering an Instance on the Same Host

When an instance of Oracle Application Server requires an image restore to correct a problem, you can use LOHA to recover the instance. Perform the following steps to recover the instance:

1. Completely shutdown the instance.

2. Perform step 1 of Section 21.3.3, "Restoring a Node on a New Host" to unpack the latest image backup of the instance.
3. Perform steps 3 and 4 of Section 21.3.3, "Restoring a Node on a New Host" to register and configure the instance.

   If the instance has any dependencies on other instances of Oracle Application Server, the other instances must be up and running.

22

Recovery Strategies and Procedures

This chapter describes Oracle Application Server recovery strategies and procedures for different types of failures and outages.

It contains the following topics:

22.1 Recovery Strategies

This section describes Oracle Application Server recovery strategies for different types of failures and outages. It contains the following topics:

22.1.1 Recovery Strategies for Data Loss, Host Failure, or Media Failure (Critical)

This section describes recovery strategies for outages that involve actual data loss or corruption, host failure, or media failure where the host or disk cannot be restarted and are permanently lost. This type of failure requires some type of data restoration before the Oracle Application Server environment (middle tier, Infrastructure, or both) can be restarted and continue with normal processing.

The strategies in this section use point-in-time recovery of the middle tier and Infrastructure. This means that, no matter where the loss occurred, the Infrastructure and the middle tier are always restored together so they are in sync as they were at the time of the last backup. Notice that in an Oracle Application Server environment recovery, the Infrastructure is always restored before the middle tier.

Assumptions

The following assumptions apply to the recovery strategies in this section:

- `ARCHIVELOG` mode was enabled for all Metadata Repository backups.
- Complete recovery of the database can be performed, that is, no redo log files have been lost.
- No administrative changes were made since the last backup. If administrative changes were made since the last backup, they will need to be reapplied after recovery is complete.

See Also: Appendix G, "Examples of Administrative Changes" to learn more about administrative changes

Determining Which Strategy to Use

Recovery strategies are listed in the following tables:

- Table 22–1, " Recovery Strategies for Data Loss, Host Failure, and Media Failure in Infrastructures"

  Use this table if you experience data loss, host failure, or media failure in an Infrastructure installation. Find the type of loss and follow the recommended procedure. The procedures apply to Infrastructure that are installed into a single Oracle home, as well as Infrastructures with Identity Management in one Oracle home and a Metadata Repository in another Oracle home or host.

- Table 22–2, " Recovery Strategies for Data Loss, Host Failure, and Media Failure in Middle-Tier Instances"

  Use this table if you experience data loss, host failure, or media failure in a middle-tier installation. Find the type of loss and follow the recommended procedure.

If the loss occurred in both the Infrastructure and middle tier, follow the Infrastructure recovery strategy first, then the middle tier.

Table 22–1 Recovery Strategies for Data Loss, Host Failure, and Media Failure in Infrastructures

Type of Loss	Recovery Strategies
Loss of host	You can restore to a new host that has the same hostname. Follow the procedure in Section 22.2.3, "Restoring an Infrastructure to a New Host".
Oracle software/binary loss or corruption	If any Oracle binaries have been lost or corrupted, you must recover the entire Infrastructure. Follow the procedure in Section 22.2.2, "Restoring an Infrastructure to the Same Host".
Database or data failure of the Metadata Repository (datafile loss, control file loss, media failure, disk corruption)	If the Metadata Repository is corrupted due to data loss or media failure, you can restore and recover it. Follow the procedure in Section 22.2.5, "Restoring and Recovering the Metadata Repository".
Deletion or corruption of configuration files	If you lose any configuration files in the Infrastructure Oracle home, you can restore them. Follow the procedure in Section 22.2.6, "Restoring Infrastructure Configuration Files".
Deletion or corruption of configuration files and data failure of the Metadata Repository	If you lose configuration files and the Metadata Repository is corrupted, you can restore and recover both. Follow these procedures: 1. Section 22.2.6, "Restoring Infrastructure Configuration Files" 2. Section 22.2.5, "Restoring and Recovering the Metadata Repository"

Table 22–2 Recovery Strategies for Data Loss, Host Failure, and Media Failure in Middle-Tier Instances

Type of Loss	Recovery Strategies
Loss of host	If the host has been lost, you have two options: ■ You can restore to a new host that has the same hostname and IP address. ■ You can restore to a new host that has a different hostname and IP address. In either case, follow the procedure in Section 22.2.8, "Restoring a Middle-Tier Installation to a New Host". Note that if the original host had a middle-tier installation and an Infrastructure, you cannot restore the middle-tier to a host with a different hostname or IP address.
Oracle software/binary deletion or corruption	If any Oracle binaries have been lost or corrupted, you must restore the entire middle tier to the same host. Follow the procedure in Section 22.2.7, "Restoring a Middle-Tier Installation to the Same Host".
Deletion or corruption of configuration files	If you lose any configuration files in the middle tier Oracle home, you can restore them. Follow the procedure in Section 22.2.9, "Restoring Middle-Tier Configuration Files".

22.1.2 Recovery Strategies for Process Failures and System Outages (Non-Critical)

This section describes recovery strategies for process failures and system outages. These types of outages do not involve any data loss, and therefore do not require any files to be recovered. In some cases, failure may be transparent and no manual intervention is required to recover the failed component. However, in some cases, manual intervention is required to restart a process or component. While these strategies do not strictly fit into the category of backup and recovery, they are included in this book for completeness.

Determining Which Strategy to Use

Recovery strategies for process failures and system outages are listed in the following tables:

- Table 22–3, " Recovery Strategies for Process Failures and System Outages in Infrastructures"

  Use this table if you experience a failure or outage in an Infrastructure. Find the type of outage and follow the recommended procedure. The procedures apply to Infrastructures that are installed into a single Oracle home, as well as Infrastructures with Identity Management in one Oracle home and a Metadata Repository in another Oracle home or host.

- Table 22–4, " Recovery Strategies for Process Failures and System Outages in Middle-Tier Instances"

  Use this table if you experience a failure or outage on a middle-tier installation. Find the type of outage and follow the recommended procedure. The table contains UNIX commands. You can use the same commands on Windows by inverting the slashes, or you can use the Services tool in the Control Panel.

Table 22–3 Recovery Strategies for Process Failures and System Outages in Infrastructures

Type of Outage	How to Check Status and Restart
Host failure - no data loss	**To restart:** 1. Restart the host. 2. Start the Infrastructure. Refer to Section 3.2.1, "Starting OracleAS Infrastructure".

Metadata Repository instance failure (loss of the contents of a buffer cache or data residing in memory)	**To check status:**
	1. Try connecting to the database using SQL*Plus.
	2. Check the state as follows:
	`SQL> select status from v$instance;`
	To restart:
	`sqlplus /nolog`
	`SQL> connect sys/password as sysdba`
	`SQL> startup`
	`SQL> quit`

Metadata Repository listener failure	**To check status:**
	`lsnrctl status`
	To restart:
	`lsnrctl start`

Oracle Internet Directory server process (`oidldapd`) failure	**To check status:**
	`ldapcheck`
	To restart:
	`opmnctl startproc ias-component=OID`

Oracle Internet Directory monitor process (`oidmon`) failure	**To check status:**
	`ldapcheck`
	To restart:
	`opmnctl startproc ias-component=OID`

Application Server Control Console failure	**To check status:**
	`emctl status iasconsole`
	To restart:
	`emctl start iasconsole`

Oracle HTTP Server process failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl startproc ias-component=HTTP_Server`

OC4J instance failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl startproc process-type=OC4J_instance_name`

Table 22–3 (Cont.) Recovery Strategies for Process Failures and System Outages in Infrastructures

Type of Outage	How to Check Status and Restart

Delegated Administration Service instance failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl startproc ias-component=OC4J process-type=OC4J_SECURITY`

OPMN daemon failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl start`

Table 22–4 Recovery Strategies for Process Failures and System Outages in Middle-Tier Instances

Type of Outage	How to Check Status and Restart
Host failure - no data loss	**To restart:** 1. Restart the host. 2. Start the middle tier. Refer to Section 3.2.3, "Starting a Middle-Tier Instance"

Application Server Control Console failure	**To check status:**
	`emctl status iasconsole`
	To restart:
	`emctl start iasconsole`

Oracle HTTP Server process failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl startproc ias-component=HTTP_Server`

OC4J instance failure	**To check status:** `opmnctl status` **To restart:** `opmnctl startproc process-type=`*`OC4J_instance_name`*

OPMN daemon failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl start`

OracleAS Web Cache failure	**To check status:**
	`opmnctl status`
	To restart:
	`opmnctl startproc ias-component=WebCache`

22.2 Recovery Procedures

This section contains the procedures for performing different types of recovery.

It contains the following topics:

- Using Application Server Control Console to Recover an Oracle Application Server Instance
- Restoring an Infrastructure to the Same Host
- Restoring an Infrastructure to a New Host
- Restoring an Identity Management Instance to a New Host
- Restoring and Recovering the Metadata Repository
- Restoring Infrastructure Configuration Files
- Restoring a Middle-Tier Installation to the Same Host
- Restoring a Middle-Tier Installation to a New Host
- Restoring Middle-Tier Configuration Files
- Restoring an Oracle Application Server Instance

22.2.1 Using Application Server Control Console to Recover an Oracle Application Server Instance

You can use the Oracle Enterprise Manager 10*g* Application Server Control Console to manage backup and recovery of an Oracle Application Server instance. Use the following procedure to recover an Oracle Application Server instance:

Before performing a restore operation (`restore_instance` or `restore_config`) on an instance in a cluster, all OC4J processes across the cluster must be stopped. Use the following command to stop the processes:

```
ORACLE_HOME/opmn/bin/opmnctl @cluster stopproc ias-component=OC4J
```

Some OC4J components (such as Wireless) do not have `ias-component=OC4J`. For these components use the uniqueid value to stop the OC4J process. To determine which components have a uniqueid, use the following command:

```
ORACLE_HOME\opmn\bin\opmnctl @cluster status -fmt %typ%uid%prt -noheaders
```

The following is an example of the output from the command:

```
CUSTOM | N/A | DSA
LOGLDR | N/A | logloaderd
DCMDaemon | 1444413512 | dcm-daemon
WebCache | 1500577871 | WebCache
WebCache-admin | 1500577872 | WebCacheAdmin
OHS | 1500577870 | HTTP_Server
performance | 1500577873 | performance_server
messaging | 1500577874 | messaging_server
OC4J | 1500577865 | OC4J_Wireless
```

Stop all the OC4J processes, for which the second column (uid) value is not "N/A", with the following command:

```
ORACLE_HOME\opmn\bin\opmnctl @cluster stopproc uniqueid=1500577865

opmnctl: stopping opmn managed processes...
```

1. From the Home page for an application server instance, click **Backup/Recovery** to display the Backup/Recovery page.
2. Click **Perform Recovery**. Depending on the type of installation, the middle tier recovery screen or the Infrastructure recovery screen displays:

3. For the Infrastructure recovery screen, you can click the **Recover Control Files** check box to recover the control files for the instance. Click **OK** to perform the restore.

After the restore operation is complete, use the following command to restart the OC4J processes across the cluster:

```
ORACLE_HOME/opmn/bin/opmnctl @cluster startproc ias-component=OC4J
```

For components that use uniqueid, you can restart their process by using the appropriate ias-component value or by using the following command:

```
opmnctl startall
```

22.2.2 Restoring an Infrastructure to the Same Host

This section describes how to restore an Infrastructure to the same host. You can use this procedure when you have lost some or all of your Oracle binaries.

Refer to Section 21.3.5, "Recovering an Instance on the Same Host" to restore the image backup of the Infrastructure Oracle home from your complete Oracle Application Server environment backup.

> **Note:** If your Infrastructure is split and has Identity Management in one Oracle home, and the Metadata Repository in another Oracle home, perform this step on both Oracle homes.

Note: If you receive a WWC-41439 error while trying to login to the Portal Home page, do one or all of the following:

- Remove aliases from your Apache configuration.
- Include the domain in the ServerName parameter.
- Fix the Host in the IASInstance element and ListenPort in the WebCacheComponent element in `iasconfig.xml` and run `ptlconfig -dad portal-site`. The `ptlconfig` script and the `iasconfig.xml` file is normally located in the directory `portal/conf` under the OracleAS Portal and OracleAS Wireless middle-tier home.

22.2.3 Restoring an Infrastructure to a New Host

Refer to Section 21.3.3, "Restoring a Node on a New Host" to perform the following types of restores:

- Restore an Infrastructure to the same host after the operating system has been reinstalled. The hostname must remain the same on the host.
- Restore an Infrastructure to a new host that has the same hostname as the original host.

Note: If your Infrastructure is split and has Identity Management in one Oracle home, and the Metadata Repository in another Oracle home, perform the procedures on both Oracle homes as described in Section 22.2.4, "Restoring an Identity Management Instance to a New Host" and Section 22.2.5.2, "Restoring and Recovering the Metadata Repository to a New Host".

22.2.4 Restoring an Identity Management Instance to a New Host

Refer to Section 21.3, "Recovering a Loss of Host Automatically" to perform the following types of restores:

- Restore Identity Management to the same host after the operating system has been reinstalled. The hostname must remain the same on the host.
- Restore Identity Management to a new host that has the same or different hostname as the original host.

22.2.5 Restoring and Recovering the Metadata Repository

The section describes how to restore and recover the Metadata Repository. You can use this when there has only been corruption to the Metadata Repository, and not to any other files in the Oracle home.

Restore and recover the Metadata Repository from your latest backup using your own procedure or the OracleAS Backup and Recovery Tool. Restart all Infrastructure processes after restoring a Metadata Repository.

The following sections describe Oracle recommended procedures for using the OracleAS Backup and Recovery Tool to restore and recover the Metadata Repository:

- Restoring and Recovering the Metadata Repository to the Same Host
- Restoring and Recovering the Metadata Repository to a New Host

22.2.5.1 Restoring and Recovering the Metadata Repository to the Same Host

This section covers several circumstances under which you may need to restore and recover the Metadata Repository to the same host:

- Corrupted or Lost Datafile
- Corrupted or Lost Control File
- Point-in-Time Recovery and Flashback Recovery

Corrupted or Lost Datafile

If a datafile is corrupted or lost, you can use the following command to restore from the latest backup and perform a full recovery:

For UNIX:

```
bkp_restore.sh -m restore_repos
```

For Windows:

```
bkp_restore.bat -m restore_repos
```

Corrupted or Lost Control File

If a control file is corrupted or lost, you can use the following command to restore a control file backup, restore the datafiles, and perform a full recovery:

For UNIX:

```
bkp_restore.sh -m restore_repos -c
```

For Windows:

```
bkp_restore.bat -m restore_repos -c
```

When you use the `-c` option, it restores the control file. This causes entries for tempfiles in locally-managed temporary tablespaces to be removed. You must add a new tempfile to the TEMP tablespace, or Oracle will display error ORA-25153: Temporary Tablespace is Empty.

To add a tempfile to the TEMP tablespace:

```
SQL> alter tablespace "TEMP" add tempfile 'ORACLE_HOME/oradata/GDB/
temp01.dbf' size 5120K autoextend on next 8k maxsize unlimited;
```

GDB is the first part of the global database name.

Note that when you restore a control file, the tool performs an "alter database open resetlogs." This invalidates all backups and archivelogs. You should immediately perform a complete cold backup of the Metadata Repository, which will serve as the new baseline for your subsequent partial online backups.

Point-in-Time Recovery and Flashback Recovery

If you lost configuration files in your middle-tier or Infrastructure installation and restored those, you may want to restore or flashback the database to the same point-in-time as the configuration file backup. You can do this using one of the following commands:

For UNIX:

```
bkp_restore.sh -m restore_repos -u timestamp

bkp_restore.sh flashback_repos -u timestamp
```

For Windows:

```
bkp_restore.bat -m restore_repos -u timestamp

bkp_restore.bat flashback_repos -u timestamp
```

Flashback recovery to a point-in-time can undo any logical data corruption or user error. Flashback cannot undo physical data corruption due to media failure. Using the `restore_repos` command, you can recover and restore the database to a point-in-time for both logical and physical data corruption. However, Flashback is faster at recovering logical data corruption because it does not require restoring backups.

You can specify any time between the time of your first backup and the current time, as long as none of the online redo logs were compromised. If any online redo logs are missing or corrupted, the latest time that can be specified is the time at which the last backup was made.

Note that when you do point-in-time recovery, the tool performs an "alter database open resetlogs." This invalidates all backups and archivelogs. You should immediately perform a complete cold backup of the Metadata Repository, which will serve as the new baseline for your subsequent partial online backups.

The Backup and Recovery Tool supports point-in-time recovery through resetlogs in all Oracle databases: Infrastructure with Identity Manager and Metadata Repository, RepCA, and generic Oracle databases (for example, OCS Infostore). The following is an example of a point-in-time recovery through resetlogs:

At time T1, a backup of the database is taken. Changes are made to the database. At time T2, a new backup is taken. More changes are made to the database. At time T3, another backup is taken. More changes are made. At time T4, the user restores and recovers the database to T3. Since this is a point-in-time recovery, the Backup and Recovery Tool opens the database with resetlogs to start a new log sequence after the recovery. At time T5, the user restores and recovers the database to T2 through the resetlogs created at T4.

Multiple backward point-in-time recoveries are supported for backups taken using `backup_instance_cold`, `backup_instance_online`, `backup_instance_incr`. To perform multiple backward point-in-time recoveries using `backup_cold`, `backup_online`, and `backup_incr`, you must follow the backup operation immediately with `backup_config`.

22.2.5.2 Restoring and Recovering the Metadata Repository to a New Host

When you restore the Metadata Repository to a new host (with the same hostname), the new host will not have the online redo logs that existed on the original host. Therefore, you cannot perform a full recovery; RMAN would give an error stating that it cannot find a certain log file (the online redo log file). Instead, you should do a point-in-time recovery using a time sometime between the first and most recent backup. You can do this by specifying the proper timestamp for the LOHA reconfigure operation. Use the procedure at Section 21.3.3, "Restoring a Node on a New Host" to restore the Metadata Repository.

During the LOHA reconfigure process, if the RMAN command returns an error and the log shows that the datafiles were restored and recovered, then LOHA will issue an "alter database open resetlogs" and the database will be opened in a consistent state. If no datafiles were restored and recovered, it is most likely that an early timestamp was specified. You should retry the command with a later timestamp.

LOHA uses the -c option during the restore process which means that the control file is restored from backup. This causes entries for tempfiles in locally-managed temporary tablespaces to be removed and a new TEMP tablespace to be added automatically. Restoring the control file means that an "alter database open resetlogs" is always performed, which invalidates all backups and archivelogs. You should immediately perform a complete cold backup of the Metadata Repository, which will serve as the new baseline for your subsequent partial online backups.

22.2.6 Restoring Infrastructure Configuration Files

This section describes how to restore the configuration files in an Infrastructure Oracle home. You can use this procedure when configuration files have been lost or corrupted.

It contains the following tasks:

- Task 1: Stop the Infrastructure
- Task 2: Restore Infrastructure Configuration Files
- Task 3: Apply Recent Administrative Changes
- Task 4: Start the Infrastructure

Task 1: Stop the Infrastructure

Refer to Section 3.2.2, "Stopping OracleAS Infrastructure" for instructions.

Task 2: Restore Infrastructure Configuration Files

> **Note:** If your Infrastructure is split and has Identity Management in one Oracle home, and the Metadata Repository in another Oracle home, perform this task on both Oracle homes.

Restore all configuration files from your most recent backup. You can perform this task using your own procedure or the OracleAS Backup and Recovery Tool. For example, to do this using the tool:

- On UNIX systems:

```
bkp_restore.sh -m restore_config -t timestamp
```

- On Windows systems:

```
bkp_restore.bat -m restore_config -t timestamp
```

> **See Also:** Chapter 20, "Oracle Application Server Backup and Recovery Tool" for more information.

Task 3: Apply Recent Administrative Changes

If you made any administrative changes since the last time you did an online backup, reapply them now.

> **See Also:** Appendix G, "Examples of Administrative Changes" to learn more about administrative changes

Task 4: Start the Infrastructure

Refer to Section 3.2.1, "Starting OracleAS Infrastructure" for instructions.

22.2.7 Restoring a Middle-Tier Installation to the Same Host

To restore a middle-tier installation to the same host, refer to Section 21.3.5, "Recovering an Instance on the Same Host".

22.2.8 Restoring a Middle-Tier Installation to a New Host

This section describes how to restore and recover a middle-tier installation to a new host. You can use this procedure to:

- Restore a middle-tier installation to the same host after the operating system has been reinstalled.
- Restore a middle-tier installation to a new host. The new host may have the same hostname and IP address as the original host, or a different hostname, IP address, or both.

If the DCM repository is a database, start the OPMN and Oracle Internet Directory processes on the corresponding infrastructure instance.

- Use the following command to start the OPMN process:

```
opmnctl start
```

- Use the following command to start the Oracle Internet Directory process:

```
opmnctl startproc ias-component=OID
```

Use the following command to check if the DCM repository is a database or a file-based repository:

```
ORACLE_HOME/dcm/bin/dcmctl whichfarm
```

The preceding command returns one of the following messages:

```
Repository Type: Database => uses a database repository
Repository Type: Distributed File Based => uses a file based repository
```

Perform the steps in Section 21.3.3, "Restoring a Node on a New Host" to restore the image backup, system files and instance reconfiguration. Note that the middle-tier configuration remains in the same state as the original instance. If the hostname remains the same, run an instance restore to bring the instance to the desired point in time. If the hostname is different, the state cannot be changed since backups of the original host are not valid for a different hostname.

Note: There is a special step required for updating OracleAS Portal and OracleAS Wireless when you change the hostname.

When you change the hostname, the OracleAS Wireless server URL changes to use the new hostname.

You must update OracleAS Portal with the new OracleAS Wireless service URL.

Refer to the section on "*Updating the Oracle AS Wireless Portal Service URL Reference*" in "*Oracle Application Server Portal Configuration Guide*" for instructions.

22.2.9 Restoring Middle-Tier Configuration Files

This section describes how to restore the configuration files in a middle-tier Oracle home. Use this procedure when configuration files have been lost or corrupted.

It contains the following tasks:

- Task 1: Stop the Middle-Tier Instance
- Task 2: Restore Middle-Tier Configuration Files
- Task 3: Apply Recent Administrative Changes
- Task 4: Start the Middle-Tier Instance

Task 1: Stop the Middle-Tier Instance

Refer to Section 3.2.4, "Stopping a Middle-Tier Instance" for instructions.

If the middle-tier instance uses a DCM repository (file-based or database), make sure the DCM repository is up.

Task 2: Restore Middle-Tier Configuration Files

Restore all configuration files from your most recent backup. You can perform this task using your own procedure or the OracleAS Backup and Recovery Tool. For example, to do this using the tool:

- For UNIX systems:

```
bkp_restore.sh -m restore_config -t timestamp
```

- For Windows systems:

```
bkp_restore.bat -m restore_config -t timestamp
```

> **See Also:** Chapter 20, "Oracle Application Server Backup and Recovery Tool" for more information

Task 3: Apply Recent Administrative Changes

If you made any administrative changes since the last time you did an online backup, reapply them now.

> **See Also:** Appendix G, "Examples of Administrative Changes" to learn more about administrative changes

Task 4: Start the Middle-Tier Instance

Refer to Section 3.2.3, "Starting a Middle-Tier Instance" for instructions.

22.2.10 Restoring a File-Based Repository to a New Host

This section describes how to restore a DCM file-based repository to a new host. This section contains the following tasks:

- Task 1: Restore Image Backup, System Files and Instance Reconfiguration
- Task 2: Inform the Original Host That It Is No Longer a Repository Host (If Required)

Task 1: Restore Image Backup, System Files and Instance Reconfiguration

If the DCM repository is a database, start the OPMN and Oracle Internet Directory processes on the corresponding infrastructure instance.

- Use the following command to start the OPMN process:

  ```
  opmnctl start
  ```

- Use the following command to start the Oracle Internet Directory process:

  ```
  opmnctl startproc ias-component=OID
  ```

  Use the following command to check if the DCM repository is a database or a file-based repository:

  ```
  ORACLE_HOME/dcm/bin/dcmctl whichfarm
  ```

  The preceding command returns one of the following messages:

  ```
  Repository Type: Database => uses a database repository
  Repository Type: Distributed File Based => uses a file based repository
  ```

Perform the steps in Section 21.3.3, "Restoring a Node on a New Host" to restore the image backup, system files and instance reconfiguration.

Task 2: Inform the Original Host That It Is No Longer a Repository Host (If Required)

Now that the file-based repository is restored to the new host, the original host may need to be informed that it is no longer a repository host. If the new host was already a part of the farm and is not a replacement for the original host, and the original host is still part of the farm, execute the following command on the original host:

```
dcmctl repositoryrelocated
```

22.2.11 Restoring an Oracle Application Server Instance

Use the following command to restore an Oracle Application Server instance to a particular point in time:

```
bkp_restore.sh -m restore_instance -t 2004-09-21_06-12-45 -c

bkp_restore.bat -m restore_instance -t 2004-09-21_06-12-45 -c
```

Before performing a restore operation (`restore_instance` or `restore_config`) on an instance in a cluster, all OC4J processes across the cluster must be stopped. Use the following command to stop the processes:

```
ORACLE_HOME/opmn/bin/opmnctl @cluster stopproc ias-component=OC4J
```

Some OC4J components (such as Wireless) do not have `ias-component=OC4J`. For these components use the uniqueid value to stop the OC4J process. To determine which components have a uniqueid, use the following command:

```
ORACLE_HOME\opmn\bin\opmnctl @cluster status -fmt %typ%uid%prt -noheaders
```

The following is an example of the output from the command:

```
CUSTOM | N/A | DSA
LOGLDR | N/A | logloaderd
DCMDaemon | 1444413512 | dcm-daemon
WebCache | 1500577871 | WebCache
WebCache-admin | 1500577872 | WebCacheAdmin
OHS | 1500577870 | HTTP_Server
performance | 1500577873 | performance_server
messaging | 1500577874 | messaging_server
OC4J | 1500577865 | OC4J_Wireless
```

Stop all the OC4J processes, for which the second column (uid) value is not "N/A", with the following command:

```
ORACLE_HOME\opmn\bin\opmnctl @cluster stopproc uniqueid=1500577865

opmnctl: stopping opmn managed processes...
```

After the restore operation is complete, use the following command to restart the OC4J processes across the cluster:

```
ORACLE_HOME/opmn/bin/opmnctl @cluster startproc ias-component=OC4J
```

For components that use uniqueid, you can restart their process by using the appropriate ias-component value or by using the following command:

```
opmnctl startall
```

23

Troubleshooting the Backup and Recovery Tool

This chapter describes common problems that you might encounter when using the Backup and Recovery Tool, and explains how to solve them. It contains the following topic:

- Problems and Solutions

23.1 Problems and Solutions

This section describes common problems and solutions. It contains the following topics:

- Receiving restore_config Operation Fails Error
- Receiving Missing Files Messages During restore_config Operation
- File-Based Repository Restoration Fails
- Cannot Run a Cold Backup on Identity Management or J2EE Instance
- Failure Due to Loss or Corruption of OPMN.XML File
- A restore_config Operation Fails
- Backup Operation Fails on a DCM File-Based Repository
- Timeout Occurs While Trying to Stop Processes Using opmnctl stopall
- Using the Backup and Recovery Tool to Perform a Recovery Fails Due to an Unknown Log Sequence Number
- Enterprise Manager Cannot Access Restored Nodes on New Hosts
- Restore of Portal Fails After Deleting OC4J Instance
- Cold Backups Do Not Shut Down All Databases in RAC Environment
- A restore_instance Fails at restore_repos Stage
- Changing ORACLE_HOME May Cause Backup or Recovery Failure
- Restore Operation Changes Farm Topology Leaving an Instance in Inconsistent State
- Post-deployment Changes to Configuration Files Are Lost After Restoring DCM-Managed Components

23.1.1 Receiving restore_config Operation Fails Error

A restore_config operation fails.

Problem

A restore_config operation fails with the following error:

```
C:\OracleAS\IM_1128/dcm/bin/dcmctl.bat applyarchiveto -archive
2004-11-29_11-23-18 -script

ADMN-906025
Base Exception:
The exception, 100999, occurred at Oracle Application Server instance
"im_1128.stajx14.us.oracle.com"
"See base exception for details.See base exception for details."
Resolution:
Resolve the indicated problem at the Oracle Application Server instance where
it occurred then resync the instance
java.lang.Exception: Could not delete file
C:\OracleAS\IM_1128\j2ee\OC4J_SECURITY\application-
deployments\wirelesssso\jazn-data.xml. Please check file permissions.
at oracle.security.jazn.smi.JAZNPlugin.commit(Unknown Source)
at oracle.ias.sysmgmt.repository.DcmPlugin.commit(Unknown Source)
```

Solution

If you see an error similar to "`Could not delete file jazn-data.xml`", execute the following steps:

- Stop all the OC4J processes using the following command:

  ORACLE_HOME`/opmn/bin/opmnctl stopproc ias-component=OC4J`

- Rerun the `restore_config` operation.

23.1.2 Receiving Missing Files Messages During restore_config Operation

A `restore_config` operation generates missing file messages.

Problem

During a restore_config operation, you receive messages indicating that files are missing, for example:

```
Could not copy file C:\Product\OracleAS\Devkit_1129/testdir/ to
C:\Product\OracleAS\Devkit_1129\backup_restore\cfg_bkp/2004-12-01_03-26-22.
```

Solution

During a `restore_config` operation, a temporary configuration backup is taken so that, if the restore fails, the temporary backup can be restored returning the instance to the same state as before the restore.

If some files are deleted (including files/directories specified in config_misc_files.inp) before a restore operation, then, during the temporary backup, messages are displayed indicating that certain files are missing. These error/warning messages should be ignored since the missing files are restored as part of the `restore_config` operation.

23.1.3 File-Based Repository Restoration Fails

A file-based repository restoration fails.

Problem

File-based repository restoration fails with the error indicating that the dcm daemons across the farm could not be restarted.

```
C:\fbfhost\backup_restore>bkp_restore.bat -m restore_repos -t
2004-12-07_13-49-13

C:\fbfhost\backup_restore>echo off
Stopping dcm-daemon across the farm ...
Importing file based repository ...
Restarting dcm-daemon across the farm ...
  Problem running command (Returned 150)
  c:\fbfhost/opmn/bin/opmnctl @farm restartproc ias-component=dcm-daemon
The file based repository has been restored.
But, dcm daemons across farm could not be restarted.
Please take the appropriate action.
See c:\logs/2004-12-07_13-50-18_restore_repos.log for more info
```

Solution

At this point, the file-based repository has been restored successfully. Now, perform the following steps on the repository host:

1. Stop the dcm-daemon process on the file based repository host:

   ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=dcm-daemon

2. Start the dcm-daemon processes across farm:

ORACLE_HOME/opmn/bin/opmnctl @farm startproc ias-component=dcm-daemon

23.1.4 Cannot Run a Cold Backup on Identity Management or J2EE Instance

You cannot run a cold backup on Identity Management or a J2EE instance.

Problem

When backup_cold is attempted on Identity Management or a J2EE instance, the following error message displays:

```
C:\Product\OracleAS\SSO_1203\backup_restore>bkp_restore.bat -v -m backup_cold

 C:\Product\OracleAS\SSO_1203\backup_restore>echo off
   ======================================
 Running command:
 C:\Product\OracleAS\SSO_1203/dcm/bin/dcmctl.bat whichfarm -v -script >>
 C:\Product\OracleAS\SSO_1203\backups\log_path/2004-12-09_03-56-55_whichfarm.log
 C:/Product/OracleAS/SSO_1203/backup_restore/config/config.inp: Invalid
 'database backup_path' specified
 VALUE_NOT_SET - No such file or directory
 Consider using '-f' to force creation of this path
 Failure: backup_cold failed
```

Solution

The `backup_cold` operation should be used only on the repository hosts—Metadata Repository instance or any instance hosting a file-based repository.

23.1.5 Failure Due to Loss or Corruption of OPMN.XML File

The loss or corruption of the opmn.xml file is causing a failure.

Problem

The loss or corruption of the opmn.xml file caused the following error:

```
ADMN-906025
Base Exception:
The exception, 100999, occurred at Oracle Application Server instance
"J2EE_1123.stada07.us.oracle.com"
```

Resolution

Perform the following steps to restore the opmn.xml file:

1. Run

   ```
   bkp_restore.bat -m restore_config -t <timestamp>
   ```

2. If that command fails, stop the OC4J processes.
3. Rerun

   ```
   bkp_restore.bat -m restore_config -t <timestamp>
   ```

23.1.6 A restore_config Operation Fails

A `restore_config` operation fails or the *`ORACLE_HOME`*`/j2ee/OC4J_SECURITY` directory is deleted.

Problem:

The *`ORACLE_HOME`*`/j2ee/OC4J_SECURITY` directory is accidently deleted or a `restore_config` operation fails with the following error:

```
ADMN-906025
Base Exception:
The exception, 806212, occurred at Oracle Application Server instance
"OID.stada07.us.oracle.com"
"OPMN Request: /start?mode=sync&process-type=OC4J_SECURITY

OPMN Response: HTTP/1.1 204 No Content
Content-Length: 724
Content-Type: text/html
Response: 0 of 1 processes started.
.
<?xml version='1.0' encoding='US-ASCII'?>
<response>
<opmn id="stada07:6200" http-status="204" http-response="0 of 1 processes
started.">
  <ias-instance id="OID.stada07.us.oracle.com">
    <ias-component id="OC4J">
      <process-type id="OC4J_SECURITY">
        <process-set id="default_island">
          <process id="511967353" pid="956" status="Init" index="1"
log="C:\Product\OracleAS\OID\opmn\logs\OC4J~OC4J_SECURITY~default_island~1"
.
operation="request" result="failure">
        <msg code="-21" text="failed to start a managed process after the maximum
retry limit">
```

Solution:

To resolve this problem, run the following command:

- On UNIX systems:

```
bkp_restore.sh -m restore_config -F DCM-resyncforce
```

- On Windows systems:

```
bkp_restore.bat -m restore_config -F DCM-resyncforce
```

23.1.7 Backup Operation Fails on a DCM File-Based Repository

The backup of a DCM file-based repository fails.

Problem:

The backup of a DCM file-based repository fails because of missing or corrupted files in the repository.

Solution:

If *.bom files are missing, use `restore_config` to restore the repository and then backup the repository.

For all other files, use `restore_repos` to restore the repository, and then run any of the backup options to backup the repository.

23.1.8 Timeout Occurs While Trying to Stop Processes Using opmnctl stopall

During `backup_instance_cold`, `backup_instance_cold_incr` and `restore_instance` operations, a timeout may occur while trying to stop processes using the `opmnctl stopall`.

Problem:

During some operations involving the backup or restore of a server instance, a timeout may occur while trying to stop processes using the `opmnctl stopall` command. This can occur because of heavy machine load or a process taking a long time to shut down. Under these conditions, you may receive an error message similar to the following:

```
Oracle Application Server instance backup failed.
Stopping all opmn managed processes ...

Failure : backup_instance_cold_incr failed

Unable to stop opmn managed processes !!!
```

Solution:

Running `opmnctl stopall` a second time should resolve this problem.

23.1.9 Using the Backup and Recovery Tool to Perform a Recovery Fails Due to an Unknown Log Sequence Number

When performing a recovery using the Backup and Recovery Tool, the RMAN recovery fails due to an unknown log sequence number. Use the following command to correct the problem:

```
sqlplus> alter database open resetlogs;
```

23.1.10 Enterprise Manager Cannot Access Restored Nodes on New Hosts

After using Loss of Host Automation to restore the nodes to new hosts, Enterprise Manager cannot access the nodes.

Problem

The scenario is that all nodes on a farm were lost. After using Loss of Host Automation to restore the nodes to new hosts, Enterprise Manager cannot access the nodes. The cause of this problem is that the dcmCache.xml files are not updated between restores of the individual nodes.

Solution

After restoring the first node, save a copy of dcmCache.xml from the second node. After restoring the second node, copy the saved copy of dcmCache.xml to the second node. Restart all processes on both nodes.

23.1.11 Restore of Portal Fails After Deleting OC4J Instance

A restore of a Portal instance fails after deleting an OC4J instance that was part of the backup being restored.

Problem

After a successful backup of an Infrastructure and a Portal with an OC4J instance, a restore of the Infrastructure succeeds, but the restore of the Portal fails. The OC4J instance was deleted before the restore.

Solution

Before running a restore on the Portal, run the following command:

```
dcmctl resyncInstance -force
```

23.1.12 Cold Backups Do Not Shut Down All Databases in RAC Environment

If the Oracle Application Server Metadata Repository is installed in an existing Oracle database (RepCA database), which is configured as a Real Application Cluster (RAC), then before performing a Full Cold Backup using Enterprise Manager or executing `backup_instance_cold` or `backup_cold` in command-line mode, you must shut down all the instances in the cluster database. You can use Enterprise Manager to shutdown the entire cluster database, run `srvctl stop database` to stop all the started instances or run SQL*PLUS to shut down each started instance.

23.1.13 A restore_instance Fails at restore_repos Stage

Running `restore_instance` fails when trying to restore the database (`restore_repos`).

Problem

Restoring an instance fails with the following error:

```
unable to find archive log
archive log thread=1 sequence=3
released channel: dev1
RMAN-00571: ===========================================================
RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS ===============
RMAN-00571: ===========================================================
RMAN-03002: failure of recover command at <time>
RMAN-06054: media recovery requesting unknown log: thread <> seq <> lows cn <>
```

Solution

Perform the following steps to resolve the problem:

- Complete database recovery by running the following command:

  ```
  sqlplus > alter database open resetlogs;
  ```

- Configuration recovery:

  ```
  perform opmnctl startall
  ```

- Configuration restore:

  On UNIX:

  ```
  bkp_restore.sh -m restore_config -t <timestamp>
  ```

  On Windows:

  ```
  bkp_restore.bat -m restore_config -t <timestamp>
  ```

23.1.14 Changing ORACLE_HOME May Cause Backup or Recovery Failure

Changing ORACLE_HOME from the ORACLE_HOME used to start the database may result in an error while performing backup or recovery operations.

Problem

Changing ORACLE_HOME to a different directory from the directory used to start the database may result in errors when trying to perform backup or recovery. For example, if you started the database with ORACLE_HOME set to home/foo and later try to connect to private/foo, you will not be able to connect to the original instance.

Solution

To verify where ORACLE_HOME resides, run the following command:

```
$ /usr/ucb/ps -auxeww | grep pmon
```

If the value returned for ORACLE_HOME is different from the environment ORACLE_HOME, restart the database with the ORACLE_HOME set for the environment.

23.1.15 Restore Operation Changes Farm Topology Leaving an Instance in Inconsistent State

A restore operation on one instance can change the farm topology leaving another instance on the farm in an inconsistent state.

Problem

The scenario: install core1 as a file-based repository host and take a cold backup. Install core2 and join it to core1 as a file-based repository client. Restore the file-based repository for core1. This will corrupt core2 as it was joined to core1 after the cold backup. Core2 points to core1 as the file-based repository host, but there is no record of core2 in core1 after the restore.

Resolution

Before restoring the file-based host (core1), run `dcmctl leavefarm` on core2. After restoring the repository, run `dcmctl joinfarm` on core2.

Alternatively, restore core2 with a backup taken prior to joining it to the core1 file-based repository.

23.1.16 Post-deployment Changes to Configuration Files Are Lost After Restoring DCM-Managed Components

Post-deployment changes to configuration files are lost after restoring DCM-managed component configurations.

Problem

After deploying Oracle Application Server, changes made to configuration files, such as web.xml (1 per application), are lost after the Backup and Recovery Tool restores DCM-managed component configurations.

Solution

After the restore operation completes, the web.xml files can be copied from the configuration backup using the following manual procedure:

1. Find the `config_backup_path` value from *`ORACLE_HOME`*`/backup_restore/config/config.inp` file.
2. Change the current directory to the `config_backup_path` directory:

   ```
   cd config_backup_path
   ```

3. Locate the config backup jar file containing the `web.xml` files with the changes.
4. Copy the config backup jar file to a temporary location:

   ```
   cp config_bkp_yyyy-mm-dd_hh-mm-ss.jar /tmp
   ```

5. Unjar the config backup jar file at temporary location:

   ```
   cd /tmp
   jar xvf config_bkp_yyyy-mm-dd_hh-mm-ss.jar
   ```

6. Find the `web.xml` files in config backup directory:

   ```
   cd config_bkp_yyyy-mm-dd_hh-mm-ss
   ```

   On UNIX:

   ```
   find . -name web.xml -print
   ./j2ee/home/applications/dms/WEB-INF/web.xml
   ./j2ee/home/applications/BC4J/webapp/WEB-INF/web.xml
   ./j2ee/home/default-web-app/WEB-INF/web.xml
   ```

7. Restore the web.xml files into the ORACLE_HOME:

   ```
   cp j2ee/home/applications/dms/WEB-INF/web.xml
   ORACLE_HOME/j2ee/home/applications/dms/WEB-INF/web.xml
   cp j2ee/home/applications/BC4J/webapp/WEB-INF/web.xml
   ORACLE_HOME/j2ee/home/applications/BC4J/WEB-INF/web.xml
   cp j2ee/home/default-web-app/WEB-INF/web.xml
   ORACLE_HOME/j2ee/home/default-web-app/WEB-INF/web.xml
   ```

   Alternatively, you can combine steps 6 and 7 in a script. This can be done in a UNIX shell script as follows:

   ```
   CSH> foreach (i) `find . -name web.xml -print`
   CSH> cp $i $ORACLE_HOME\$i
   CSH> end
   ```

Part VI

Appendixes and Glossary

This part contains the following appendixes:

- Appendix A, "Managing and Configuring Application Server Control"
- Appendix B, "Oracle Application Server Command-Line Tools"
- Appendix C, "URLs for Components"
- Appendix D, "Oracle Application Server Port Numbers"
- Appendix E, "Metadata Repository Schemas"
- Appendix F, "printlogs Tool Syntax and Usage"
- Appendix G, "Examples of Administrative Changes"
- Appendix H, "Supplementary Procedures for Configuring LDAP-Based Replicas"
- Appendix I, "Viewing Oracle Application Server Release Numbers"
- Appendix J, "Troubleshooting Oracle Application Server"
- "Glossary"

A

Managing and Configuring Application Server Control

When you install Oracle Application Server, the installation procedure automatically starts Oracle Enterprise Manager 10*g* Application Server Control and its related processes. You can then immediately start using the Application Server Control Console to manage the application server components.

You can also control and configure the Application Server Control. For example, you can start and stop the Application Server Control, change the Application Server Control Console password, and configure security for the Application Server Control.

This appendix covers how to manage and configure the Application Server Control. It contains the following topics:

- Starting and Stopping the Application Server Control
- Understanding Application Server Control Console Processes on UNIX
- Changing the ias_admin Password
- Configuring Security for Application Server Control Console
- Using the EM_OC4J_OPTS Environment Variable to Set Additional Application Server Control Options
- Enabling ODL for the Application Server Control Log File
- Enabling Enterprise Manager Accessibility Mode
- Managing Multiple Oracle Application Server Instances on a Single Host

A.1 Starting and Stopping the Application Server Control

To use the Oracle Enterprise Manager home pages, you must start the Application Server Control.

The Application Server Control is started automatically after you install the application server, but if you need to stop or start the Application Server Control later, refer to the following sections for more information:

- Starting and Stopping the Application Server Control Console on UNIX
- Starting and Stopping the Application Server Control Console on Windows
- Verifying That the Application Server Control Is Running

A.1.1 Starting and Stopping the Application Server Control Console on UNIX

On a UNIX system, you must start the Application Server Control manually after each system restart, or create a script to automatically start it during system start. To start or stop the Application Server Control on a UNIX system, use the `emctl` command shown in Table A–1.

The `emctl` command is available in the *`ORACLE_HOME`*`/bin` directory after you install Oracle Application Server.

Table A–1 Starting and Stopping the Application Server Control Console

Action	Using the emctl command
Start the Application Server Control Console in the Oracle home	`emctl start iasconsole`
Start the Application Server Control Console in both the Infrastructure and middle-tier Oracle homes[1]	NA
Stop the Application Server Control Console in the Oracle home	`emctl stop iasconsole`
Stop the Application Server Control Console in both the Infrastructure and middle-tier Oracle homes[1]	NA
Verify the status of the Application Server Control Console	`emctl status iasconsole`

[1] You must run the command from the middle-tier Oracle home and both must be installed on the same host.

A.1.2 Starting and Stopping the Application Server Control Console on Windows

To start or stop the Application Server Control on Windows systems, use one of the following methods:

- From the Windows **Start** menu, navigate to the Oracle Enterprise Manager menu item for the Oracle home and select **Start AS Console** or **Stop AS Console**.

  For example, to start the Application Server Control on Windows 2000, select **Start** > **Programs** > **Oracle** - *Oracle_Home Enterprise Manager* > **Start AS Console**.

- From the Windows Services control panel:

  1. Open the Services control panel.

     For example, on Windows 2000, select **Start > Settings > Control Panel > Administrative Tools** and then double-click the Services icon.

  2. Locate the Application Server Control in the list of services.

     The name of the service is consists of "Oracle," followed by the name of the home directory you specified during the installation, followed by the word "ASControl." For example, if you specified `AS10g` as the Oracle home, the Service name would be:

     ```
     OracleAS10gASControl
     ```

  3. After you locate the service, you can use the Services control panel to start or stop the Application Server Control service.

     By default, the Application Server Control service is configured to start automatically when the system starts.

A.1.3 Verifying That the Application Server Control Is Running

You can verify the Application Server Control is started by pointing your browser to the Application Server Control Console URL:

```
http://hostname.domain:port
```

For example:

```
http://hostname.domain:1156
```

There are two ways to locate the Application Server Control Console port number:

- Review the contents of the `portlist.ini` file, which is located in the following directory in the Oracle Application Server Oracle home:

  ```
  (UNIX) ORACLE_HOME/install/portlist.ini
  (Windows) ORACLE_HOME\install\portlist.ini
  ```

- Enter the following command:

  ```
  (UNIX) ORACLE_HOME/bin/emctl status iasconsole
  (Windows) ORACLE_HOME\bin\emctl status iasconsole
  ```

  See Also: Section 2.3.1, "Displaying the Application Server Control Console"

A.2 Understanding Application Server Control Console Processes on UNIX

When you start the Application Server Control, Enterprise Manager starts three distinct processes on your UNIX system. To identify these processes, you can do the following:

1. Locate and view the contents of the following file in the application server home directory:

   ```
   ORACLE_HOME/bin/emctl.pid
   ```

   This file contains the process ID for the Application Server Control. For example:

   ```
   cat emctl.pid
   5874
   ```

2. Use the following operating system command to list information about the process, including the parent process ID:

   ```
   ps -ef | grep process_id_from_the_emctl.pid_file
   ```

   For example:

   ```
   ps -ef | grep 5874
   pjones 5874 7983 0 14:40:44 pts/13 1:08/
   disk03/oracle/app1/jdk/bin/java -Xmx256m
   -DORACLE_HOME=/disk03/oracle/appserver
   ```

3. Note the number that appears immediately after the process ID; this is the process ID for the Application Server Control parent process.

4. Use the following operating system command to list all the processes associated with the Application Server Control Console:

   ```
   ps -ef | grep parent_process_id
   ```

Sample output from this command is shown in Example A–1. Descriptions of each process shown in the example are provided in Table A–2.

Example A–1 Viewing Application Server Control Console Processes

```
ps -ef | grep 7983
pjones  5873  7983 0 14:40:44 pts/10 14:42 /disk03/oracle/app1/bin/emagent
pjones  7983    1  0 14:40:41 pts/10  0:27 /disk03/oracle/app1/perl/bin/perl
pjones  5874  7983 0 14:40:44 pts/10  2:05 /disk03/oracle/app1/jdk/bin/java
-Xmx256m -DORACLE_HOME=/private/90
```

Table A–2 Summary of Application Server Control Console Processes

Process	Description
emagent	This is the first process shown in Example A–1. This process is for the Oracle Management Agent, which is a local version of the Management Agent designed specifically for monitoring Oracle Application Server components.
perl	This is the second process shown in Example A–1. This process is for the Management Watchdog Process, which monitors the Management Agent and the Application Server Control to make sure both processes are running and available at all times.
java	This is the third process shown in Example A–1. This process is for the Application Server Control itself.

A.3 Changing the ias_admin Password

The `ias_admin` password is required to use the Application Server Control Console. The following sections describe how you can change the `ias_admin` user password:

- Changing the Password Using the Application Server Control Console
- Changing the Password Using the emctl Command-Line Tool

Caution: If you use Infrastructure Services, you must adhere to the Oracle Internet Directory password policy when setting the `ias_admin` password. This is because, even though the `ias_admin` password is not stored in Oracle Internet Directory, it may be used to set component passwords within Oracle Internet Directory. The default password policy is a minimum of five characters, with at least one numeric character.

For more information, see the *Oracle Internet Directory Administrator's Guide*.

A.3.1 Changing the Password Using the Application Server Control Console

To change the `ias_admin` user password using the Application Server Control Console:

1. Navigate to the Application Server home page and select **Preferences** in the top right corner of the page.

   Application Server Control Console displays the Change Password page.

2. Enter the current `ias_admin` password, the new password, and the new password again for confirmation.

The new password must be between 5 and 30 characters, it must begin with an alphabetic character, and it must contain at least one number.

3. Click **OK** to reset the `ias_admin` password for the current application server instance.

   The next time you log in, you must use the new password.

A.3.2 Changing the Password Using the emctl Command-Line Tool

To change the `ias_admin` user password using a command-line tool:

1. Enter the following command in the Oracle home of your Oracle Application Server installation:

   ```
   (UNIX) ORACLE_HOME/bin/emctl set password old_password new_password
   (Windows) ORACLE_HOME\bin\emctl set password old_password new_password
   ```

   For example:

   ```
   (UNIX) ORACLE_HOME/bin/emctl set password m5b8r5 b8s0d9
   (Windows) ORACLE_HOME\bin\emctl set password m5b8r5 b8s0d9
   ```

2. Restart the Application Server Control.

See Also: Section A.1, "Starting and Stopping the Application Server Control"

A.4 Configuring Security for Application Server Control Console

The Application Server Control Console relies on several underlying technologies, including a version of the Management Agent that is designed to provide monitoring data to the Application Server Control Console.

By default, you access the Application Server Control Console through your Web browser using the non-secure, HTTP protocol. In addition, communications between the local Management Agent and the Application Server Control Console are transferred over insecure connections.

To secure the communications between the Management Agent and the Application Server Control Console, and to provide HTTPS browser access to the Application Server Control Console, Enterprise Manager provides the `emctl secure iasconsole` command-line utility.

The `emctl secure iasconsole` utility enables HTTPS and public key infrastructure (PKI) components, including signed digital certificates, for communications between the Application Server Control Console and the local Management Agent.

See Also: *Oracle Application Server Security Guide*

To configure security for the Application Server Control Console:

1. Stop the Application Server Control Console by entering the following command:

   ```
   (UNIX) ORACLE_HOME/bin/emctl stop iasconsole
   (Windows) net stop SERVICE_NAME
   ```

See Also: Section A.1, "Starting and Stopping the Application Server Control"

2. Enter the following command:

   ```
   (UNIX) ORACLE_HOME/bin/emctl secure iasconsole
   (Windows) ORACLE_HOME\bin\emctl secure iasconsole
   ```

   Enterprise Manager secures the Application Server Control Console. Sample output of the `emctl secure iasconsole` command is shown in Example A–2.

3. Start the Application Server Control Console by entering the following command:

   ```
   (UNIX) ORACLE_HOME/bin/emctl start iasconsole
   (Windows) net start SERVICE_NAME
   ```

4. Test the security of the Application Server Control Console by entering the following URL in your Web browser:

   ```
   https://hostname.domain:port/
   ```

   For example:

   ```
   https://mgmthost1.myco:1156/
   ```

5. If you are using OracleAS Portal, update the Portal Service Monitoring link in OracleAS Portal so you can continue to access the Application Server Control Console directly from OracleAS Portal.

   See Also: "Updating Oracle Enterprise Manager Link in OracleAS Portal" in the *Oracle Application Server Portal Configuration Guide*

Example A–2 Sample Output from the emctl secure iasconsole Command

```
./emctl stop iasconsole
Oracle Enterprise Manager 10g Application Server Control Release 10.1.2.0.2
Copyright (c) 1996, 2005 Oracle Corporation.  All rights reserved.
http://appserv1.acme.com:1811/emd/console/aboutApplication
Stopping Oracle Enterprise Manager 10g Application Server Control ...  ...
Stopped.

./emctl secure iasconsole
Oracle Enterprise Manager 10g Application Server Control Release 10.1.2.0.2
Copyright (c) 1996, 2005 Oracle Corporation.  All rights reserved.
http://appserv1.acme.com:1811/emd/console/aboutApplication
Generating Standalone Console Root Key (this takes a minute)...   Done.
Fetching Standalone Console Root Certificate...   Done.
Generating Standalone Console Agent Key...   Done.
Storing Standalone Console Agent Key...   Done.
Generating Oracle Wallet for the Standalone Console Agent...   Done.
Configuring Agent for HTTPS...   Done.
EMD_URL set in /dsk02/oracle/appserv1/sysman/config/emd.properties
Generating Standalone Console Java Keystore...   Done.
Configuring the website ...   Done.
Updating targets.xml ...    Done.
```

A.5 Using the EM_OC4J_OPTS Environment Variable to Set Additional Application Server Control Options

The following topics describe how you can use the EM_OC4J_OPTS environment variable to configure some additional Application Server Control options:

- Summary of Options You Can Set with the EM_OC4J_OPTS Environment Variable

A.5.1 Summary of Options You Can Set with the EM_OC4J_OPTS Environment Variable

You can use the EM_OC4J_OPTS environment variable to set the following options, which can affect the performance of the Application Server Control Console pages:

- By default, Application Server Control retrieves metric data as it is requested. In other words, each time you display a page that contains performance metrics, Application Server Control retrieves the data in real time by running a series of tasks that often involve connections to other software components. Depending upon the type of metric and the type of component, these operations can affect how quickly the page is displayed.

  To retrieve cached metrics (metrics which are collected and stored in memory by the Oracle Management Agent) define the environment variable EM_OC4J_OPTS to the following:

  ```
  -Doracle.sysman.refreshFlag=true
  ```

  Setting this option to TRUE can improve the performance of specific pages in the Application Server Control Console; however, the data displayed on those pages may not be as recent as it would be when this option is set to FALSE.

- By default, the Application Server Control Console displays a progress page when operations take time to complete. To disable the processing page so that the Application Server Control Console waits for start, stop, and other such actions without displaying a progress page, define the environment variable EM_OC4J_OPTS to the following before starting the Application Server Control Console:

  ```
  -Doracle.sysman.eml.util.iAS.waitForCompletion=true
  ```

- By default, when Application Server Control attempts to retrieve performance data, it waits two (2) seconds before displaying the requested page. If the data has not been retrieved within 2 seconds, some metric values do not appear on the page when it is rendered in the Web browser. To display the data after the page has been rendered, click the **Refresh Data** icon near the timestamps on the page.

  To increase the timeout for status and host-related metrics such as Memory and CPU usage on the Application Server home page, define the environment variable EM_OC4J_OPTS to the following before starting the Application Server Control Console:

  ```
  -Doracle.sysman.ias.ApplicationServerObject.timeout=true
  ```

  When this option is set to TRUE, certain Application Server Control Console pages may take longer to display, but you will experience fewer metric collection errors.

  Note that this setting affects only components that are not managed by Oracle Process Manager and Notification Server (OPMN).

  > **See Also:** *Oracle Process Manager and Notification Server Administrator's Guide* for a complete list of the Oracle Application Server components that are managed by OPMN

A.5.2 Setting the EM_OC4J_OPTS Environment Variable

On UNIX systems, set the EM_OC4J_OPTS environment variable as follows:

```
setenv EM_OC4J_OPTS "-Doracle.sysman.eml.util.iAS.waitForCompletion=true"
```

To set multiple configuration options with the EM_OC4J_OPTS variable, separate the options with a space and enclose the entire value of the variable within quotation marks. For example:

```
setenv EM_OC4J_OPTS "-Doracle.sysman.eml.util.iAS.waitForCompletion=true
-Doracle.sysman.ias.ApplicationServerObject.timeout=true"
```

On Windows systems, use the System Properties control panel to define EM_OC4J_OPTS as a system variable.

A.6 Enabling ODL for the Application Server Control Log File

By default, the log file generated for Application Server Control is saved in text format. However, you can configure Application Server Control so its log file will be saved using the Oracle Diagnostic Logging (ODL) format.

When you enable ODL for the Application Server Control log files, the logging and diagnostic information is saved in XML format and the contents of the log files are loaded automatically into the Log Repository. You can then use the Log Repository to search for diagnostic information generated by the Application Server Control.

See Also: Chapter 5, "Managing Log Files"

By default, Application Server Control logs information and errors to the following log file in the application server home directory:

```
(UNIX) ORACLE_HOME/sysman/log/emias.log
(Windows) ORACLE_HOME\sysman\log\emias.log
```

After you perform the following procedure, Application Server Control will instead log information and error messages to the following file, which formats the data according to the ODL standard:

```
(UNIX) ORACLE_HOME/sysman/log/log.xml
(Windows) ORACLE_HOME\sysman\log\log.xml
```

As soon as Application Server Control creates the `log.xml` file and you start the Log Loader, the Log Loader begins loading the logging data into the Oracle Application Server Log Repository on the Log Loader's next run.

Refer to the following sections for more information:

- Configuring the Application Server Control Logging Properties to Enable ODL
- About the Application Server Control ODL Logging Properties
- Configuring Logging Properties When ODL Is Not Enabled

A.6.1 Configuring the Application Server Control Logging Properties to Enable ODL

To configure the Application Server Control to support ODL:

1. Use a text editor to edit the following file in the Oracle Application Server home directory:

   ```
   (UNIX) ORACLE_HOME/sysman/config/emiasconsolelogging.properties
   (Windows) ORACLE_HOME\sysman\config\emiasconsolelogging.properties
   ```

2. Follow the instructions in the file to replace the default properties with those that are commented by default.

Example A–3 shows the properties in the `emiasconsolelogging.properties` file that enable ODL for the Application Server Control log file.

3. Save and close the `emiasconsolelogging.properties` file.
4. Restart Application Server Control.

Example A–3 ODL Logging Properties for the Application Server Control Console

```
# To support the ODL log appender, replace the lines above
# with the following and restart EM.  The resulting ODL log files
# will be read by the Log Loader and written to the Log Repository.
#
# log4j.appender.emiaslogAppender=oracle.core.ojdl.log4j.OracleAppender
# log4j.appender.emiaslogAppender.ComponentId=EM
# log4j.appender.emiaslogAppender.LogDirectory=/private/shiphomes/
  m21_infra/sysman/log
# log4j.appender.emiaslogAppender.MaxSize=20000000
# log4j.appender.emiaslogAppender.MaxSegmentSize=5000000
```

A.6.2 About the Application Server Control ODL Logging Properties

Table A–3 describes the Oracle Diagnostic Logging (ODL) logging properties available in the `emiasconsolelogging.properties` file.

Table A–3 ODL Properties in Application Server Control Console Logging Properties

Property	Description
log4j.appender.emiaslogAppender.LogDirectory	Determines the directory where the `log.xml` file will be saved.
log4j.appender.emiaslogAppender.MaxSize	Determines the maximum amount of disk space to be used by the `log.xml` file and the logging rollover files.
log4j.appender.emiaslogAppender.MaxSegmentSize	Determines the maximum size of the log.xml file. When the `log.xml` file reaches this size, a rollover file is created.

When you enable ODL, the resulting `log.xml` file increases in size over time as information is written to the file. The file is designed to reach a maximum size, determined by the `MaxSegmentSize` property described in Table A–3. When the file reaches the predefined maximum size, Application Server Control renames (or rolls) the logging or trace information to a new file name and starts a new log or trace file. This process keeps the log file from growing too large.

To be sure you have access to important log information, Application Server Control will rollover the `log.xml` file until the log file and its rollover files consume a predefined, maximum amount of disk space, determined by the MaxSize property shown in Example A–3. When the log file and its rollover files reach this predefined target, Application Server Control deletes the oldest rollover file.

As a result, you will often see multiple log files in the log directory. The following example shows three Application Server Control rollover files and the current log file in the log directory:

```
log.xml
log1.xml
log2.xml
log3.xml
```

A.6.3 Configuring Logging Properties When ODL Is Not Enabled

If you do not enable ODL, you can still configure the logging properties for the Application Server Control by modifying the following configuration files:

- `emiasconsolelogging.properties`

  Modify the properties in this file to configure the amount of information saved to the `emias.log` file, which contains general logging information about the Application Server Control.

- `emagentlogging.properties`

  Modify the properties in this file to configure the amount of information saved to the `emagent.log` file, which contains logging information specific to the Management Agent.

For more information about the configuration settings in these files, see "Locating and Configuring Log Files" in *Oracle Enterprise Manager Advanced Configuration*.

A.7 Enabling Enterprise Manager Accessibility Mode

The following sections provide information on the benefits of running Enterprise Manager in accessibility mode, as well as instructions for enabling accessibility mode:

- Making HTML Pages More Accessible
- Providing Textual Descriptions of Enterprise Manager Charts
- Modifying the uix-config.xml File to Enable Accessibility Mode

A.7.1 Making HTML Pages More Accessible

Enterprise Manager takes advantage of user interface development technologies that improve the responsiveness of some user operations. For example, when you navigate to a new record set in a table, Enterprise Manager does not redisplay the entire HTML page.

However, this performance-improving technology is generally not supported by screen readers. When you enable accessibility mode, you disable this feature, and as a result, make the Enterprise Manager HTML pages more accessible for disabled users.

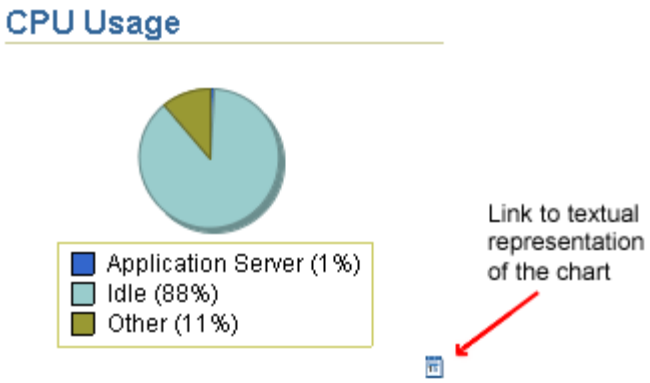
A.7.2 Providing Textual Descriptions of Enterprise Manager Charts

Throughout Enterprise Manager, charts are used to display performance data. For most users, these charts provide a valuable graphical view of the data that can reveal trends and help identify minimum and maximum values for performance metrics.

However, charts do not convey information in a manner that can be read by a screen reader. To remedy this problem, you can configure Enterprise Manager to provide a complete textual representation of each performance chart. When you enable accessibility mode, Enterprise Manager displays a small icon for each chart that can be used as a drill-down link to the textual representation.

Figure A–1 shows an example of the icon that appears below each chart after you enable accessibility mode.

Figure A–1 Icon Representing the Textual Representation of a Chart



A.7.3 Modifying the uix-config.xml File to Enable Accessibility Mode

1. Locate the `uix-config.xml` configuration file in the Oracle Application Server home directory:

   `(UNIX)` *`ORACLE_HOME`*`/sysman/webapps/emd/WEB-INF`
   `(Windows)` *`ORACLE_HOME`*`\sysman\webapps\emd\WEB-INF`

2. Open the `uix-config.xml` file using your favorite text editor and locate the following entry:

```
<!-- An alternate configuration that disables accessibility features  -->
<default-configuration>
  <accessibility-mode>inaccessible</accessibility-mode>
</default-configuration>
```

3. Change the value of the `accessibility-mode` property from `inaccessible` to `accessible`.
4. Save and close the file.
5. Restart the Application Server Control Console.

A.8 Managing Multiple Oracle Application Server Instances on a Single Host

If you have installed multiple Oracle Application Server instances on a single host, you can optionally reduce the memory and CPU consumption by performing the following postinstallation configuration procedure.

By default, each Oracle Application Server instance on the host has its own Application Server Control, which is used to manage the components of that particular Oracle Application Server instance.

Use the instructions in this document to configure a single Application Server Control to manage two Oracle Application Server instances installed on the same host.

Note: The Application Server Control that you use to manage both application server instances on the host is referred to as the **active Application Server Control**. The other Application Server Control on the host is referred to as the **inactive Application Server Control**.

This document contains the following sections:

- Restrictions and Supported Configurations
- Creating a New targets.xml for the Active Application Server Control
- Updating the StandaloneConsoleURL Property in the Inactive targets.xml File
- Updating the opmn.xml File to Refer to the Active Application Server Control
- Restarting the Active Application Server Control and Verifying the Results
- Deinstallation Procedures

A.8.1 Restrictions and Supported Configurations

The following sections describe the restrictions and supported configurations for performing the procedure described in this document:

- General Restrictions
- Supported Installation Types
- Support for Separately Installed Components

A.8.1.1 General Restrictions

Before you begin configuring a single Application Server Control to manage multiple application server instances on a host, consider the following restrictions:

- This procedure allows you to manage multiple Oracle Application Server 10*g* Release 2 (10.1.2) instances that were installed on the same host using the Oracle Universal Installer. This procedure does not support previous versions of Oracle Application Server or Oracle Application Server instances installed without the use of the Oracle Universal Installer.
- This procedure can be performed only if all the application servers running on the host have been installed using the same operating system user account.
- You can perform this procedure only with application server instances of the same version. For example, you can perform this procedure with two Oracle Application Server 10*g* Release 2 (10.1.2.0.0) instances or with two Oracle Application Server 10*g* Release 2 (10.1.2.0.2) instances. You cannot, however, perform this procedure with one 10.1.2.0.0 instance and one 10.1.2.0.2 instance.

  Note that the Oracle Application Server version number is always available on the Application Server Home page in the Application Server Control Console.

A.8.1.2 Supported Installation Types

The procedures described in this document are supported only for specific Oracle Application Server configurations. Specifically, you can perform this procedure to support two Oracle Application Server instances on the same host.

This procedure can be performed in both secure (HTTPS) and nonsecure (HTTP) installations.

Table A–4 shows the configurations that are supported, as well as notes about each installation type combination.

Table A–4 Supported Configurations for Managing Multiple Application Server Instances with a Single Application Server Control

Instance 1	Instance 2	Notes
J2EE and Web Cache	J2EE and Web Cache	This configuration is supported only if you are not using Oracle Application Development Framework. Oracle ADF is not supported when you use one Application Server Control to manage two J2EE and Web Cache instances. The active Application Server Control can reside in either home.
J2EE and Web Cache	OracleAS Infrastructure	The active Application Server Control must reside in the J2EE & Web Cache Oracle home.
Portal and Wireless	OracleAS Infrastructure	The active Application Server Control must reside in the Portal & Wireless Oracle home.
Business Intelligence & Forms	OracleAS Infrastructure	The active Application Server Control must reside in the Business Intelligence and Forms Oracle home.

A.8.1.3 Support for Separately Installed Components

When performing the procedure described in this document, you should consider whether or not you have installed additional software on top of your standard Oracle Application Server installations. Specifically:

- If you have installed Oracle Content Management SDK (Oracle CM SDK), then the Oracle CM SDK software must be installed in the Oracle home for the active Application Server Control.

  In addition, when performing this procedure, Oracle CM SDK is supported only when it is installed in a J2EE and Web Cache Oracle home or in a Portal and Wireless Oracle home.

- If you have installed Oracle Workflow, then the Oracle Workflow software must be installed in the Oracle home for the active Application Server Control.

  In addition, when performing this procedure, Oracle Workflow is supported only when it is installed in a Portal and Wireless Oracle home.

A.8.2 Creating a New targets.xml for the Active Application Server Control

By default, each Oracle Application Server instance has its own Application Server Control Console URL and its own list of targets to manage. The targets managed by each Application Server Control are defined in the following configuration file in the home directory of the Oracle Application Server instance:

```
(UNIX) $ORACLE_HOME/sysman/emd/targets.xml
(Windows) %ORACLE_HOME%\sysman\emd\targets.xml
```

The first step in managing multiple application server instances from a single Application Server Control is to combine the `targets.xml` files for each instance into one.

To create a new `targets.xml` that includes the application server targets for multiple application server instances, use the following procedure:

1. Stop the Application Server Control for each of the application server instances on the host.

On UNIX systems, enter the following command in each Oracle Application Server Oracle home to stop the Application Server Control:

```
ORACLE_HOME/bin/emctl stop iasconsole
```

On Windows systems, use the Services control panel to stop the Application Server Control service for each Oracle home.

2. If it is not set already, set the ORACLE_HOME environment variable so it represents the complete path to the home directory of the active Application Server Control, which you will use to manage multiple application server instances on the host.

   For example:

   ```
   (UNIX) setenv ORACLE_HOME /dev01/oracle/oas1/
   (Windows) set ORACLE_HOME=D:\oracle\oas1\
   ```

3. Set the JAVA_HOME environment variable so it represents the location of your Java executable.

   For example:

   ```
   (UNIX) setenv JAVA_HOME $ORACLE_HOME/jdk
   (Windows)  set JAVA_HOME=%ORACLE_HOME%\jdk
   ```

4. Change directory to the following location in the home directory of the active Application Server Control:

   ```
   (UNIX) $ORACLE_HOME/sysman/emd/
   (Windows) %ORACLE_HOME%\sysman\emd\
   ```

5. Copy the `targets.xml` file so you have a backup copy that will not be overwritten:

   ```
   (UNIX) cp targets.xml old_targets.xml
   (Windows) copy targets.xml old_targets.xml
   ```

6. Enter the following command at the command prompt:

   ```
   (UNIX) $JAVA_HOME/bin/java -jar $ORACLE_HOME/jlib/emConfigInstall.jar
    listtargetsfully source_home active_home > targets_temp.xml

   (Windows) %JAVA_HOME%\bin\java -jar %ORACLE_HOME%\jlib\emConfigInstall.jar
    listtargetsfully source_home active_home > targets_temp.xml
   ```

   Replace *source_home* with the full path to the Oracle home of the application server instance that will be managed by the active Application Server Control.

   Replace *active_home* with the full path to the Oracle home for the active Application Server Control.

   For example:

   ```
   (UNIX) $JAVA_HOME/bin/java -jar $ORACLE_HOME/jlib/emConfigInstall.jar
    listtargetsfully /dev0/oracle/oas2/ $ORACLE_HOME > targets_temp.xml

   (Windows) %JAVA_HOME%\bin\java -jar %ORACLE_HOME%\jlib\emConfigInstall.jar
    listtargetsfully C:\oracle\oas2 %ORACLE_HOME% > targets_temp.xml
   ```

   This command lists the contents of the `targets.xml` file in the source Oracle home and redirects the output to a temporary version of the `targets.xml` file. This temporary version of the file (`targets_temp.xml`) contains the contents of

the targets.xml files in the source Oracle home. The temporary file also includes any encrypted information (such as target monitoring credentials) in a format that can be read successfully by the active Application Server Control.

Note: It is important that you use the emConfigInstall.jar administrative tool to create the targets_temp.xml temporary file. Do not attempt to copy and paste the contents of the inactive targets.xml into the targets.xml of the active Oracle home; otherwise, encrypted information in the targets.xml file will be lost.

---

7. Using a text editor, open the targets_temp.xml file and copy all of the content, except the following:
   - The <Targets> and </Targets> tags at the beginning and at the end of the file
   - The target definition for the host target. For example:

     ```
     <Target TYPE="host" NAME="sys1.acme.com"
     DISPLAY_NAME="sys1.acme.com" VERSION="1.0">
     </Target>
     ```

8. Paste the content you copied from the targets_temp.xml file into the targets.xml file in the Oracle home of the active Application Server Control.

   Be sure to paste the content at the end of the file, but before the </Targets> tag. In other words, be sure that all the target definitions are within the <Targets> and </Targets> tags.

9. Locate the following entry for each oracle_ias target type in the new targets.xml file:

   ```
   <Property NAME="StandaloneConsoleURL"
   VALUE="http://node_name:port/emd/console"/>
   ```

   This entry identifies the URL and port number of the Application Server Control Console used to manage the application server instance.

10. Make sure the port number for each of these entries matches the port number of the active Application Server Control Console, which you will be using to manage multiple application server targets on this host.

11. Save and close the updated targets.xml file.

A.8.3 Updating the StandaloneConsoleURL Property in the Inactive targets.xml File

By default, when you install multiple instances of Oracle Application Server on a host, each instance is assigned a unique port number from which you can view the Application Server Control Console for that instance. However, when you configure your system to use only one Application Server Control for multiple instances, all the instances on the host must reference the same URL and port number.

As a result, you must make sure that other software components on the system also use only the port for the active Application Server Control.

Specifically, you must make sure that the Distributed Configuration Management (DCM) software associated with the inactive Application Server Control is updated to recognize the port number of the active Application Server Control.

You accomplish this task by updating the original `targets.xml` in the Oracle home directory of the inactive Application Server Control so that the `StandaloneConsoleURL` property refers to the active Application Server Control URL and port number.

> **Note:** In the following procedure, the term *INACTIVE_ORACLE_HOME* refers to the Oracle home that will not be running an Application Server Control. The inactive Oracle home will be managed by the active Application Server Control.

Perform the following task in the Oracle home of the inactive Application Server Control:

1. If you have not done so already, stop the inactive Application Server Control.

   On UNIX systems, enter the following command:

   ```
   INACTIVE_ORACLE_HOME/bin/emctl stop iasconsole
   ```

   On Windows systems, use the Services control panel to stop the Application Server Control service for the inactive Oracle home.

2. Use a text editor to open the `targets.xml` file, which is located in the following directory of the inactive Oracle home:

   ```
   (UNIX) INACTIVE_ORACLE_HOME/sysman/emd/
   (Windows) INACTIVE_ORACLE_HOME\sysman\emd\
   ```

3. Locate the `StandaloneConsoleURL` property for the `oracle_ias` target:

   ```
   <Property NAME="StandaloneConsoleURL"
   VALUE="http://node_name:port/emd/console"/>
   ```

   This entry identifies the URL and port number of the Application Server Control Console used to manage this application server instance.

4. Make sure the port number for this entry matches the port number of the active Application Server Control Console, which you will be using to manage this application server target.

5. Save and close the updated `targets.xml` file.

6. Start and then stop the inactive Application Server Control Console.

   On UNIX systems, use the following commands to start and then stop the Application Server Control:

   ```
   INACTIVE_ORACLE_HOME/bin/emctl start iasconsole
   INACTIVE_ORACLE_HOME/bin/emctl stop iasconsole
   ```

   On Windows systems, use the Services control panel to start and then stop the inactive Application Server Control service.

   When you restart the Application Server Control Console, DCM is initialized and registers the new port value.

   Now that you have configured the active Application Server Control, you no longer need to run this instance of the Application Server Control.

A.8.4 Updating the opmn.xml File to Refer to the Active Application Server Control

The Oracle Process Management and Notification (OPMN) software within each Oracle Application Server Oracle home is configured to work with the local Application Server Control for that instance.

When you configure your system to use only one Application Server Control, you must edit the OPMN configuration file in the Oracle home of the inactive Application Server Control. Specifically, you must configure OPMN so that it references the active Application Server Control and not the local, inactive Application Server Control.

> **Note:** In the following procedure, the term *INACTIVE_ORACLE_HOME* refers to the Oracle home that will not be running an Application Server Control. The inactive Oracle homes will be managed by the active Application Server Control.

Perform this procedure in the Oracle home of the inactive Application Server Control:

1. Stop OPMN.

   On UNIX systems, enter the following command in the Oracle home of the inactive Application Server Control:

   ```
   INACTIVE_ORACLE_HOME/opmn/bin/opmnctl stopall
   ```

   On Windows systems, use the Services control panel to stop the process manager service for the inactive Oracle home.

2. Use a text editor to open the `opmn.xml` file for the instance:

   ```
   (UNIX) INACTIVE_ORACLE_HOME/opmn/conf/opmn.xml
   (Windows) INACTIVE_ORACLE_HOME\opmn\conf\opmn.xml
   ```

3. Locate the following entry in the file:

   ```
   <ias-component id="dcm-daemon" status="enabled" id-matching="true">
             .
             .
             .
          <data id="java-parameters" value="-Xmx256m -Xrs
              -Doracle.ias.sysmgmt.logging.loglevel=ERROR
              -Djava.net.preferIPv4Stack=true -Djava.io.tmpdir=$TMP"/>
             .
             .
             .
   </ias-component>
   ```

4. Add the following to the existing `java-parameters` data tag:

   ```
   -DemLocOverride=oracle_home_of_the_active_application_server_control
   ```

   For example:

   ```
   <data id="java-parameters" value="-Xmx256m -Xrs
         -Doracle.ias.sysmgmt.logging.loglevel=ERROR
         -Djava.net.preferIPv4Stack=true -Djava.io.tmpdir=$TMP
         -DemLocOverride=/dev0/oracle/oas1"/>
   ```

   Note that the additional Java parameter must be inserted before the closing quotation mark.

5. Save and close the `opmn.xml` file.
6. Start OPMN.

   On UNIX systems, enter the following command:

   ```
   INACTIVE_ORACLE_HOME/opmn/bin/opmnctl startall
   ```

   On Windows systems, use the Services control panel to start the process manager service for the inactive Oracle home.

A.8.5 Restarting the Active Application Server Control and Verifying the Results

The procedure you use to verify the results of this procedure varies depending upon your configuration. See the following topics for more information:

- Verifying the Procedure for Infrastructure Installations
- Verifying the Procedure for Middle-Tier Installations

A.8.5.1 Verifying the Procedure for Infrastructure Installations

Use the following verification procedure if one of the application server instances you are managing with a single Application Server Control is an OracleAS Infrastructure installation that includes Oracle Identity Management:

1. Navigate to the home directory of the Identity Management installation and start Oracle Internet Directory by entering the following command:

   ```
   (UNIX) INACTIVE_ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=oid
   (Windows) INACTIVE_ORACLE_HOME\opmn\bin\opmnctl startproc ias-component=oid
   ```

2. Start the active Application Server Control.

   On UNIX systems, use the following command:

   ```
   $ORACLE_HOME/bin/emctl start iasconsole
   ```

   On Windows systems, use the Services control panel to start the Application Server Control service.

   There is no need to start the Application Server Control in the other, inactive Oracle Application Server home directory.

3. Open your browser and enter the host name and port for the active Application Server Control Console.

   Enterprise Manager displays the Farm page, which lists the application server instances that are currently using this OracleAS Metadata Repository.

   When you click an application server name on the Farm page, notice that you are navigating to the same port number in all cases. You have access to all the management features, but only one Application Server Control is running on the host.

A.8.5.2 Verifying the Procedure for Middle-Tier Installations

If you are managing two J2EE and Web Cache instances, use the following procedure to verify that you have configured the active Application Server Control successfully:

1. Start the active Application Server Control.

   On UNIX systems, use the following command:

   ```
   $ORACLE_HOME/bin/emctl start iasconsole
   ```

On Windows systems, use the Services control panel to start the Application Server Control service.

There is no need to start the Application Server Control in the other, inactive Oracle Application Server home directory.

2. Open your browser and enter the host name and port for the active Application Server Control Console.

   The result depends upon whether or not the instances you are managing are part of an Oracle Application Server Farm:

   - If the instances are not part of an OracleAS Farm, Enterprise Manager displays the Application Servers page, which lists the application servers on the host. Click the name of an application server to navigate to the Application Server home page for that instance.
   - If the instances are part of an OracleAS Farm, Enterprise Manager displays the OracleAS Farm page, which lists all the application server standalone instances and OracleAS Clusters that use this Farm repository.

   In either situation, when you click the name of an application server instance, notice that you are navigating to the same port number in all cases. You have access to all the management features, but only one Application Server Control is running on the host.

   Further, note that when you click the name of the inactive application server, no start time appears in the **Start Time** column for the Management component listed in the System Components table. This is because the inactive Application Server Control is not running and has not been started.

A.8.6 Deinstallation Procedures

After you have performed the procedures in this document, you may want to deinstall one of the application server instances on the host.

The following sections provide instructions you should follow before you remove either the active or the inactive application server instance:

- Deinstalling the Oracle Home with the Active Application Server Control
- Deinstalling the Oracle Home with the Inactive Application Server Control

> **See Also:** Oracle Application Server Installation Guide for instructions on deinstalling Oracle Application Server instances

A.8.6.1 Deinstalling the Oracle Home with the Active Application Server Control

If you deinstall the Oracle home with the active Application Server Control, use the following instructions to configure the inactive Application Server Control so it can once again manage the remaining Oracle Application Server instance:

1. Stop OPMN in the Oracle home of the inactive Application Server Control.

   On UNIX systems, enter the following command in the Oracle home of the inactive Application Server Control:

   ```
   INACTIVE_ORACLE_HOME/opmn/bin/opmnctl stopall
   ```

   On Windows systems, use the Services control panel to stop the process manager service for the inactive Oracle home.

2. Remove the `-DemLocOverride` entry from the `opmn.xml` file.

   Refer to the instructions in the section Section A.8.4, "Updating the opmn.xml File to Refer to the Active Application Server Control" to locate the entry in the `opmn.xml` file.

3. Start OPMN.

   On UNIX systems, enter the following command:

   ```
   INACTIVE_ORACLE_HOME/opmn/bin/opmnctl startall
   ```

   On Windows systems, use the Services control panel to start the process manager service for the inactive Oracle home.

4. Modify the `StandAloneConsoleURL` property in the `targets.xml` file so it refers to the original port number of the Application Server Control Console in the inactive Oracle home.

   Refer to the instructions in the section Section A.8.3, "Updating the StandaloneConsoleURL Property in the Inactive targets.xml File" to locate the entry in the `targets.xml` file.

   To identify the original port number for the `StandaloneConsoleURL` property, check the contents of the following file, which contains the original port numbers assigned during the installation procedure:

   ```
   (UNIX) INACTIVE_ORACLE_HOME/install/setupinfo.txt
   (Windows) INACTIVE_ORACLE_HOME\install\setupinfo.txt
   ```

5. Start the inactive Application Server Control.

   On UNIX systems, use the following commands to start the Application Server Control:

   ```
   ORACLE_HOME/bin/emctl start iasconsole
   ```

   On Windows systems, use the Services control panel to start the Application Server Control service.

   When you start the Application Server Control, DCM is initialized and registers the new port value in the `opmn.xml` file.

After you have reconfigured the inactive Application Server Control, you can manage the remaining instance with its original Application Server Control.

A.8.6.2 Deinstalling the Oracle Home with the Inactive Application Server Control

If you deinstall the Oracle home with the inactive Application Server Control, you must modify the `targets.xml` in the Oracle home of the active Application Server Control so it no longer references components that were in the inactive Oracle Application Server instance.

To remove the references to components in the Oracle home of the inactive Application Server Control:

1. If you have not done so already, stop the active Application Server Control.

   On UNIX systems, enter the following command:

   ```
   ACTIVE_ORACLE_HOME/bin/emctl stop iasconsole
   ```

   On Windows systems, use the Services control panel to stop the Application Server Control service for the active Oracle home.

2. Use a text editor to open the `targets.xml` file, which is located in the following directory of the active Oracle home:

   ```
   (UNIX) ACTIVE_ORACLE_HOME/sysman/emd/
   (Windows)(Windows) ACTIVE_ORACLE_HOME\sysman\emd\
   ```

3. Remove all the target definitions that refer to components in the inactive Oracle home that you will be deinstalling.

   One way to identify the targets that represent components in the inactive Oracle home is by checking the `OracleHome` property. Each target (except the Host target) should have an `OracleHome` property, which indicates the location of the component. For example:

   ```
   <Property NAME="OracleHome" VALUE="D:\oracle\appserv42"/>
   ```

   As a result, you can remove the targets that include an `OracleHome` property that references the Oracle home of the inactive Application Server Control.

   For each component in the inactive Oracle home, be sure to remove the `<target>` and `</target>` tags, as well as all the XML tags between the `<target>` and `</target>` tags.

4. After you remove all the target tags that refer to the inactive Oracle home, save the updated `targets.xml` file and start the active Application Server Control.

   On UNIX systems, use the following command to start the Application Server Control:

   ```
   ACTIVE_ORACLE_HOME/bin/emctl start iasconsole
   ```

   On Windows systems, use the Services control panel to start the Application Server Control service.

After you reconfigure and restart the active Application Server Control, you can manage the remaining active instance with its original Application Server Control.

B

Oracle Application Server Command-Line Tools

Table B–1 summarizes the command-line tools available in Oracle Application Server, with descriptions and pointers to more information.

Table B–1 Oracle Application Server Command-Line Tools

Command	Path from Oracle Home	Description
bulkdelete	UNIX: ldap/bin/bulkdelete.sh Windows: ldap\bin\bulkdelete.bat	Delete a subtree efficiently in Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
bulkload	UNIX: ldap/bin/bulkload.sh Windows: ldap\bin\bulkload.bat	Create Oracle Internet Directory entries from data residing in or created by other applications. **See:** *Oracle Identity Management User Reference*
bulkmodify	UNIX: bin/bulkmodify Windows: bin\bulkmodify	Modify a large number of existing Oracle Internet Directory entries in an efficient way. **See:** *Oracle Identity Management User Reference*
catalog	UNIX: ldap/bin/catalog.sh Windows: ldap\bin\catalog.bat	Add and delete catalog entries in Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
dcmctl	UNIX: dcm/bin/dcmctl Windows: dcm\bin\dcmctl.bat	Manage application server instances and OracleAS Clusters, deploy applications, manage the DCM repository. **See:** *Distributed Configuration Management Administrator's Guide*
dipassistant	UNIX: bin/dipassistant Windows: bin\dipassistant.bat	Directory Integration and Provisioning Assistant—assists in performing all operations in the Oracle Directory Integration and Provisioning platform. **See:** *Oracle Identity Management User Reference*
dmstool	UNIX: bin/dmstool Windows: bin\dmstool.bat	View performance metrics and set reporting intervals. **See:** *Oracle Application Server Performance Guide*
emctl	UNIX: bin/emctl Windows: bin\emctl.bat	Start, stop, and manage security for Oracle Enterprise Manager 10*g*. **See:** Chapter 2, "Introduction to Administration Tools"
eulbuilder.jar	UNIX: bin/eulbuilder.jar Windows: bin\eulbuilder.jar	Discoverer EUL Java command-line interface. Create and manipulate Discoverer EULs without installing Oracle Discoverer Administrator. **See:** *Oracle Business Intelligence Discoverer EUL Command Line for Java User's Guide*

Table B–1 (Cont.) Oracle Application Server Command-Line Tools

Command	Path from Oracle Home	Description
fplsqlconv90	UNIX: bin/fplsqlconv90 Windows: bin\fplsqlconv90	Update obsolete usage in your PL/SQL code in order to migrate your Forms6*i* applications to Oracle Application Server Forms Services. **See:** Oracle Application Server Forms Services Online Help
hiqpurge	UNIX: ldap/bin/hiqpurge.sh Windows: ldap\bin\hiqpurge.bat	Move the changes from the human intervention queue to the purge queue. **See:** *Oracle Identity Management User Reference*
hiqretry	UNIX: ldap/bin/hiqretry.sh Windows: ldap\bin\hiqretry.bat	Move the changes from the human intervention queue to the retry queue. **See:** *Oracle Identity Management User Reference*
iasua	UNIX: upgrade/iasua.sh Windows: upgrade\iasua.bat	Oracle Application Server Upgrade Assistant. **See:** *Oracle Application Server Upgrade and Compatibility Guide*
ifbld90	UNIX: bin/ifbld90 Windows: bin\ifbld90	Start Forms Developer with specific options for a Forms session. **See:** Oracle Application Server Forms Services Online Help
ifcmp90	UNIX: bin/ifcmp90 Windows: bin\ifcmp90	Start Form Compiler to generate a form. **See:** Oracle Application Server Forms Services Online Help
iff2xml90	UNIX: bin/iff2xml90 Windows: bin\iff2xml90	Traverse a module object hierarchy and produce an XML representation of it. **See:** Oracle Application Server Forms Services Online Help
ifweb90	UNIX: bin/ifweb90 Windows: bin\ifweb90	Preview a form in a Web browser. **See:** Oracle Application Server Forms Services Online Help
ifxml2f90	UNIX: bin/ifxml2f90 Windows: bin\ifxml2f90	Take well-defined XML format and convert it back into a module. **See:** Oracle Application Server Forms Services Online Help
ifxmlv90	UNIX: bin/ifxmlv90 Windows: bin\ifxmlv90	XML Validator that can be used on the command line or called from Java to validate .xml files or XMLDocument Java objects respectively against the Forms XML Schema. **See:** Oracle Application Server Forms Services Online Help
jazn.jar	UNIX: j2ee/home/jazn.jar Windows: j2ee\home\jazn.jar	Manage both XML-based and LDAP-based JAAS data. **See:** *Oracle Application Server Containers for J2EE Security Guide*
ldapadd	UNIX: bin/ldapadd Windows: bin\ldapadd	Add entries, their object classes, attributes, and values to Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
ldapaddmt	UNIX: bin/ldapaddmt Windows: bin\ldapaddmt	Add entries, their object classes, attributes, and values to Oracle Internet Directory. Like `ldapadd`, except supports multiple threads for adding entries concurrently. **See:** *Oracle Identity Management User Reference*
ldapbind	UNIX: bin/ldapbind Windows: bin\ldapbind	Determine if you can authenticate a client to a server. **See:** *Oracle Identity Management User Reference*

Table B–1 (Cont.) Oracle Application Server Command-Line Tools

Command	Path from Oracle Home	Description
ldapcompare	UNIX: bin/ldapcompare Windows: bin\ldapcompare	Match attribute values you specify in the command-line with the attribute values in the Oracle Internet Directory entry. **See:** *Oracle Identity Management User Reference*
ldapdelete	UNIX: bin/ldapdelete Windows: bin\ldapdelete	Remove entire entries from Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
ldapmoddn	UNIX: bin/ldapmoddn Windows: bin\ldapmoddn	Modify the DN or RDN of an Oracle Internet Directory entry. **See:** *Oracle Identity Management User Reference*
ldapmodify	UNIX: bin/ldapmodify Windows: bin\ldapmodify	Perform actions on attributes in Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
ldapmodifymt	UNIX: bin/ldapmodifymt Windows: bin\ldapmodifymt	Modify several Oracle Internet Directory entries concurrently. **See:** *Oracle Identity Management User Reference*
ldapsearch	UNIX: bin/ldapsearch Windows: bin\ldapsearch	Search and retrieve specific entries in Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
ldifmigrator	UNIX: bin/ldifmigrator Windows: bin\ldifmigrator.bat	Migrate data from application-specific repositories into Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
ldifwrite	UNIX: bin/ldifwrite Windows: bin\ldifwrite	Convert to LDIF all or part of the information residing in an Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
ocactl	UNIX: oca/bin/ocactl Windows: oca\bin\ocactl.bat	OracleAS Certificate Authority administration tool. **See:** *Oracle Application Server Certificate Authority Administrator's Guide*
oidctl	UNIX: bin/oidctl Windows: bin\oidctl	Start and stop Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
oidmon	UNIX: bin/oidmon Windows: bin\oidmon	Initiate, monitor, and terminate Oracle Internet Directory processes. **See:** *Oracle Identity Management User Reference*
oidpasswd	UNIX: bin/oidpasswd Windows: bin\oidpasswd	Change the Oracle Internet Directory database password. **See:** *Oracle Identity Management User Reference*
oidprovtool	UNIX: bin/oidprovtool Windows: bin\oidprovtool.bat	Administer provisioning profile entries in Oracle Internet Directory. **See:** *Oracle Identity Management User Reference*
oidreconcile	UNIX: bin/oidreconcile Windows: bin\oidreconcile	Synchronize Oracle Internet Directory entries. **See:** *Oracle Identity Management User Reference*
oidstats	UNIX: ldap/admin/oidstats.sh Windows: ldap\admin\oidstats.bat	Analyze the various database ods schema objects to estimate statistics. **See:** *Oracle Identity Management User Reference*
ojspc	UNIX: bin/ojspc Windows: bin\ojspc.bat	JSP back precompiler. **See:** *Oracle Application Server Containers for J2EE Support for JavaServer Pages Developer's Guide*

Table B–1 (Cont.) Oracle Application Server Command-Line Tools

Command	Path from Oracle Home	Description
opmnctl	UNIX: opmn/bin/opmnctl.exe Windows: opmn\bin\opmnctl.exe	Start, stop, and get status on OPMN-managed processes. **See:** *Oracle Process Manager and Notification Server Administrator's Guide*
ossoca.jar	UNIX: sso/lib/ossoca.jar Windows: sso\lib\ossoca.jar	Configure OracleAS Single Sign-On, including additional languages. **See:** *Oracle Application Server Single Sign-On Administrator's Guide*and*Oracle Application Server Globalization Guide*
ossoreg.jar	UNIX: sso/lib/ossoreg.jar Windows: sso\lib\ossoreg.jar	Register `mod_osso`. **See:** *Oracle Application Server Single Sign-On Administrator's Guide*
portalRegistrar	UNIX: wireless/bin/internal/portalRegistrar.sh Windows: wireless\bin\internal\portalRegistrar.bat	Reregister the mobile gateway parameter with OracleAS Portal. **See:** *Oracle Application Server Portal Configuration Guide* and *Oracle Application Server Wireless Administrator's Guide*
printlogs	UNIX: diagnostics/bin/printlogs Windows: diagnostics\bin\printlogs.bat	Print the contents of diagnostic log files to standard output. **See:** Appendix F, "printlogs Tool Syntax and Usage"
remtool	UNIX: ldap/bin/remtool Windows: ldap\bin\remtool	Search for problems and seek to rectify them in the event of an Oracle Internet Directory replication failure. **See:** *Oracle Identity Management User Reference*
reRegisterSSO	UNIX: wireless/bin/reRegisterSSO.sh Windows: wireless\bin\reRegisterSSO.bat	Reregister the Wireless Single Sign-On partner application with the Single Sign-On server. **See:** *Oracle Application Server Wireless Administrator's Guide*
resetiASpasswd	UNIX: bin/resetiASpasswd.sh Windows: bin\resetiASpasswd.bat	Reset the internal password that instances use to authenticate themselves with Oracle Internet Directory. Resets it to a randomly generated password. **See:** *Oracle Application Server Security Guide*
rwbuilder	UNIX: bin/rwbuilder Windows: bin\rwbuilder	Invoke the Reports Builder. **See:** *Oracle Application Server Reports Services Publishing Reports to the Web*
rwcgi	UNIX: bin/rwcgi Windows: bin\rwcgi	Like `rwservlet`, translate and deliver information between HTTP and the Reports Server. The `rwservlet` command is the recommended choice; `rwcgi` is maintained only for backward compatibility. **See:** *Oracle Application Server Reports Services Publishing Reports to the Web*
rwclient	UNIX: bin/rwclient Windows: bin\rwclient	Parse and transfer a command line to the specified (or default) Reports Server. **See:** *Oracle Application Server Reports Services Publishing Reports to the Web*
rwconverter	UNIX: bin/rwconverter Windows: bin\rwconverter	Convert one or more report definitions or PL/SQL libraries from one storage format to another. **See:** *Oracle Application Server Reports Services Publishing Reports to the Web*

Table B–1 (Cont.) Oracle Application Server Command-Line Tools

Command	Path from Oracle Home	Description
rwrun	UNIX: bin/rwrun Windows: bin\rwrun	Run a report using the OracleAS Reports Services in-process server. **See:** *Oracle Application Server Reports Services Publishing Reports to the Web*
rwserver	UNIX: bin/rwserver Windows: bin\rwserver.bat	Invoke the Reports Server. **See:** *Oracle Application Server Reports Services Publishing Reports to the Web*
schemasync	UNIX: bin/schemasync Windows: bin\schemasync.bat	Synchronize schema elements—namely attributes and object classes—between an Oracle directory server and third-party LDAP directories. **See:** *Oracle Identity Management User Reference*
ssocfg	UNIX: sso/bin/ssocfg.sh Windows: sso\bin\ssocfg.bat	Update host, port, and protocol of OracleAS Single Sign-On URL. **See:** *Oracle Application Server Single Sign-On Administrator's Guide*
ssooconf.sql	UNIX: portal/admin/plsql/sso/ssooconf.sql Windows: portal\admin\plsql\sso\ssooconf.sql	Script to point OracleAS Single Sign-On server to a different Oracle Internet Directory. **See:** *Oracle Application Server Single Sign-On Administrator's Guide*
stopodiserver	UNIX: ldap/odi/admin/stopodiserver.sh Windows: ldap\odi\admin\stopodiserver.bat	In a client-only installation where the Oracle Internet Directory Monitor and Oracle Internet Directory Control Utility are not available, you can start the directory integration and provisioning server without the `oidctl` tool. To stop the server, use the `stopodiserver` tool. **See:** *Oracle Identity Management User Reference*
uddiadmin.jar	UNIX: uddi/lib/uddiadmin.jar Windows: uddi\lib\uddiadmin.jar	Manage the UDDI registry, which is part of OracleAS Web Services. **See:** *Oracle Application Server Web Services Developer's Guide*
webcachectl	UNIX: bin/webcachectl Windows: bin\webcachectl.exe	In a standalone environment, administer OracleAS Web Cache processes, including the administration server process, cache server process, and auto-restart process. **See:** *Oracle Application Server Web Cache Administrator's Guide*

C

URLs for Components

Table C–1 shows the URLs and login IDs to use to access components after installation.

The URLs in the table are shown with the default ports. The components in your environment might use different ports. To determine the port numbers for components, you can look in the `ORACLE_HOME/install/portlist.ini` file.

Table C–1 URLs for Components

Component	URL (with Default Port Number)	Entry in portlist.ini	Login and Password
Welcome Pages	UNIX: http://*host*:7777 Windows: Infrastructure: http://*host*:7777 Middle-tier: http://*host*:80	Oracle HTTP Server port or Web Cache Listen port	n/a
Oracle HTTP Server	UNIX: http://*host*:7777 (without OracleAS Web Cache) http://*host*:7778 (with OracleAS Web Cache) Windows: Infrastructure: http://*host*:7777 Middle-tier: http://*host*:80	Oracle HTTP Server Listen port	n/a
OracleAS Web Cache Manager	http://*host*:9400/webcacheadmin	Web Cache Administration port	`administrator` or `ias_admin` Use the password for `ias_admin` that you supplied during installation.
OracleAS Portal	UNIX: http://*host*:7777/pls/portal Windows: http://*host*:80/pls/portal	Web Cache Listen port	`portal` Use the password for `ias_admin` that you supplied during installation.

Table C–1 (Cont.) URLs for Components

Component	URL (with Default Port Number)	Entry in portlist.ini	Login and Password
OracleAS Wireless	UNIX: http://*host*:7777/webtool/login.uix Windows: http://*host*:80/webtool/login.uix	Web Cache Listen port	`orcladmin` The default password is the same as the `ias_admin` password *of the Infrastructure* instance used by this middle-tier.
OracleAS UDDI Registry	UNIX: http://*host*:7777/uddi Windows: http://*host*:80/uddi	Web Cache Listen port	n/a
OracleAS Single Sign-On Administration Pages	http://*host*:7777/pls/orasso	Oracle HTTP Server Listen port	`orcladmin` The default password is the same as the `ias_admin` password, which you specified during installation.
Oracle Enterprise Manager 10*g* Application Server Control	UNIX: http://*host*:1156 Windows: http://*host*:18100	Application Server Control port	`ias_admin` Use the `ias_admin` password you supplied during installation.
Oracle Internet Directory Manager	UNIX: *ORACLE_HOME*/bin/oidadmin Windows: Select **Start** > **Programs** > **Oracle** - *OracleHome* > **Integrated Management Tools** > **Oracle Directory Manager**.	n/a	`orcladmin` The default password is the same as the `ias_admin` password, which you specified during installation.
Oracle Delegated Administration Services	http://*host*:7777/oiddas	Oracle HTTP Server Listen port	`orcladmin` The default password is the same as the `ias_admin` password, which you specified during installation.
OracleAS Certificate Authority Administration Interface	http://*host*:6600/oca/admin	Oracle Certificate Authority SSL Server Authentication port	`Certificate Authority Administrator` Use the password you supplied for the OracleAS Certificate Authority administrator during installation.
Oracle Business Intelligence Discoverer Viewer	http://*host*:7777/discoverer/viewer	Web Cache Listen port	n/a
Oracle Business Intelligence Discoverer Plus	http://*host*:7777/discoverer/plus	Web Cache Listen port	n/a

Table C–1 (Cont.) URLs for Components

Component	URL (with Default Port Number)	Entry in portlist.ini	Login and Password
Oracle Business Intelligence Discoverer Provider	http://*host*:7777/discoverer/provider	Web Cache Listen port	n/a
OracleAS Forms Services	http://*host*:7777/forms/frmservlet	Web Cache Listen port	n/a
OracleAS Reports Services	http://*host*:7777/reports/rwservlet/getserverinfo	Web Cache Listen port	`orcladmin` The default password is the same as the `ias_admin` password *of the Infrastructure* instance used by this middle-tier.

D

Oracle Application Server Port Numbers

This appendix provides information about Oracle Application Server port numbers.

It contains the following topics:

- Port Numbers and How They Are Assigned (Sorted by Installation Type)

  This section lists the allotted port range, the default port number, information about when the port number is assigned and where to find information about changing the port number.

- Port Numbers (Sorted by Port Number)

  This section provides a table that lists all allotted port ranges. It is useful for determining if a particular port number is used by Oracle Application Server.

D.1 Port Numbers and How They Are Assigned (Sorted by Installation Type)

Most port numbers are assigned by Oracle Application Server during installation. Oracle Application Server chooses a free port from the allotted port range.

This section provides the following information for each Oracle Application Server service that uses a port:

- **Component or Service:** The name of the component and service and information about when the port number is assigned and where to find information about changing the port number, if it can be changed.
- **Allotted Port Range:** The set of port numbers Oracle Application Server attempts to use when assigning a port.
- **Default Port Number:** The first port number Oracle Application Server attempts to assign to a service. It is usually the lowest number in the allotted port range.
- **Protocol:** Protocol used.

The ports are sorted by the following installation types:

- J2EE and OracleAS Web Cache Ports
- Portal and Wireless Ports
- Business Intelligence and Forms Ports
- Infrastructure Ports
- Oracle Application Server Integration Ports
- Oracle Enterprise Manager 10g Grid Control Ports

- Oracle Content Management Software Development Kit Ports
- OracleAS Developer Kits

D.1.1 J2EE and OracleAS Web Cache Ports

Table D–1 lists the ports in a J2EE and Web Cache installation. Unless otherwise noted in the Component or Service column of the table:

- The port number is assigned during installation
- If the port number is assigned during installation, you can override the port number during installation by using the `staticports.ini` file.

  For most ports, you can override the default port assignment during installation by specifying a port number in `staticports.ini`. You create a template called `staticports.ini` with the port numbers you would like to use, and launch Oracle Universal Installer with special options.

  See Also: *Oracle Application Server Installation Guide* for information on how to use `staticports.ini`

- You can change the port number after installation.

Table D–1 J2EE and Web Cache Ports

Component or Service	Allotted Port Range	Default Port Number	Protocol
Oracle HTTP Server			
Listen Port See Section 4.3.3, "Changing the Oracle HTTP Server Listen Ports" to change the port number.	7777 - 7877	Without Web Cache: 7777 With Web Cache: 7778	HTTP
Port See Section 4.3.3, "Changing the Oracle HTTP Server Listen Ports" to change the port number.	7777 - 7877	Without Web Cache: 7777[1] With Web Cache: 7777	HTTP
Listen (SSL) port This port is not used unless you enable SSL after installation. Refer to *Oracle HTTP Server Administrator's Guide*. See Section 4.3.3, "Changing the Oracle HTTP Server Listen Ports" to change the port number.	4443, 8250 - 8350	Without Web Cache: 4443 With Web Cache: 8250	HTTPS
SSL Port This port is not used unless you enable SSL after installation. Refer to *Oracle HTTP Server Administrator's Guide*. See Section 4.3.3, "Changing the Oracle HTTP Server Listen Ports" to change the port number.	443, 4443, 8250 - 8350	Without Web Cache: 4443[2] With Web Cache: 8250	HTTPS
Diagnostic See Section 4.3.4, "Changing the Oracle HTTP Server Diagnostic Port" to change the port number.	7200 - 7299	7200	TCP
OracleAS Web Cache			

Table D–1 (Cont.) J2EE and Web Cache Ports

Component or Service	Allotted Port Range	Default Port Number	Protocol
HTTP Listen See Section 4.3.5.1, "Changing the OracleAS Web Cache Listen Ports" to change the port number.	7777 - 7877	UNIX: 7777 Windows: 80	HTTP
HTTP Listen (SSL) This port is not used unless you enable SSL after installation. Refer to *Oracle Application Server Web Cache Administrator's Guide*. See Section 4.3.5.1, "Changing the OracleAS Web Cache Listen Ports" to change the port number.	8250 - 8350	8250[2]	HTTPS
Administration See Section 4.3.5.2, "Changing the OracleAS Web Cache Administration Port" to change the port number.	9400 - 9499	9400	HTTP
Invalidation See Section 4.3.5.3, "Changing the OracleAS Web Cache Invalidation Port" to change the port number.	9400 - 9499	9401	HTTP
Statistics See Section 4.3.5.4, "Changing the OracleAS Web Cache Statistics Port" to change the port number.	9400 - 9499	9402	HTTP
OC4J			
AJP You cannot override this port number during installation. See Section 4.3.2, "Changing OC4J Ports" to change the port number.	12501 - 12600	12501	TCP
JMS You cannot override this port number during installation. See Section 4.3.2, "Changing OC4J Ports" to change the port number.	12601 - 12700	12601	TCP
RMI You cannot override this port number during installation. See Section 4.3.2, "Changing OC4J Ports" to change the port number.	12401 - 12500	12401	TCP
IIOP Port is assigned after installation, when you configure IIOP. Refer to *Oracle Application Server Containers for J2EE Services Guide*. See Section 4.3.2, "Changing OC4J Ports" to change the port number.	13301 - 13400	13301	TCP
IIOPS1 (Server only) Port is assigned after installation, when you configure IIOPS1. See Section 4.3.2, "Changing OC4J Ports" to change the port number.	13401 - 13500	13401	TCP
IIOPS2 (Server and client) Port is assigned after installation, when you configure IIOPS2. See Section 4.3.2, "Changing OC4J Ports" to change the port number.	13501 - 13600	13501	TCP

Table D–1 (Cont.) J2EE and Web Cache Ports

Component or Service	Allotted Port Range	Default Port Number	Protocol
OPMN			
ONS Local See Section 4.3.9, "Changing OPMN Ports (ONS Local, Request, and Remote)" to change the port number.	6100 - 6199	6100	HTTP/ TCP
ONS Remote See Section 4.3.9, "Changing OPMN Ports (ONS Local, Request, and Remote)" to change the port number.	6200 - 6299	6200	HTTP/ TCP
ONS Request See Section 4.3.9, "Changing OPMN Ports (ONS Local, Request, and Remote)" to change the port number.	6003 - 6099	6003	HTTP/ TCP
Oracle Enterprise Manager			
Application Server Control Console See Section 4.3.1, "Changing Oracle Enterprise Manager Ports" to change the port number.	1156, 1810-1829, 18100 - 18119	UNIX: 1156 Windows: 18100	HTTP
Application Server Control Console (SSL) Port is assigned after installation, when you configure the Application Server Control Console for SSL. Refer to Section A.4. See Section 4.3.1, "Changing Oracle Enterprise Manager Ports" to change the port number.	1156, 1810-1829, 18100 - 18119	UNIX: 1156 Windows: 18100	HTTPS
Application Server Control Console RMI See Section 4.3.1, "Changing Oracle Enterprise Manager Ports" to change the port number.	1850, 18140 - 18159	1850	TCP
Oracle Management Agent See Section 4.3.1, "Changing Oracle Enterprise Manager Ports" to change the port number.	1157, 18120 - 18139	1157	TCP
Miscellaneous Services			
DCM Discovery See Section 4.3.6, "Changing the DCM Discovery Port" to change the port number	7100 - 7199	7100	TCP
Java Object Cache See Section 4.3.7, "Changing the Java Object Cache Port" to change the port number.	7000 - 7099	7000	TCP
Log Loader See Section 4.3.8, "Changing the Log Loader Port" to change the port number.	44000 - 44099	44000	TCP
Port Tunneling Port is assigned after installation, when you configure Port Tunneling. See Section 4.3.10, "Changing the Port Tunneling Port" to change the port number.	7501 - 7599	7501	TCP

[1] The default port is 80 on middle-tier installations on Windows.

[2] The default port is 443 on middle-tier installations on Windows.

D.1.2 Portal and Wireless Ports

A Portal and Wireless installation uses the ports listed in:

- Table D–1, " J2EE and Web Cache Ports"
- Table D–2, " Portal and Wireless Ports"

Unless otherwise noted in the Component or Service column of the table:

- The port number is assigned during installation
- If the port number is assigned during installation, you can override the port number during installation by using the `staticports.ini` file.

  For most ports, you can override the default port assignment during installation by specifying a port number in `staticports.ini`. You create a template called `staticports.ini` with the port numbers you would like to use, and launch Oracle Universal Installer with special options.

  > **See Also:** *Oracle Application Server Installation Guide* for information on how to use `staticports.ini`

- You can change the port number after installation.

Table D–2 Portal and Wireless Ports

Component / Service	Allotted Port Range	Default Port Number	Protocol
Oracle Ultra Search			
RMI Daemon You cannot override this port number during installation. You cannot change this port number.	1098	1098	TCP
RMI Registry You cannot override this port number during installation. You cannot change this port number.	1099	1099	TCP
OracleAS Portal			
OracleAS Portal[1]	N/A	N/A	N/A
OracleAS Wireless			
OracleAS Wireless[1]	N/A	N/A	N/A
Wireless Notification Dispatcher Calendar You cannot override this port number during installation. You cannot change this port.	9100 - 9199	9100	TCP

[1] This service does not have its own port. You can access it through the HTTP listener port.

D.1.3 Business Intelligence and Forms Ports

A Business Intelligence and Forms installation uses the ports listed in:

- Table D–1, " J2EE and Web Cache Ports"
- Table D–2, " Portal and Wireless Ports"
- Table D–3, " Business Intelligence and Forms Ports"

Unless otherwise noted in the Component or Service column of the table:

- The port number is assigned during installation
- If the port number is assigned during installation, you can override the port number during installation by using the `staticports.ini` file.

  For most ports, you can override the default port assignment during installation by specifying a port number in `staticports.ini`. You create a template called `staticports.ini` with the port numbers you would like to use, and launch Oracle Universal Installer with special options.

  See Also: *Oracle Application Server Installation Guide* for information on how to use `staticports.ini`

- You can change the port number after installation.

Table D–3 Business Intelligence and Forms Ports

Component / Service	Allotted Port Range	Default Port Number	Protocol
OracleBI Discoverer			
OracleBI Discoverer[1]	N/A	N/A	N/A
OracleBI Discoverer Preferences	16001 - 16020	16001	TCP
OracleAS Forms Services			
OracleAS Forms Services	N/A	N/A	N/A
OracleAS Reports Services			
Bridge (RWBRIDGE) See Section 4.3.15.1, "Changing the OracleAS Reports Services Bridge Port" to change the port number.	14011 - 14020	14011	TCP
Discovery Service (RWNETWORK) See Section 4.3.15.2, "Changing the OracleAS Reports Services Network Port" to change the port number.	14021 - 14030	14021	UDP
SQL*Net *For 6i Backward Compatibility Only* See Section 4.3.15.3, "Changing the OracleAS Reports Services SQL*Net Port" to change the port number.	14040 - 14049	14040	TCP

[1] This service does not have its own port. You access it through the HTTP listener port.

D.1.4 Infrastructure Ports

An Infrastructure installation uses the ports listed in:

- Table D–1, " J2EE and Web Cache Ports"
- Table D–4, " Infrastructure Ports"

Unless otherwise noted in the Component or Service column of the table:

- The port number is assigned during installation
- If the port number is assigned during installation, you can override the port number during installation by using the `staticports.ini` file.

  For most ports, you can override the default port assignment during installation by specifying a port number in `staticports.ini`. You create a template called

`staticports.ini` with the port numbers you would like to use, and launch Oracle Universal Installer with special options.

> **See Also:** *Oracle Application Server Installation Guide* for information on how to use `staticports.ini`

- You can change the port number after installation.

Table D–4 Infrastructure Ports

Component / Service	Allotted Port Range	Default Port Number	Protocol
Oracle Enterprise Manager			
Oracle Management Agent Section 4.3.1, "Changing Oracle Enterprise Manager Ports"	1157, 18120 - 18139	1157	TCP
Console HTTP You cannot change this port, although you can override the port number during installation.	5500 - 5559	5500	HTTP
Oracle Internet Directory			
Oracle Internet Directory Section 4.4.2, "Changing Oracle Internet Directory Ports"	389, 13060 - 13129	389	TCP
Oracle Internet Directory (SSL) Section 4.4.2, "Changing Oracle Internet Directory Ports"	636, 13130 - 13159 13161 - 13199	636	TCP
OracleAS Certificate Authority			
Server Authentication Virtual Host (SSL) Section 4.4.4, "Changing OracleAS Certificate Authority Ports"	6600 - 6619	6600	HTTPS
Mutual Authentication Virtual Host (SSL) Section 4.4.4, "Changing OracleAS Certificate Authority Ports"	6600 - 6619	6601	HTTPS
OracleAS Metadata Repository			
Oracle Net Listener You cannot override this port number during installation. Section 4.4.1, "Changing the OracleAS Metadata Repository Net Listener Port"	1521	1521	TCP
OracleAS Single Sign-On			
OracleAS Single Sign-On[1]	N/A	N/A	N/A
Oracle Application Server Guard			
Oracle Application Server Guard	7890 - 7895	7890	TCP

[1] This service does not have its own port. You can access it through the HTTP listener port.

D.1.5 Oracle Application Server Integration Ports

For information about Oracle Application Server Integration component ports, such as OracleAS Integration B2B, Oracle BPEL Process Manager, and Oracle BPEL Process Analytics, see the Oracle Application Server Integration documentation.

D.1.6 Oracle Enterprise Manager 10*g* Grid Control Ports

Table D–5 lists the ports used in an Oracle Enterprise Manager 10*g* Grid Control installation.

Unless otherwise noted in the Component or Service column of the table:

- The port number is assigned during installation
- If the port number is assigned during installation, you can override the port number during installation by using the `staticports.ini` file.

  For most ports, you can override the default port assignment during installation by specifying a port number in `staticports.ini`. You create a template called `staticports.ini` with the port numbers you would like to use, and launch Oracle Universal Installer with special options.

  See Also: *Oracle Application Server Installation Guide* for information on how to use `staticports.ini`

- You can change the port number after installation.

Table D–5 Oracle Enterprise Manager 10g Grid Control Ports

Service	Allotted Port Range	Default Port Number	Protocol
Grid Control Console[1]	N/A	N/A	N/A
Oracle Management Agent You cannot override this port number during installation. See *Oracle Enterprise Manager Advanced Configuration* to change the port number.	1830 - 1849	1830	TCP
Oracle Management Service (SSL and non-SSL) You cannot override this port number during installation. See *Oracle Enterprise Manager Advanced Configuration* to change the port number.	4889 - 4899	4889	HTTP

[1] This service does not have its own port. You can access it through the HTTP listener port. Refer to `setupinfo.txt` in the `install` directory within the Grid Control Oracle home for the exact URL.

D.1.7 Oracle Content Management Software Development Kit Ports

Table D–6 lists the ports used in an Oracle Content Management Software Development Kit installation.

Unless otherwise noted in the Component or Service column of the table:

- The port number is assigned during installation
- If the port number is assigned during installation, you can override the port number during installation by using the `staticports.ini` file.

  For most ports, you can override the default port assignment during installation by specifying a port number in `staticports.ini`. You create a template called `staticports.ini` with the port numbers you would like to use, and launch Oracle Universal Installer with special options.

  See *Oracle Application Server Installation Guide* for information on how to use `staticports.ini`

- You can change the port number after installation.

Table D–6 Oracle Content Management Software Development Kit Ports

Service	Allotted Port Range	Default Port Number	Protocol
AFP: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	548	548	AFP
CUP: *(myhost.mydomain Node)* You cannot override this port number during installation. To change the port number, edit `CupServerConfiguration` and update `IFS.SERVER.PROTOCOL.CUP.Port`. Then reload the CUP server. Refer to *Oracle Content Management SDK Administrator's Guide*.	4180	4180	CUP
Domain Controller You cannot change this port number.	53140 - 53999	N/A	N/A
FTP: *(myhost.mydomain Node)* You cannot override this port number during installation. To change the port number, edit `FtpServerConfiguration` and update `IFS.SERVER.PROTOCOL.FTP.Port`. Then reload the FTP server. Refer to *Oracle Content Management SDK Administrator's Guide*.	21	21	FTP
IMAP: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	143	143	IMAP
IMAP (SSL): *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	993	993	IMAP
NB UDP: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	137	137	SMB
NFS: *(myhost.mydomain Node)* You cannot override this port number during installation. To change the port number, edit `NfsServerConfiguration` and update `IFS.SERVER.PROTOCOL.NFS.Port`. Then reload the NFS Server. Refer to *Oracle Content Management SDK Administrator's Guide*.	2049	2049	NFS
NFS Mount Point: *(myhost.mydomain Node)* You cannot override this port number during installation. To change the port number, edit `NfsServerConfiguration` and update `IFS.SERVER.PROTOCOL.NFS.MountPort`. Then reload the NFS Server. Refer to *Oracle Content Management SDK Administrator's Guide*.	N/A	N/A	NFS
Node Guardian: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	53140 - 53999	N/A	N/A

Table D–6 (Cont.) Oracle Content Management Software Development Kit Ports

Service	Allotted Port Range	Default Port Number	Protocol
Node Guardian: *(myhost.mydomain HTTP Node)* You cannot override this port number during installation. You cannot change this port number.	53140 - 53999	N/A	N/A
Node Manager: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	53140 - 53999	N/A	N/A
Node Manager: *(myhost.mydomain HTTP Node)* You cannot override this port number during installation. You cannot change this port number.	53140 - 53999		N/A
SMB: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	139	139	SMB
SMTP: *(myhost.mydomain Node)* You cannot override this port number during installation. You cannot change this port number.	25	25	SMTP

D.1.8 OracleAS Developer Kits

OracleAS Developer Kits use the same ports as the J2EE and Web Cache installation type.

See Also: Section D.1.1, "J2EE and OracleAS Web Cache Ports"

D.2 Port Numbers (Sorted by Port Number)

Table D–7 lists Oracle Application Server ports numbers and services, sorted in ascending order by port number.

Table D–7 Port Numbers (Sorted by Port Number)

Port Number	Service
21	Oracle Content Management Software Development Kit FTP
25	Oracle Content Management Software Development Kit SMTP
137	Oracle Content Management Software Development Kit NB UDP
139	Oracle Content Management Software Development Kit SMB
143	Oracle Content Management Software Development Kit IMAP (non-SSL)
389	Oracle Internet Directory (non-SSL)
443	Oracle HTTP Server Port (SSL) (Windows only) OracleAS Web Cache HTTP Listen (SSL) (Windows only)
548	Oracle Content Management Software Development Kit AFP
636	Oracle Internet Directory Server (SSL)
993	Oracle Content Management Software Development Kit IMAP (SSL)
1098	Oracle Ultra Search RMI Daemon

Table D–7 (Cont.) Port Numbers (Sorted by Port Number)

Port Number	Service
1099	Oracle Ultra Search RMI Registry
1156	Oracle Enterprise Manager 10*g* Application Server Control Console (non-SSL and SSL) (UNIX)
1157	Oracle Management Agent
1521	OracleAS Metadata Repository Oracle Net Listener
1810-1829	Oracle Enterprise Manager 10*g* Application Server Control Console (non-SSL and SSL)
1830-1849	Oracle Management Agent
1850-1869	Oracle Enterprise Manager 10*g* Application Server Control Console RMI
2049	Oracle Content Management Software Development Kit NFS
2550 - 2577	OracleAS Integration B2B Attunity Adapters (Legacy Adapters)
4180	Oracle Content Management Software Development Kit CUP
4443	Oracle HTTP Server Listen (SSL) and Oracle HTTP Server Port (SSL)
4550 - 4599	OracleAS Integration B2B Actional Listener
4889 - 4899	Oracle Management Service (SSL and non-SSL)
5110 - 5119	OracleAS Integration B2B Adapter RMI
5500 - 5559	Oracle Enterprise Manager Console HTTP
6003 - 6099	OPMN ONS Request
6100 - 6199	OPMN ONS Local
6200 - 6299	OPMN ONS Remote
6600 - 6619	OracleAS Certificate Authority Server Authentication Virtual Host (SSL) OracleAS Certificate Authority Mutual Authentication Virtual Host (SSL)
7000 - 7099	Java Object Cache
7100 - 7199	DCM Discovery
7200 - 7299	Oracle HTTP Server Diagnostic
7501 - 7599	Port Tunneling
7777 - 7877	Oracle HTTP Server Listen and Oracle HTTP Server Port OracleAS Web Cache HTTP Listen
7890 - 7895	Oracle Application Server Guard
8250 - 8350	Oracle HTTP Server Listen (SSL) and Oracle HTTP Server Port (SSL) OracleAS Web Cache HTTP Listen (SSL)
8777 - 8900	OracleAS Integration B2B Integration Manager OracleAS Integration B2B Adapter Framework
9100 - 9199	Wireless Notification Dispatcher Calendar
9400 - 9499	OracleAS Web Cache Administration OracleAS Web Cache Invalidation OracleAS Web Cache Statistics
9700	Oracle BPEL Process Manager
9901	OracleAS Integration InterConnect RMI port for HTTP

Table D–7 (Cont.) Port Numbers (Sorted by Port Number)

Port Number	Service
9998-9999	SOAP server
12401 - 12500	OC4J RMI
12501 - 12600	OC4J AJP
12601 - 12700	OC4J JMS
13060 - 13129	Oracle Internet Directory (non-SSL)
13130 - 13159	Oracle Internet Directory (SSL)
13161 - 13199	Oracle Internet Directory (SSL)
13301 - 13400	OC4J IIOP
13401 - 13500	OC4J IIOPS1 (Server only)
13501 - 13600	OC4J IIOPS2 (Server and client)
14011 - 14020	OracleAS Reports Services bridge (RWBRIDGE)
14021 - 14030	OracleAS Reports Services Discovery Service (RWNETWORK)
14040 - 14049	OracleAS Reports Services SQL*Net
16001 - 16020	OracleBI Discoverer Preferences
18100 - 18119	Oracle Enterprise Manager 10*g* Application Server Control Console (non-SSL and SSL)
18120 - 18139	Oracle Management Agent
18140 - 18159	Oracle Enterprise Manager 10*g* Application Server Control Console RMI
20300 - 20350	OracleAS Integration B2B Actional Agent
44000 - 44099	Log Loader
53140 - 53999	Oracle Content Management Software Development Kit Domain Controller Oracle Content Management Software Development Kit Node Guardian Oracle Content Management Software Development Kit Node Manager

E

Metadata Repository Schemas

A Metadata Repository is an Oracle database that is pre-seeded with additional schemas to support Oracle Application Server. This appendix provides information about those schemas.

It contains the following topics:

- Metadata Repository Schema Descriptions
- Metadata Repository Schemas, Tablespaces, and Default Datafiles

E.1 Metadata Repository Schema Descriptions

This section lists the Metadata Repository schemas and describes their contents.

The schemas are divided into three categories:

- Identity Management Schemas

  These schemas are used by Identity Management components, such as OracleAS Single Sign-On and Oracle Internet Directory.

- Product Metadata Schemas

  These schemas are used by middle-tier application components, such as OracleAS Portal and OracleAS Wireless.

- Management Schemas

  These schemas are used for Oracle Application Server management.

There is one additional schema that does not fall into the previously listed categories: INTERNET_APPSERVER_REGISTRY. This schema contains release numbers for Metadata Repository schemas.

> **See Also:** Section I.5, "Viewing Metadata Repository Release Numbers" for information on using the INTERNET_APPSERVER_REGISTRY schema to query release numbers

E.1.1 Identity Management Schemas

Table E–1 lists the schemas used by Identity Management components, sorted alphabetically by component.

Table E–1 Identity Management Schemas

Component	Schema	Description
Oracle Internet Directory	ODS	For internal use
OracleAS Certificate Authority	OCA	For internal use
OracleAS Certificate Authority	ORAOCA_PUBLIC	For internal use
OracleAS Single Sign-On	ORASSO	For internal use
OracleAS Single Sign-On	ORASSO_DS	For internal use
OracleAS Single Sign-On	ORASSO_PA	For internal use
OracleAS Single Sign-On	ORASSO_PS	For internal use
OracleAS Single Sign-On	ORASSO_PUBLIC	For internal use

E.1.2 Product Metadata Schemas

Table E–2 lists the schemas used by middle-tier application components, sorted alphabetically by component.

Table E–2 Product Metadata Schemas

Component	Schema	Description
Oracle Ultra Search	WK_TEST	Oracle Ultra Search default instance schema—contains the document information and document index of the default Oracle Ultra Search instance
Oracle Ultra Search	WKPROXY	Oracle Ultra Search proxy database user—does not contain any data
Oracle Ultra Search	WKSYS	Oracle Ultra Search metadata repository—contains metadata information on data sources, crawler configuration, crawling schedules, trace logs, attribute mappings, authentication, and user privileges of Oracle Ultra Search instances
Oracle Workflow	OWF_MGR	Contains design-time and runtime workflow tables, queues, PL/SQL code, directory service database views and local tables, and metadata for workflow processes and business events
OracleAS Integration B2B	B2B	Contains the design and runtime repository. The design repository has modeling metadata and profile data for an integration. These describe the behavior of the integration and sequence of steps required to execute the business process. The modeling and profile metadata is the design of the integration prior to deployment and execution. Once the integration is deployed, the runtime repository contains the metadata required to execute the integration as well as the business process instance, event instances, role instances, and other data created during execution.
OracleAS Integration B2B	IP	N/A[1]
Oracle BPEL Process Analytics	BAM	Contains instance and metadata database objects for Oracle BPEL Process Analytics.
Oracle BPEL Process Manager	ORABPEL	Contains instance and metadata database objects for Oracle BPEL Process Manager.
OracleAS Portal	PORTAL	Contains Portal database objects and code. This schema also represents the proxy user account that `mod_plsql` uses to connect to the database through the credentials provided in the corresponding DAD.

Table E–2 (Cont.) Product Metadata Schemas

Component	Schema	Description
OracleAS Portal	PORTAL_APP	Contains information for authentication of external JSP applications
OracleAS Portal	PORTAL_DEMO	Contains demonstration code
OracleAS Portal	PORTAL_PUBLIC	OracleAS Portal schema for all lightweight users, who are mapped to this schema by default. All procedures publicly accessible through the Web are granted execute to `PUBLIC`, which makes them accessible through this schema.
OracleAS UDDI Registry	UDDISYS	Contains UDDI entities such as business entities, business services, binding templates, tModels, and publisher assertions; taxonomy structures such as North American Industry Classification System (NAICS), Universal Standard Products and Services Codes (UNSPSC), and ISO 3166 Geographic Taxonomy (ISO 3166); UDDI replication-related internal tables; and other administration-related views and tables
OracleAS Web Clipping	WCRSYS	Web Clipping Repository for support with OracleAS Wireless—contains clipping definitions, user customizations, and PL/SQL packages for their access
OracleAS Wireless	WIRELESS	Contains user content (folders, services, links, notifications, presets), user customization data, groups, roles, transient user information, style sheets, logical device definitions, Java transformers (serialized), adapters, location data, configuration data, process runtime state, and application metrics
OracleBI Discoverer	DISCOVERER5	Contains metadata for Discoverer Portlet Provider, portlet definitions for user portlets, and cached data obtained by running scheduled Discoverer queries. Has `RESOURCE` and `CONNECT` privileges.
N/A	DSGATEWAY[2]	N/A

[1] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the IP schema contains no data. It has been replaced by the B2B schema and is provided only for backward compatibility.

[2] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the DSGATEWAY schema is not used. It is provided for backward compatibility.

E.1.3 Management Schemas

Table E–3 lists the management schemas.

Table E–3 Management Schema

Component	Schema	Description
Distributed Configuration Management (DCM)	DCM	Contains configuration information for OC4J and Oracle HTTP Server instances, application server instances, OracleAS Clusters, and farms
Oracle Enterprise Manager	OEM_REPOSITORY	Repository for Database Control

E.2 Metadata Repository Schemas, Tablespaces, and Default Datafiles

Table E–4 lists the tablespace and default datafile for each Metadata Repository schema. It is sorted alphabetically by component.

Table E–4 Metadata Repository Tablespaces and Default Datafiles

Component	Schema	Tablespace	Default Datafile
Distributed Configuration Management (DCM)	DCM	DCM	dcm.dbf
Metadata Repository Version	INTERNET_APPSERVER_ REGISTRY	IAS_META	ias_meta01.dbf
Oracle Enterprise Manager	OEM_REPOSITORY	SYS_AUX	sysaux01.dbf
Oracle Internet Directory	ODS	OLTS_ATTRSTORE	attrs1_oid.dbf
Oracle Internet Directory	ODS	OLTS_BATTRSTORE	battrs1_oid.dbf
Oracle Internet Directory	ODS	OLTS_CT_STORE	gcats1_oid.dbf
Oracle Internet Directory	ODS	OLTS_DEFAULT	gdefault1_oid.dbf
Oracle Internet Directory	ODS	OLTS_SVRMGSTORE	svrmg1_oid.dbf
Oracle Ultra Search	WK_TEST	SYSAUX	sysaux01.dbf
Oracle Ultra Search	WKPROXY	SYSAUX	sysaux01.dbf
Oracle Ultra Search	WKSYS	SYSAUX	sysaux01.dbf
Oracle Workflow	OWF_MGR	IAS_META	ias_meta01.dbf
OracleAS Certificate Authority	OCA	OCATS	oca.dbf
OracleAS Certificate Authority	ORAOCA_PUBLIC	OCATS	oca.dbf
OracleAS Integration B2B	B2B	B2B_DT	b2b_dt.dbf
OracleAS Integration B2B	B2B	B2B_RT	b2b_rt.dbf
OracleAS Integration B2B	B2B	B2B_LOB	b2b_lob.dbf
OracleAS Integration B2B	B2B	B2B_IDX	b2b_idx.dbf
OracleAS Integration B2B	IP[1]	N/A	N/A
Oracle BPEL Process Analytics	BAM	BAM	bam.dbf
Oracle BPEL Process Manager	ORABPEL	ORABPEL	orabpel.dbf
OracleAS Portal	PORTAL	PORTAL	portal.dbf
OracleAS Portal	PORTAL	PORTAL_DOC	ptldoc.dbf
OracleAS Portal	PORTAL	PORTAL_IDX	ptlidx.dbf
OracleAS Portal	PORTAL	PORTAL_LOG	ptllog.dbf
OracleAS Portal	PORTAL_APP	PORTAL	portal.dbf
OracleAS Portal	PORTAL_DEMO	PORTAL	portal.dbf
OracleAS Portal	PORTAL_PUBLIC	PORTAL	portal.dbf
OracleAS Single Sign-On	ORASSO	IAS_META	ias_meta01.dbf
OracleAS Single Sign-On	ORASSO_DS	IAS_META	ias_meta01.dbf
OracleAS Single Sign-On	ORASSO_PA	IAS_META	ias_meta01.dbf
OracleAS Single Sign-On	ORASSO_PS	IAS_META	ias_meta01.dbf
OracleAS Single Sign-On	ORASSO_PUBLIC	IAS_META	ias_meta01.dbf
OracleAS UDDI Registry	UDDISYS	UDDISYS_TS	uddisys01.dbf
OracleAS Web Clipping	WCRSYS	WCRSYS_TS	wcrsys01.dbf

Table E–4 (Cont.) Metadata Repository Tablespaces and Default Datafiles

Component	Schema	Tablespace	Default Datafile
OracleAS Wireless	WIRELESS	IAS_META	ias_meta01.dbf
OracleBI Discoverer	DISCOVERER5	DISCO_PTM5_META	discopltm1.dbf
OracleBI Discoverer	DISCOVERER5	DISCO_PTM5_CACHE	discopltm1
N/A	DSGATEWAY[2]	DSGATEWAY_TAB	oss_sys01.dbf

[1] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the IP schema does not contain any data. It has been replaced by the B2B schema and is provided only for backward compatibility.

[2] Beginning with Oracle Application Server 10*g* Release 2 (10.1.2), the DSGATEWAY schema is not used. It is provided for backward compatibility.

F

printlogs Tool Syntax and Usage

This appendix describes the `printlogs` command-line tool. You can use `printlogs` to print the contents of Oracle Application Server diagnostic log files to standard output.

It contains the following topics:

F.1 Introduction

The `printlogs` command-line tool reads logs generated by Oracle Application Server components and prints the content of the logs to standard output in a common format. `printlogs` supports many options for reading and filtering log files, and formatting the output.

> **See Also:** Chapter 5, "Managing Log Files" for more information on Oracle Application Server logging

Location

The `printlogs` command-line tool is located in:

- For UNIX systems:

  ORACLE_HOME`/diagnostics/bin`

- For Windows systems:

  ORACLE_HOME`\diagnostics\bin`

Notes

- To run `printlogs`, you must log in as a user who has permission to read all of the log files in your Oracle home, for example, the user who installed Oracle Application Server.
- By default, `printlogs` operates on the Oracle home in which it resides. You can override this with the `-home` option. Note that `printlogs` does not use the `ORACLE_HOME` environment variable.

- `printlogs` options are not case-sensitive.
- By default, `printlogs` uses the contents of the following directory to determine which log files to read, the location of log files, and additional configuration information about each log file:

  ```
  (UNIX) ORACLE_HOME/diagnostics/config/registration
  (Windows) ORACLE_HOME\diagnostics\config\registration
  ```

  You can override this with the `-repository`, `-registration`, and `-logs` options.

  See Also: Section 5.6.4, "Component Diagnostic Log File Registration" for more information

F.2 Basic Syntax

```
printlogs [input options] [filter options] [output options] [general options]
```

Input Options

```
[-home oracle_home_path] [-repository]

[-home oracle_home_path] [-registration registration_directory_path]
[filter options] [output options] [general options]
[-logs log_path [log_path ...]]
```

Filter Options

```
[-tail n] [-last n[s|m|h|d]] [-query expression]

expression:
simple_expression
-not simple_expression
simple_expression -and simple_expression
simple_expression -or simple_expression

simple_expression:
field_name op value
( expression )

op:
-eq | -eq_case | -contains | -contains_case |
-startswith | -startswith_case | -from | -to
```

field_name: An ODL log record field name. See Section F.4, "Log Record Fields" for a list of field names.

value: A string or timestamp, depending on the operation (*op*)

Output Options

```
[-odl | -odl_complete | -text | -text_short | -text_full] [-orderBy
orderByFieldList]

[-count [groupByFieldList]]
```

General Options

```
[-help] [-f] [-sleep n] [-notailopt]
```

F.3 Detailed Option Descriptions

This section provides detailed descriptions of `printlogs` options. It contains the following sections:

- Input Options
- Filter Options
- Output Options
- General Options

F.3.1 Input Options

Table F–1 describes the input options you can use to specify the location of logs and log definitions. The default is the local Oracle home.

Table F–1 Input Options

Input Option	Description
`-home` *`oracle_home_path`*	Specify an alternate Oracle home directory from where to read logs and log definitions.
`-repository`	Specify that log records should be read from the common repository instead of directly from each log. The common repository is updated by Log Loader. Log Loader must be running in order for the repository to contain the contents of Oracle Application Server component logs.
`-repos`	Same as -repository

`-registration` *`registration_directory_path`*	Specify an alternate registration directory that contains definitions of log files to be read by `printlogs`. The default registration directory is:
	`(UNIX)` *`ORACLE_HOME`*`/diagnostics/config/registration`
	`(Windows)` *`ORACLE_HOME`*`\diagnostics\config\registration`

`-logs` *`log_path`* `[log_path ...]`	Specify one or more logs to be read by `printlogs`. *`log_path`* is the full path to the log file, or the path relative to the current directory.
	The registration directory is used to find the definition of each log. If one of the specified logs is not defined in the registration directory, it is read by the default "UnformattedTextLogReader".
	The path list is terminated by the end of the argument list or by the first argument following the `-logs` option that starts with a hyphen (-). Therefore, a log path cannot start with a hyphen (-). If a path starts with hyphen (-), precede the path with: `./` (UNIX) or `.\` (Windows). For example, to print log file -error.log, use:
	`(UNIX) printlogs -logs ./-error.log`
	`(Windows) printlogs -logs .\-error.log`

F.3.2 Filter Options

Table F–2 describes the filter options you can use to define which log records `printlogs` should print. The default is to print all records generated in the last 10 minutes.

Table F–2 Filter Options

Filter Option	Description
`-tail` *n*	Perform an operation similar to the UNIX "tail" command before reading a log. The *n* argument must be a positive number. The meaning of the *n* argument depends on the log type. For ODL logs, `printlogs` searches backward from the end of the log for *n* occurrences of the pattern "`<MESSAGE>`" and starts reading the log from that point. For other log types, it reads the last *n* lines of the log. Note that the use of the `-tail` option disables the use of a default value for the `-last` option. See the `-last` option for default value details.
`-last` *n*`[s|m|h|d]`	Print only logs generated in a specified period of time. The default is 10 minutes. If you do not specify the `-last` option, the period of time is set to 10 minutes except when the `-tail` option is specified. When the `-tail` option is specified, the default value for `-last` is disabled. If you specify both `-tail` and `-last` options, both options are used. You can use the *n* argument to specify a different period of time. The *n* argument must be a positive number. You can use a suffix to specify a unit of time: `s` for seconds, `m` for minutes, `h` for hours, and `d` for days. The default unit of time is minutes. To search through the logs generated over a large period of time, you can use a large value such as `100d`. The value of the `-last` option is used by `printlogs` to perform a "tail optimization" before it starts reading the logs. It performs an operation similar to the UNIX "tail" command to each log until it finds a timestamp that is within the desired range. This speeds up most inquiries significantly, but, if the log contains records out of timestamp order, it can cause `printlogs` to miss some records. It can also make queries slower in a few cases, for example, when you search the entire log. You can disable "tail optimization" with the -notailopt option.
`-query` *expression*	Apply *expression* to each log record to filter out undesirable records. See Table F–3 for a description of *expression*.

Table F–3 describes the query expressions you can use with the `-query` filter option in the `printlogs` command.

Table F–3 Query Expression Options

Query Expression Option	Description
`()`	Delimiters for complex sub-expressions. Parenthesis have special meaning to most UNIX command shells and you must use an escape character with them. This is not necessary on Windows.
`-not`	Logical negation
`-and`	Logical AND
`-or`	Logical OR
fieldname	An ODL log record field name. See Section F.4, "Log Record Fields" for a list of available field names.
`-eq`	Equality operation (case-insensitive). You can use this operation with all log record fields.
`-eq_case`	Same as -eq, except case-sensitive
`-contains`	Contains operation (case-insensitive). The result is true only if the log record field value contains the value operand string. You can use this operation only with "string" log record fields (all fields except `TSTZ_ORIGINATING` and `TSTZ_NORMALIZED`).
`-contains_case`	Same as -contains, except case-sensitive

Table F–3 (Cont.) Query Expression Options

Query Expression Option	Description
-startswith	Starts with operation (case-insensitive). The result is true only if the log record field value starts with the value operand string. You can use this operation only with "string" log record fields (all fields except TSTZ_ORIGINATING and TSTZ_NORMALIZED).
-startswith_case	Same as -startswith, except case-sensitive
-from	This operation can only be used with timestamped log record fields (TSTZ_ORIGINATING and TSTZ_NORMALIZED). The result is true only if the log record timestamp is equal to or greater than the operand value. The operand value must be in one of the following formats: ■ ISO 8601 time format: 2003-06-30T12:00:00:0000-08:00 ■ printlogs text output format: 30/JUN/2003:12:00:00.000-08:00 ■ date/time of the default Java locale format The fraction of seconds is optional for the ISO 8601 time format and the printlogs text output format. If the operand contains a space, the operand must be enclosed in quotes. By default, printlogs searches for timestamped records generated in the last 10 minutes. You can use the -last n[s|m|h|d] option in conjunction with the -from option to ensure the search period includes the specified timestamped records.
-to	This operation can only be used with timestamped log record fields (TSTZ_ORIGINATING and TSTZ_NORMALIZED). The result is true only if the log record timestamp is less than or equal to the operand value. The operand value must be in one of the following formats: ■ ISO 8601 time format: 2003-06-30T12:00:00:0000-08:00 ■ printlogs text output format: 30/JUN/2003:12:00:00.000-08:00 ■ date/time of the default Java locale format The fraction of seconds is optional for the ISO 8601 time format and the printlogs text output format. If the operand contains a space, the operand must be enclosed in quotes. By default, printlogs searches for timestamped records generated in the last 10 minutes. You can use the -last n[s|m|h|d] option in conjunction with the -to option to ensure the search period includes the specified timestamped records.

F.3.3 Output Options

Table F–4 describes the output options you can use to specify an output format. The default format is -text_short.

Table F–4 Output Options

Output Option	Description
-odl	Specify that the output should be in ODL format. This option outputs an ODL document without the enclosing LOG tags. The generated output is not a complete XML document.
-odl_complete	Specify that the output should be in ODL format and that a complete XML document should be generated

Table F–4 (Cont.) Output Options

Output Option	Description
`-text_short`	Specify that the output should be in a short text format including only the following fields: `TSTZ_ORIGINATING`, `COMPONENT_ID`, `MSG_TYPE`, `MODULE_ID`, `EXEC_CONTEXT_ID`, `MSG_TEXT`, and `SUPPL_DETAIL`. This is the default output format.
`-text`	Same as -text_short
`-text_full`	Specify that the output should be in full text format, including all message fields.
`-orderBy` *`orderByFieldList`*	Sort the result in the specified order. The *`orderByFieldList`* argument is a list of log record field names separated by spaces. The field names can have an optional suffix of `:asc` or `:desc` to specify ascending or descending order. The default sort order is ascending. `printlogs` sorts the result in memory. If the result is large, it could run out of memory. In this case, you must provide additional filtering options to reduce the number of records in the result.
`-count [`*`groupByFieldList`*`]`	Report only the record count. The *`groupByFieldList`* argument is an optional list of log record field names separated by spaces. If you supply this argument, `printlogs` reports the record count for each supplied field.

F.3.4 General Options

Table F–5 describes the general options you can use to obtain help, cause `printlogs` to loop, and disable optimization.

Table F–5 General Options

General Option	Description
`-help`	Print detailed help.
`-f`	Follow. When you use this option, `printlogs` will not return after printing the result. Instead, it will go on an infinite loop where it sleeps for a number of seconds (specified with the -sleep n option), and then checks each log again and prints any new records that satisfy the query predicate.
`-sleep` *`n`*	Set the sleep time, in seconds, for the -f option. The default value is 20 seconds.
`-notailopt`	Disable the "tail optimization" that is usually performed with the -last option.

F.4 Log Record Fields

The `printlogs` command automatically translates the contents of any log file that it reads to the Oracle Diagnostic Logging (ODL) format. The ODL log record fields can be used to create a query expression, or to specify a group-by or order-by field list. Each field must be referred to by its name, as described in Table F–6. Some of these fields are designated for future use, and currently are not used in any diagnostic messages generated by an Oracle Application Server.

Table F–6 Log Record Fields

Log Record Field Name	Description
`COMPONENT_ID`	The component that originated the message
`DETAIL_PATH`	A URL for additional information about the message
`DOWNSTREAM_COMPONENT_ID`	The component that the originating component is working with on the downstream (server) side
`EID.SEQ`	The sequence number that is associated with the error instance

Table F–6 (Cont.) Log Record Fields

Log Record Field Name	Description
EID.UNIQUE_ID	A global unique identifier of an error instance associated with the message. This identifier can be used to correlate error messages from different components.
EXEC_CONTEXT_ID.SEQ	The sequence number that is associated with the execution context
EXEC_CONTEXT_ID.UNIQUE_ID	A global unique identifier of the thread of execution in which the originating component participates. This identifier can be used to correlate messages from several components that may be involved in the same thread of execution.
HOST_ID	The name of the host where the message originates
HOST_NWADDR	The network address of the host where the message originates
HOSTING_CLIENT_ID	An identifier for the client or security group to which the message relates
MODULE_ID	An identifier of the module that originated the message
MSG_ARG	A list of arguments to be bound with the message text. The argument is a list of an optional name and value. *Note: This field is not currently supported.*
MSG_GROUP	The name of the group to which the message belongs
MSG_ID	A message number, or some other value, that uniquely identifies the message within the component
MSG_LEVEL	The level qualifies the message type, indicating the degree of severity of the message. The value is an integer from 1 (highest severity) to 32 (lowest severity).
MSG_TEXT	A descriptive text for the message
MSG_TYPE	The type of the message. The defined message types are: INTERNAL_ERROR, ERROR, WARNING, NOTIFICATION, and TRACE. The value UNKNOWN may be used when the type is not known.
ORG_ID	The organization that wrote the component that originated the message
PROCESS_ID	An identifier of the process or execution unit that generated the message. For Java processes, the value can also include a thread identifier.
SUPPL_DETAIL	Supplemental information about the message
TSTZ_NORMALIZED	Date and time when the message was generated, adjusted for time difference between the host on which the message was generated and the host of the common repository. This field is only used if the log record is being read from a database repository.
TSTZ_ORIGINATING	Date and time when the message was generated
UPSTREAM_COMPONENT_ID	The component that the originating component is working with on the upstream (client) side
USER_ID	The user whose execution context originated the message

F.5 Environment Variable

You can use an environment variable to pass information to `printlogs`. Table F–7 describes the environment variable in detail.

Table F–7 Environment Variable

Environment Variable	Description
ORACLE_PRINTLOGS_JVM_ARGS	Provide additional arguments to the JVM that runs `printlogs`. It is usually not necessary to provide additional JVM arguments, but this environment variable can be used in some situations, such as to set memory size, or provide additional properties to `printlogs`.

F.6 Examples

- To print records from all known logs in the last 10 minutes:

  ```
  printlogs
  ```

- To print records from all known logs in the last 10 minutes and follow:

  ```
  printlogs -f
  ```

  After reaching the end of all log files, `printlogs` will go into an infinite loop where it sleeps for 20 seconds, then reads and prints any new records that are added to the log files.

- To print records from all known logs in the specified Oracle home in the last 2 days, in ODL format:

  ```
  printlogs -home /private/orahome2 -last 7d -odl
  ```

- To print records that are timestamped between 14:00 and 14:05 hours:

  ```
  printlogs -last 100d -query TSTZ_ORIGINATING -from 2003-07-15T14:00:00-07:00
  -and TSTZ_ORIGINATING -to 2003-07-15T14:05:00-07:00
  ```

  In this example, assume that the specified time interval is more than 10 minutes before the current time. By default, `printlogs` searches logs generated in the last 10 minutes. Therefore, you need to use the `-last` option to increase the overall search length to include the timestamp interval. To save the trouble of calculating the amount of time to the timestamp interval, you can specify a very large value, such as `-last 100d`.

- To print records from OC4J logs that contain the word "exception" and are for the local Oracle home:

  ```
  printlogs -last 1d -query \( COMPONENT_ID -eq OC4J -and MODULE_ID -startswith
  home \) -and MSG_TEXT -contains exception
  ```

  Note: On the Windows platform the parentheses should not be escaped.

- To print records in the last 10 minutes, sorted in ascending order by component id, and in descending order by time:

  ```
  printlogs -orderBy COMPONENT_ID TSTZ_ORIGINATING:desc
  ```

- To print the number of records from all known logs in the last 10 minutes, grouping by component and message type:

  ```
  printlogs -count COMPONENT_ID MESSAGE_TYPE
  ```

- To print records in the last hour from `daemon_logs` and `dcmctl_logs`:

  ```
  cd ORACLE_HOME/dcm/logs
  printlogs -last 1h -logs daemon_logs dcmctl_logs
  ```

  Note that this example uses log file names relative to the current directory.

- To print records in the last 10 minutes from `ipm.log` and `ons.log`:

  ```
  printlogs -logs ORACLE_HOME/opmn/logs/ipm.log ORACLE_HOME/opmn/logs/ons.log
  ```

  Note that this example uses the full path to the log files and can be run from any directory.

G

Examples of Administrative Changes

This appendix provides examples of administrative changes that can be performed on an Oracle Application Server environment. It is a companion to Part V, "Backup and Recovery" in this book, and to the Disaster Recovery section in *Oracle Application Server High Availability Guide*.

It contains the following topics:

- How to Use This Appendix
- Examples of Administrative Changes (by Component)

G.1 How to Use This Appendix

Some administrative operations cause configuration changes to your Oracle Application Server environment. These are called **administrative changes**, and include deploying and undeploying applications, changing the topology, changing ports, creating and deleting users, and changing passwords. As an administrator, you should be aware when administrative changes occur, because you may need to back up your environment or perform some synchronization procedures.

This appendix provides examples of administrative changes, listed by component. You can use this as a guide for performing the following procedures:

- Backup and Recovery

  Oracle recommends you perform a backup after each administrative change to your environment. You can use this appendix to determine the types of administrative changes that require you to back up your environment.

  See Also: Part V, "Backup and Recovery"

- Disaster Recovery Synchronization Between the Primary and Standby Sites

  When you implement Disaster Recovery, you must update standby sites when you make an administrative change to your environment. You can use this appendix to determine the types of administrative changes that require you to update your standby sites.

  See Also: *Oracle Application Server High Availability Guide*

G.2 Examples of Administrative Changes (by Component)

Table G–1 provides examples of administrative changes, by component. Consult your component documentation to learn more about these operations.

Table G–1 Examples of Administrative Changes

Component	Examples of Administrative Changes
Delegated Administration Services	Manual edits to Oracle Delegated Administration Services configuration files, such as `das.properties`
Directory Integration and Provisioning	Directory Integration and Provisioning administrative and configuration operations, such as running the `odisrvreg` or `remtool` utilities (password management)
Distributed Configuration Management (DCM)	DCM administrative and configuration operations performed using the Application Server Control Console
	Manual edits to DCM configuration files
	DCM administrative and configuration operations using `dcmctl`, such as `configrepositoryssl`, `joincluster`, `joinfarm`, `leavecluster`, `leavefarm`, `repositoryrelocated`, `resetDCMcacheport`, `resethostinformation`, `restoreinstance`, and `set` operations
	DCM administrative and configuration operations performed using the `dcmctl` utility, such as deploying and undeploying applications and making configuration changes
Dynamic Monitoring Service (DMS)	DMS administrative and configuration operations performed using the Application Server Control Console
	Manual edits to DMS configuration files, such as `dms.conf`
Log Loader	Log Loader administrative and configuration operations performed using the Application Server Control Console
	Manual edits to Log Loader configuration files, such as `logloader.properties`, `logloader.xml`, and files in *`ORACLE_HOME`*`/diagnostics/config/registration`
Oracle Application Server Containers for J2EE (OC4J)	OC4J administrative and configuration operations performed using the Application Server Control Console
	Manual edits to OC4J configuration files
	OC4J administrative and configuration operations using the `dcmctl` utility, such as deploying and undeploying applications, and creating OC4J instances
Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider (JAZN)	JAZN administrative and configuration operations performed using the Application Server Control Console
	JAZN administrative and configuration operations performed using the `admintool` utility, such as adding and removing users, and changing roles, permissions, privileges, and passwords
Oracle Enterprise Manager 10*g* Application Server Control Console	Application server-wide or component-specific administrative and configuration operations performed using the Application Server Control Console, such as changing the `ias_admin` password, changing port numbers, deploying and undeploying applications, and operations that result in configuration file changes
Oracle HTTP Server	Oracle HTTP Server administrative and configuration operations performed using the Application Server Control Console, such as modifying the number of VMs and creating virtual hosts
	Manual edits to Oracle HTTP Server configuration files
	Oracle HTTP Server administrative and configuration operations using the `dcmctl` utility
Oracle Internet Directory	Oracle Internet Directory administrative and configuration operations, such as running the `oidpasswd` or `remtool` utilities (password management), and installing and removing components

Table G–1 (Cont.) Examples of Administrative Changes

Component	Examples of Administrative Changes
Oracle Process Manager and Notification Server (OPMN)	OPMN administrative and configuration operations performed using the Application Server Control Console
	Manual edits to OPMN configuration files, such as `opmn.xml`
	OPMN administrative and configuration operations using the `dcmctl` utility
Oracle Ultra Search	Manual edits to Oracle Ultra Search configuration files, such as `crawler.dat`, `data-sources.xml`, `truststore.dat`, and `ultrasearch.properties`
OracleAS Certificate Authority (OCA)	OCA administrative and configuration operations using the `ocactl` utility with the following options: `setpasswd`, `generatewallet`, `convertwallet`, `importwallet`, `revokecert`, `renewcert`, `updateconnection`, and `changesecurity`
	Using the administrative interface to enroll the OCA Web administrator
OracleAS Forms Services	OracleAS Forms Services administrative and configuration operations performed using the Application Server Control Console, such as operations on the "Forms/Configuration", "Forms/Environment Property", and "Forms/Overview" pages
OracleAS Portal	OracleAS Portal administrative and configuration operations performed using the Application Server Control Console
	OracleAS Portal administrative and configuration operations using the Administration screen in the Portal User Interface
	Manual edits to OracleAS Portal configuration files
	Running the `ptlconfig` script
	Running any Portal-specific scripts that modify the database-side configuration for Portal, for example, disabling OracleAS Web Cache or changing some background job frequencies in Portal
Oracle BPEL Process Analytics	Oracle BPEL Process Analytics administrative and configuration operations performed using the Application Server Control Console
OracleAS Reports Services	OracleAS Reports Services administrative and configuration operations performed using the Application Server Control Console, such as operations on the "Reports/Configuration" page
	Manual edits to OracleAS Reports Services configuration files
	When the Reports server receives a job insert or update, such as when adding a new job or moving a job from one queue to another. *Note: Oracle recommends that you perform backup and file synchronization more frequently when running OracleAS Reports Services.*
OracleAS Single Sign-On	OracleAS Single Sign-On administrative and configuration operations performed using the Application Server Control Console, such as changing the `ORASSO` schema password
	Configuration changes such as adding or removing an OracleAS Single Sign-On middle-tier instance, changing OracleAS Single Sign-On to use SSL, and performing Windows Native Authentication configuration changes
OracleAS Web Cache	OracleAS Web Cache configuration properties performed using options from the Administration tab (**Web Cache Home** page -> **Administration** tab) in the Application Server Control Console.
OracleAS Wireless	OracleAS Wireless administrative and configuration operations performed using the Application Server Control Console, such as deploying and undeploying applications, changing ports, making changes to groups or users, and changing configuration parameters

H

Supplementary Procedures for Configuring LDAP-Based Replicas

This appendix contains auxiliary procedures that are referred to in Chapter 9 and Chapter 12.

It contains the following topics:

- About LDAP-Based Replicas
- Installing and Setting Up an LDAP-Based Replica

H.1 About LDAP-Based Replicas

This section describes how to install and configure an LDAP-based replica. It contains the following topics:

- What Is an LDAP-Based Replica?
- How Is the LDAP-Based Replica Used for Changing Infrastructure Services?

H.1.1 What Is an LDAP-Based Replica?

Oracle Internet Directory replication is the process of copying and maintaining the same data (or naming context) on multiple directory servers. Simply put, replication is a means of having two identical directories that contain the same information. One directory is called the master (or supplier). This directory contains the master copy of the naming context. The other directory is called the replica (or consumer). The master supplies replication updates to the replica, which keeps the master and replica in sync.

There are different types of replicas. This procedure uses an LDAP-based replica, which means the protocol for transferring data between the master and the replica is LDAP.

> **See Also:** *Oracle Internet Directory Administrator's Guide* for more information on directory replication and LDAP-based replicas

For the purposes of this procedure, the master and replica directories are part of a larger environment that includes the Identity Management installations that contain the directories, and the Metadata Repositories that support them. This is called the LDAP-based Replica Environment, and it contains the following:

Master—The Identity Management installation containing the Oracle Internet Directory that holds the master copy of the naming context. It supplies replication updates to the replica.

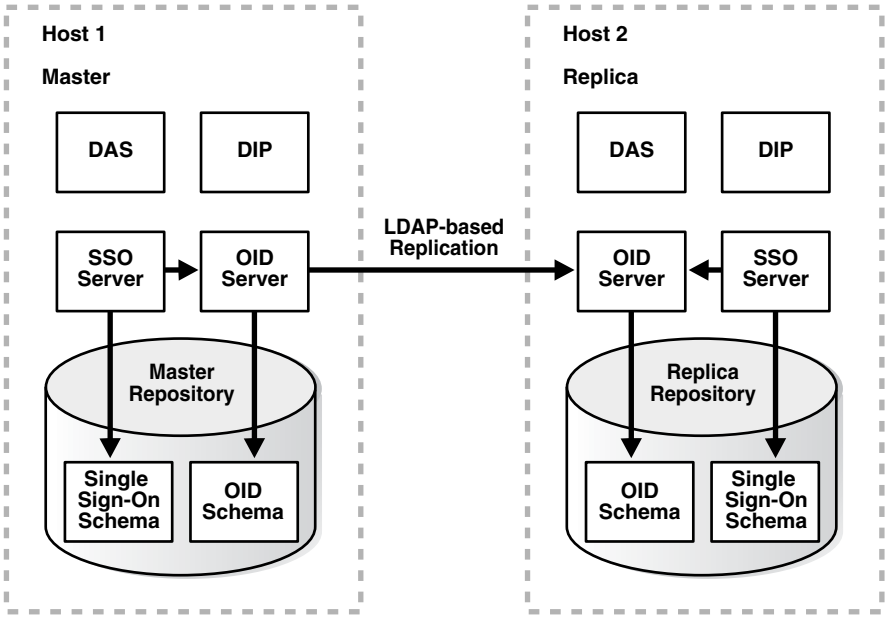
Master Repository—The Metadata Repository that the master uses to store its Identity Management schemas.

Replica—The Identity Management installation containing the replicated Oracle Internet Directory.

Replica Repository—The Metadata Repository that the replica uses to store its Identity Management schemas.

Figure H–1 illustrates the LDAP-based replica environment.

Figure H–1 LDAP-Based Replica Environment



H.1.2 How Is the LDAP-Based Replica Used for Changing Infrastructure Services?

Typically, an LDAP-based replica is used to provide high availability and improved performance for directory users. For the purposes of changing Infrastructure services, the LDAP-based Replica is used as follows:

- For Section 9.4, "Moving Identity Management to a New Host", the LDAP-based replica is created as a way of moving Identity Management from one host to another. The Master is the original Identity Management installation, and the Replica is the new Identity Management installation. In this case, replication is used to create an identical copy of the original Identity Management on a new host. You can then change your middle tiers from the old Identity Management (Master) to the new Identity Management (Replica) and discard the Master.
- For Chapter 12, the replica is used to create a test to production environment. The Master is the production Identity Management, and the Replica is the test Identity Management. When you are ready to merge your test environment into your production environment, you can migrate data from your test Identity Management (Replica) to your production Identity Management (Master) and change your middle-tiers from the test Identity Management to the production Identity Management. You can then discard the test Identity Management or continue to use it for testing.

H.2 Installing and Setting Up an LDAP-Based Replica

This section describes how to install and set up an LDAP-based replica environment.

H.2.1 Things to Know Before You Start

You should be aware of these important items before you start the procedure:

- This procedure uses a single Infrastructure Oracle home that contains Identity Management and the Metadata Repository. However, you can split the Infrastructure installation so that Identity Management is in one Oracle home and the Metadata Repository is in another Oracle home. You can also distribute the Identity Management components (OracleAS Single Sign-On, Oracle Internet Directory, Delegated Administration Services, Directory Integration and Provisioning) across different hosts. If you do this, perform the operations on each component in their respective Oracle homes.
- The replica always uses port 389 for the non-SSL Oracle Internet Directory port, and 636 for the SSL Oracle Internet Directory port, regardless of what is reported by Oracle Universal Installer, or printed in *ORACLE_HOME*`/install/portlist.ini`. Make sure no other processes are using ports 389 and 636 on the replica host before you start the procedure.
- Make sure you use the `ldapsearch` and `ldapmodify` commands that are in *ORACLE_HOME*`/bin`. (Some operating systems ship their own version of these commands—do not use those.)
- These procedures use the `remtool` and `oidpasswd` commands. The messages returned by these commands are in UTF-8 encoding and are unreadable in most non-English environments. To work around this, set the `NLS_LANG` environment variable to `american_america.`*`character_set`* before running these commands. Most character sets (for example, `US7ASCII`) will work.

  See Also: *Oracle Application Server Globalization Guide*

- Make sure the `ORACLE_HOME` and `ORACLE_SID` environment variables are set. This applies to all platforms.

H.2.2 Procedure

This section contains the procedure for setting up an LDAP-based replica. It contains the following tasks:

- Task 1: Obtain the Master and Master Repository
- Task 2: Install Middle-Tier Instances (Optional)
- Task 3: Install and Configure the Replica

Task 1: Obtain the Master and Master Repository

Most likely, you already have your Master and Master Repository.

- If you are following the procedure in Section 9.4, "Moving Identity Management to a New Host", the Master and Master Repository are the installations you would like to move to a new host, and the LDAP-based replica will be the relocated installations.
- If you are following the procedure in Chapter 12, the Master and Master Repository are your production environment, and the replica will be your test environment.

If you are starting from scratch, you can install a Master and Master Repository as follows:

1. Install Oracle Application Server using Oracle Universal Installer.
2. Choose the Infrastructure Installation.
3. Choose to install Identity Management and OracleAS Metadata Repository.
4. Choose to configure the following components: Oracle Internet Directory, OracleAS Single Sign-On, Delegated Administration Services, and Directory Integration and Provisioning.

Task 2: Install Middle-Tier Instances (Optional)

Most likely, you already have middle-tier instances using the Master for Identity Management services. This is fine, and, if desired, you can install and configure additional instances to use the Master now, or at the end of this procedure after you have configured the Replica, or both.

These middle-tier instances can use the Master Repository for their product metadata, or they can use a different repository.

Task 3: Install and Configure the Replica

You can install and configure the Replica using Oracle Universal Installer. Be sure to install the Replica on a different host than the Master.

> **See Also:** *Oracle Application Server Installation Guide* for information on installing an Oracle Internet Directory replica

When the installation has finished, replication is configured and all components are up and running. You can return to the main procedure from where you started (either Section 9.4, "Moving Identity Management to a New Host" or Chapter 12).

I

Viewing Oracle Application Server Release Numbers

This appendix describes how to view Oracle Application Server release numbers.

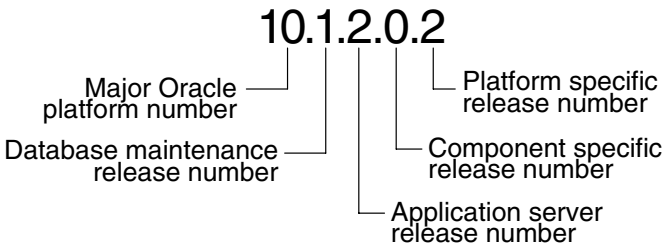
It contains the following topics:

> **Note:** Oracle recommends you keep a log of all interim patches applied to your Oracle Application Server installations.

I.1 Release Number Format

To understand the release level nomenclature used by Oracle, examine the example of an Oracle release number shown in Figure I–1.

Figure I–1 Example of an Oracle Release Number



In Figure I–1, each digit is labeled:

- Major Oracle Platform Number

  This is the most general identifier. It represents a major new edition (or version) of an application, such as Oracle database server or Oracle Application Server, and indicates that the release contains significant new functionality.

- Database Maintenance Release Number

This digit represents a maintenance release level. Some new features may also be included.

- Application Server Release Number

  This digit reflects the release level of Oracle Application Server.

- Component Specific Release Number

  This digit identifies a release level specific to a component. Different components can have different numbers in this position depending upon, for example, component patch sets or interim releases.

- Platform Specific Release Number

  This digit identifies a platform-specific release.

I.2 Viewing Oracle Application Server Installation Release Numbers

All Oracle Application Server installations have a release number. This number is updated when you apply a patch set release or upgrade the installation.

You can view the release number of an Oracle Application Server installation using Oracle Universal Installer, as follows:

1. Launch Oracle Universal Installer:

   ```
   (UNIX) ORACLE_HOME/oui/bin/runInstaller.sh
   (Windows) ORACLE_HOME\oui\bin\runInstaller.bat
   ```

2. Click **Installed Products** to open the Inventory Page.
3. In the Inventory Page, expand **Oracle Homes**. You will see entries for all installations on your host.
4. Expand the Oracle home entry for the installation you are interested in.
5. You will see an entry with the release number for your original installation, followed by entries for any patch sets that have been applied.

I.3 Viewing Component Release Numbers

All Oracle Application Server components have a release number and many contain services that have release numbers. These numbers *may* be updated when you apply a patch set release or upgrade the installation.

You can view the release number of components and their services in the following ways:

- On the Filesystem
- Using Oracle Universal Installer

On the Filesystem

You can view component release numbers as follows on UNIX:

```
cd ORACLE_HOME/inventory
ls -d Components*/*/*
```

Using Oracle Universal Installer

If you installed Oracle Application Server using Oracle Universal Installer, you can view component release numbers as follows:

1. Launch Oracle Universal Installer:

   ```
   (UNIX) ORACLE_HOME/oui/bin/runInstaller.sh
   (Windows) ORACLE_HOME\oui\bin\runInstaller.bat
   ```

2. Click **Installed Products** to open the Inventory Page.
3. In the Inventory Page, expand **Oracle Homes**. You will see entries for all installations on your host.
4. Expand the Oracle home entry for the installation you are interested in.
5. You will see an entry with the release number for your original installation, followed by entries for any patch sets that have been applied.
6. Expand the initial entry to view the component release numbers at installation time. If you have subsequent patch set entries, expand them to see the component release numbers updated for each patch set.

I.4 Viewing Oracle Internet Directory Release Numbers

Oracle Internet Directory has a server release number, which is the version of the binaries. It also has schema and context versions. All of these numbers correspond to the Oracle Application Server installation release number through the third digit. These numbers *may* be updated when you apply a patch set release or upgrade the installation.

Viewing the Oracle Internet Directory Server Release Number

The Oracle Internet Directory server release number is the version of the binaries. You can view the Oracle Internet Directory server release number as follows:

1. Make sure the ORACLE_HOME environment variable is set.
2. Run the following command:

   ```
   (UNIX) ORACLE_HOME/bin/oidldapd -version
   (Windows) ORACLE_HOME\bin\oidldapd -version
   ```

Viewing the Oracle Internet Directory Schema and Context Versions

You can view the Oracle Internet Directory schema and context versions in this file:

```
(UNIX) ORACLE_HOME/ldap/schema/versions.txt
(Windows) ORACLE_HOME\ldap\schema\versions.txt
```

The contents of this file are kept up-to-date, however, you can also query the schema and context release from Oracle Internet Directory, just to be sure.

To view the schema version:

1. Make sure the ORACLE_HOME environment variable is set.
2. Run the following command:

   ```
   ldapsearch -h oid_host -p oid_port -D "cn=orcladmin"
   -w orcladmin_password -b "cn=base,cn=oracleschemaversion" -s
   base "objectclass=*" orclproductversion
   ```

   The output will be in this form:

   ```
   cn=BASE,cn=OracleSchemaVersion
   orclproductversion=90500
   ```

To view the context version:

1. Make sure the ORACLE_HOME environment variable is set.
2. Run the following command:

```
ldapsearch -h oid_host -p oid_port -D "cn=orcladmin"
 -w orcladmin_password -b "cn=oraclecontext" -s
base "objectclass=*" orclversion
```

The output will be in this form:

```
cn=oraclecontext
orclversion=101200
```

I.5 Viewing Metadata Repository Release Numbers

Metadata Repositories have the following release numbers:

- Database release number

  This is the Oracle Database 10*g* database release number.

- Metadata Repository Container release number

  This is the release number for the Metadata Repository. The number is equal to the Oracle Application Server installation release number.

- Schema release numbers

  The Oracle Application Server schemas in the Metadata Repository have release numbers. These numbers do not necessarily correspond to Oracle Application Server release numbers or database release numbers.

Viewing the Database Release Number

The Metadata Repository is an Oracle Database 10*g* database that has a release number. This number is updated when you apply a patch set release or upgrade the database.

You can view the Metadata Repository release number using SQL*Plus as follows (you can be connected to the database as any user to issue these commands):

```
SQL> COL PRODUCT FORMAT A40
SQL> COL VERSION FORMAT A15
SQL> COL STATUS FORMAT A15
SQL> SELECT * FROM PRODUCT_COMPONENT_VERSION;

PRODUCT                                  VERSION         STATUS
---------------------------------        --------------  ----------------
NLSRTL                                   10.1.4.0.2      Production
Oracle Database 10g Enterprise Edition   10.1.4.0.2      64bi
PL/SQL                                   10.1.4.0.2      Production
TNS for Solaris:                         10.1.4.0.2      Production
```

Viewing Metadata Repository Container and Schema Release Numbers

You can view the Metadata Repository Container release number, as well as schema release numbers, using SQL*Plus as follows (you must log in as a user with SYSDBA privileges):

```
SQL> COL COMPONENT_NAME FORMAT A35
SQL> COL ID FORMAT A15
SQL> COL VERSION FORMAT A15
```

```
SQL> SELECT * FROM IAS_VERSIONS;

COMPONENT_NAME                    ID              VERSION
--------------------------------  --------------  -------------------
Metadata Repository Container     mrc             10.1.0.4.0
Oracle Ultrasearch                ultrasearch     10.1.0
```

`IAS_VERSIONS` is a public synonym to a view owned by the `INTERNET_APPSERVER_REGISTRY` user. If the preceding query returns an error, it may be because:

- There was an error in seeding one or more components
- Not all of the components whose underlying tables are read by the view are present in the database

Either case indicates that the database is not properly seeded to be a Metadata Repository.

To get the same result by querying the underlying table:

```
SQL> SELECT * FROM INTERNET_APPSERVER_REGISTRY.SCHEMA_VERSIONS;
```

I.6 Using the OPatch Utility

The OPatch utility is a tool that allows the application and rollback of interim patches to Oracle products, such as Oracle Application Server. For the latest information about the opatch utility, and to check for updates, refer to Oracle MetaLink at

`http://www.oracle.com/support/metalink/index.html`

I.6.1 Requirements

The OPatch utility has the following requirements:

- Perl environment, included with Oracle Application Server or downloaded with a patch set.
- The Oracle home environment variable (ORACLE_HOME) must point to a valid Oracle home directory and match the value used during installation of the Oracle home directory.
- If the `-invPtrLoc` command-line argument was used during installation, then it must be used when using the OPatch utility. Oracle recommends the use of the default central inventory for a platform.
- The `jar`, `java`, `ar`, `cp`, and `make` commands must be available in the PATH statement. The commands are not available for all platforms.
- The library path must be set correctly for Oracle Real Application Clusters environments. Refer to the FAQ document in the `opatch/doc` directory for additional information.

> **See Also:** For the latest information about the OPatch utility, and to check for updates, refer to Oracle*MetaLink* at
>
> `http://www.oracle.com/support/metalink/index.html`

I.6.2 Running the OPatch Utility

The OPatch utility is located in the `ORACLE_HOME/OPatch` directory. It is run with options and command-line arguments. The following command shows the syntax for the OPatch utility:

```
path_to_opatch/opatch option -command_line_arguments
```

In the preceding command, the following variables are used:

- *`command_line_arguments`*: The command-line arguments for the option. Values are described in the following sections.
- *`option`*: The OPatch option. Values are described in the following table:

Option	Description
apply	Installs an interim patch. Refer to Section I.6.2.1 for more information.
lsinventory	Lists what is currently installed on the system. Refer to Section I.6.2.2 for more information.
query	Queries a given patch for specific details. Refer to Section I.6.2.3 for more information.
rollback	Removes an interim patch. Refer to Section I.6.2.4 for more information.
version	Prints the current version of the patch tool. Refer to Section I.6.2.5 for more information.

To view additional information for any option, use the following command:

```
path_to_OPatch/opatch option -help
```

If using Perl, then use the following command:

```
perl opatch.pl option -help
```

I.6.2.1 apply Option

The `apply` option applies an interim patch to a specified Oracle home. The ORACLE_HOME environment variable must be set to the Oracle home to be patched. The following syntax is used for this option:

```
path_to_opatch/opatch apply [patch_location] [-delay (value)] [-force] \
[-invPtrLoc (path)] [-jdk (location)] [-jre (location)] [-local] \
[-minimize_downtime] [-no_bug_superset] [-no_inventory] \
[-oh (Oracle home location)] \
[-post (options to be passed into post) [-opatch_post_end]]\
[-pre (options to be passed into pre) [-opatch_pre_end]] \
[-retry (value)] [-silent] [-verbose]
```

The following table lists the command-line arguments available for use with the `apply` option:

Argument	Description
delay	Specifies how many seconds to wait before attempting to lock the inventory in the case of a previous failure.
force	Removes conflicting patches from the system. If a conflict exists which prevents the patch from being applied, then the `-force` command-line argument can be used to apply the patch.

Argument	Description
invPtrLoc	Specifies the location of the `oraInst.loc` file. This command-line argument is needed when the `-invPtrLoc` argument was used during installation. Oracle recommends the use of the default central inventory for a platform.
jdk	Specifies the location of a particular JDK (jar) to use instead of the default location under the Oracle home directory.
jre	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory.
local	Specifies that the OPatch utility patch the local node and update the inventory of the local node. It does not propagate the patch or inventory update to other nodes. This command-line argument can be used on Oracle Real Application Clusters environments and non-clustered environments. If an entire cluster is shutdown before patching, then this argument can be used for non-rolling patches.
minimize_downtime	Specifies the order of nodes to be patched by the OPatch utility. This command-line argument only applies to Oracle Real Application Clusters environments. It cannot be used with the `-local` command-line argument or a rolling patch.
no_bug_superset	Specifies to error out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory.
no_inventory	Bypasses the inventory for reading and updates. This command-line argument cannot be used with the `-local` command-line argument. This command-line argument puts the installation into an unsupported state.
oh	Specifies the Oracle home directory to use instead of the default.
opatch_post_end	Marks the end of the `post` options. This command-line argument is used with the `post` command-line argument. If this argument is not used, then everything after `post` is passed into `post`.
opatch_pre_end	Marks the end of the `pre` options. This command-line argument is used with the `pre` command-line argument. If this argument is not used, then everything after `pre` is passed into `pre`.
post	Specifies the parameters to be passed inside the `post` script besides the standard parameters.
pre	Specifies the parameters to be passed inside the `pre` script besides the standard parameters.
retry	Specifies how many times the OPatch utility should try when there is an inventory lock failure.
patch_location	Specifies the directory of the interim patch. This should be a directory with the same name as the patch.
silent	Suppresses user interaction, and defaults any answers to "yes."
verbose	Prints output to the screen as well as to the log file.

Note: If a patch consists of SQL changes, then they are only staged. Follow the instructions included with the patch to apply the patch manually on the affected instances. For some products, such as OracleAS Portal, the SQL application may be implemented as a post-staging action by the tool. These patches cannot be rolled back.

I.6.2.2 lsinventory Option

The `lsinventory` option reports what has been installed on the system for a particular Oracle home directory, or for all installations. The following syntax is used for this option:

```
path_to_opatch/opatch lsinventory [-all] [-detail] [-invPrtLoc (path)] \
[-jre (location)] [-oh (Oracle home location)]
```

The following table lists the command-line arguments available for use with the `lsinventory` option:

Argument	Description
all	Reports the name and installation directory for each found Oracle home directory.
detail	Reports the installed products and other details. This command-line argument cannot be used with the `-all` command-line argument.
invPtrLoc	Specifies the location of the `oraInst.loc` file. This command-line argument is needed when the `invPtrLoc` command-line argument was used during installation. Oracle recommends the use of the default central inventory for a platform.
jre	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory.
oh	Specifies the Oracle home directory to use instead of the default directory.

The following is a sample output of `opatch lsinventory -detail`:

```
ORACLE_HOME      LOCATION
-----------      --------
Home1            /private/phi_local/OraHome1
   There is no Interim Patch
Home2            /private/phi_local/OraHome2
   There is no Interim Patch
Home3           /private/phi_local/OraHome6
   Installed Patch List:
    ====================
    1) Patch 20 applied on Mon Jul 11 15:53:51 PDT 2005
       [ Base Bug(s): 21  ]
    2) Patch 80 applied on Fri Jul 01 16:15:52 PDT 2005
       [ Base Bug(s): 80 81  ]
```

I.6.2.3 query Option

The `query` option queries a specific patch for specific details. It provides information about the patch and the system being patched. The following syntax is used for this option:

```
path_to_opatch/opatch query [-all] [-get_base_bug] [-get_component] \
[-get_date] [-get_os] [-get_system_change] [-is_rolling]
```

The following table lists the command-line arguments available for use with the `query` option:

Argument	Description
all	Retrieves all information about a patch. This is equivalent to setting all command-line arguments.

Argument	Description
get_base_bug	Describes the base bugs fixed by a patch.
get_component	Describes the Oracle components, optional or required, for a patch.
get_date	Provides the build date of a patch.
get_os	Provides the operating system description supported by a patch.
get_system_change	Describes the changes that will be made to the system by a patch. This command-line argument is not available.
is_rolling	Specifies if the patch is a rolling patch for Oracle Real Application Clusters. The set of patches need not be applied to the whole cluster at the same time. The patches can be applied to a select set of nodes at a time.

I.6.2.4 rollback Option

The `rollback` option removes a specific interim patch from the appropriate Oracle home directory. The following syntax is used for this option:

```
path_to_opatch/opatch rollback -id patch_id -ph (patch directory) \
[-delay] (value) [-invPtrLoc (path)] [-jdk (location)] [-jre (location)]\
[-local] [-oh (Oracle home location)] \
[-post (options to be passed into post) [-opatch_post_end]] \
[-pre (options to be passed into pre) [-opatch_pre_end]] [-retry (value)] \
[-silent] [-verbose]
```

The following table lists the command-line arguments available for use with the `rollback` option:

Argument	Description
delay	Specifies how many seconds the OPatch utility should wait before attempting to lock inventory again, if the `-retry` command-line argument is used with the `apply` option.
id	Indicates the patch to be rolled back. Use the `-lsinventory` option to display all patch identifiers. To successfully rollback a patch, the patch identifier must be supplied.
invPtrLoc	Specifies the location of the `oraInst.loc` file. This command-line argument is needed when the `-invPtrLoc` command-line argument was used during installation. Oracle recommends the use of the default central inventory for a platform.
jdk	Specifies the location of a particular JDK (jar) to use instead of the default location under the Oracle home directory.
jre	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory.
local	Specifies that the OPatch utility patch the local node and update the inventory of the local node. It does not propagate the patch or inventory update to other nodes. This command-line argument can be used on Oracle Real Application Clusters environments and non-clustered environments. If an entire cluster is shutdown before patching, then this argument can be used for non-rolling patches.
oh	Specifies the Oracle home directory to use instead of the default directory.
opatch_post_end	Marks the end of the `post` options. This command-line argument is used with the `post` command-line argument. If this argument is not used, then everything after `post` is passed into `post`.

Argument	Description
opatch_pre_end	Marks the end of the `pre` options. This command-line argument is used with the `pre` command-line argument. If this argument is not used, then everything after `pre` is passed into `pre`.
ph	Specifies the valid patch directory area. The utility will use the command types found in the patch directory to identify which commands are used for the current operating system.
post	Specifies the parameters to be passed inside the `post` script besides the standard parameters.
pre	Specifies the parameters to be passed inside the `pre` script besides the standard parameters.
retry	Specifies how many times the OPatch utility should try in case of an inventory lock failure.
silent	Suppresses user interaction, and defaults any answers to "yes."
verbose	Prints output to the screen as well as to the log file.

I.6.2.5 version Option

The `version` option shows the current version number of the OPatch utility. The following syntax is used for this option:

```
path_to_opatch/opatch version
```

J

Troubleshooting Oracle Application Server

This appendix provides information on how to troubleshoot problems that you might encounter when using Oracle Application Server. It contains the following topics:

- Diagnosing Oracle Application Server Problems
- Common Problems and Solutions
- Troubleshooting Application Server Control
- Need More Help?

See Also:

- Chapter 18, "Troubleshooting SSL" for information about troubleshooting SSL.
- Chapter 23, "Troubleshooting the Backup and Recovery Tool" for specific information about troubleshooting the Backup and Recovery Tool

J.1 Diagnosing Oracle Application Server Problems

Oracle Application Server components generate log files containing messages that record all types of events, including startup and shutdown information, errors, warning messages, access information on HTTP requests, and additional information. The log files can be used to identify and diagnose problems. See Chapter 5, "Managing Log Files" for more information about log files.

J.2 Common Problems and Solutions

This section describes common problems and solutions. It contains the following topics:

- Oracle Application Server Infrastructure Instance Will Not Start
- Cannot Reset Administrator (ias_admin) Password
- Cannot Restore Backup to a Different Host
- Application Performance Impacted by Garbage Collection Pauses
- Application Server Returns Connection Refused Errors
- Oracle HTTP Server Unable to Start Due to Port Conflict
- Machine Overloaded by Number of HTTPD Processes
- Oracle Application Server Process Does Not Start

J.2.1 Oracle Application Server Infrastructure Instance Will Not Start

The Oracle Application Server Infrastructure will not start.

Problem

Some common symptoms and likely causes of this problem are:

- `opmnctl` startall/stopall is in a hung state. This may indicate that the listener or database is down.
- OPMN startup fails or is in an unstable state and `opmnctl` startall/stopall is in a hung state. This may indicate that the machine has run out of memory.
- You are receiving a missing component error. This usually indicates that entries are missing in the opmn.xml file or someone has incorrectly edited the file.
- Oracle Internet Dependency failed. This occurs when the correct order of starting and stopping is not followed.

Solutions

The following list provides solutions to problems in the same order as the symptoms listed earlier:

- Ensure that the database and listener are running.
- Ensure that the machine's memory meets the memory requirements listed in the *Oracle Application Server Installation Guide*.
- Check the opmn.xml file to see if the missing component is correctly entered in the file.
- The correct order in which to start Infrastructure components is:
  1. Oracle Database Server Net Listener
  2. Metadata Repository
  3. Identity Management
  4. Application Server Control Console

> **Note:** See Section 3.2.1, "Starting OracleAS Infrastructure" for instructions on how to start an Infrastructure.

J.2.2 Cannot Reset Administrator (ias_admin) Password

For information on resetting the `ias_admin` password, see Section J.3.1.1, "Resetting the Administrator (ias_admin) Password".

J.2.3 Cannot Restore Backup to a Different Host

For information on restoring a backup to a different host, see Section 22.2.3, "Restoring an Infrastructure to a New Host".

J.2.4 Application Performance Impacted by Garbage Collection Pauses

Application performance slows or application is unresponsive.

See the section "Application Performance Impacted by Garbage Collection Pauses" in the *Oracle Application Server Containers for J2EE User's Guide* for information about the causes and solutions to this problem.

J.2.5 Application Server Returns Connection Refused Errors

In load conditions (for example, when the number of users concurrently connecting to the application server increases significantly in a short time), the server may respond with the following error message:

```
IOException in sending request - Connection refused
```

Problem

If the number of simultaneous users is increasing, the server may be utilizing the maximum Oracle HTTP Server child processes allowable to service requests.

Solution

You may need to increase the MaxClients directive for the Oracle HTTP Server. The MaxClients directive specifies a limit on the number of clients who can simultaneously connect.

Use one of the following methods to determine if this is the case:

- Search the Oracle HTTP Server error log file for the following message:

  ```
  server reached MaxClients setting, consider raising the MaxClients setting
  ...
  ```

  By default, the error log file is located in:

  ```
  (UNIX) ORACLE_HOME/Apache/Apache/logs/error_log
  (Windows) ORACLE_HOME\Apache\Apache\logs\error_log
  ```

- Interactively monitor child process activity using the metrics in the Application Server Control Console. In particular, view the following in the HTTP_Server Home page:
  - In the Status section, **Active Connections,** which shows the number of clients currently executing HTTP requests.
  - In the Response and Load section, **Active Requests,** which shows the total number of active requests currently being processed.
- Look at the information provided by mod_status. The mod_status module provides an HTML page that shows the current server statistics. Check to see if all the processes are busy. (By default, Mod_status is enabled for localhost access only.) For more information on mod_status, see:

  http://httpd.apache.org/docs/mod/mod_status.html

In addition, consider increasing the maximum queue length for pending connections (the ListenBackLog directive) and consider the impact of persistent connections (the KeepAlive directive).

For more information about the Oracle HTTP Server directives and how to change their values, see the *Oracle HTTP Server Administrator's Guide*. For more information about tuning Oracle HTTP Server processes, see the *Oracle Application Server Performance Guide*.

J.2.6 Oracle HTTP Server Unable to Start Due to Port Conflict

You can get the following error if Oracle HTTP Server is unable to start due to port conflict:

```
[crit] (98) Address already in use: make_sock: could not bind to port 7778
```

See the section "Oracle HTTP Server Unable to Start Due to Port Conflict" in the Troubleshooting Oracle HTTP Server appendix of the *Oracle HTTP Server Administrator's Guide* for information about the cause and solution to this problem.

J.2.7 Machine Overloaded by Number of HTTPD Processes

When there are too many httpd processes running on a machine, the response time plummets.

See the section "Machine Overloaded by Number of HTTPD Processes" in the Troubleshooting Oracle HTTP Server appendix of the *Oracle HTTP Server Administrator's Guide* for information about the cause and solution to this problem.

J.2.8 Oracle Application Server Process Does Not Start

You are not able to start an Oracle Application Server process using OPMN.

See Section A.1.1, "Oracle Application Server Process Does Not Start" in the *Oracle Process Manager and Notification Server Administrator's Guide* for information about the causes and solutions to this problem.

J.2.9 OPMN Start Up Consumes CPU Processing Capability

On some computers, when OPMN starts up, it consumes large amounts of CPU processing capability.

See Section A.1.15, "OPMN Start Up Consumes CPU Processing Capability" in the *Oracle Process Manager and Notification Server Administrator's Guide* for information about the causes and solutions to this problem.

J.2.10 OPMN Cannot Start

OPMN cannot start. This may be caused by a corrupt `opmn.xml` file.

See Section B.1.1, "OPMN Cannot Start" in the *Distributed Configuration Management Administrator's Guide* for information about the causes and solutions to this problem.

J.2.11 DCM Daemon Cannot Start

The DCM daemon does not start.

See Section B.1.2, "DCM Daemon Cannot Start " in the *Distributed Configuration Management Administrator's Guide* for information about the causes and solutions to this problem.

J.2.12 DCM Unable to Connect to the Directory

DCM returns ADMN-100999 and the base exception is "Unable to connect to Directory."

See Section B.1.8, "Unable to Connect to the Directory" in the *Distributed Configuration Management Administrator's Guide* for information about the causes and solutions to this problem.

J.2.13 DCM Cannot Access the Infrastructure Database

DCM returns ADMN-202026 and the base exception is "Unable to connect to Directory."

See Section B.1.9, "Cannot Access the Infrastructure Database" in the *Distributed Configuration Management Administrator's Guide* for information about the causes and solutions to this problem.

J.2.14 OracleAS Web Cache Fails to Initialize or Restart a Managed Process

OracleAS Web Cache fails to initialize or restart a managed process.

Problem

You might receive, in the OracleAS Web Cache event log, the following errors when OracleAS Web Cache fails to initialize or fails to restart a managed process:

- Error Restarting Web Cache. Reason Web Cache failed to restart a managed process after the maximum retry limit
- The following errors:

```
[25/Nov/2004:19:12:40 +0000] [notification 9403] [ecid: -] Maximum number of
file/socket descriptors set to 950.
[25/Nov/2004:19:12:40 +0000] [notification 13002] [ecid: -] Maximum allowed
incoming connections are 700
[25/Nov/2004:19:12:40 +0000] [alert 13305] [ecid: -] Failed to assign port
7777: Address already in use
[25/Nov/2004:19:12:40 +0000] [alert 9707] [ecid: -]  Failed to start the
server.
[25/Nov/2004:19:12:40 +0000] [alert 9609] [ecid: -] The server process could
not initialize.
[25/Nov/2004:19:12:40 +0000] [notification 9610] [ecid: -]  The server is
exiting.
```

Solution

Check to see if the assigned port number is occupied by executing the following command:

```
netstat -a |grep "port number"
```

Also, check to see if the port number is less than 1024. If the port number is less than 1024, but it can be set to a higher number, set the port number to a number greater than 1024.

If the listen port number must be set to less than 1024 (typical setting for SSL listen ports), then, on UNIX, you must enable the Oracle Application Server instance as root. For instructions on setting the port number to less than 1024, please refer to Section 4.3.5.1, "Changing the OracleAS Web Cache Listen Ports".

J.2.15 Browser Displaying a Page Not Displayed Error

Browsers return an error saying that a page cannot be displayed.

See Section E.1.7, "Browser Displaying a Page Not Displayed Error" in the *Oracle Application Server Web Cache Administrator's Guide* for information about the causes and solutions to this problem.

J.2.16 Unable to Access OracleAS Portal

You are not able to access your portal instance. For example, pages are not displayed, or you get an "HTTP 503 Service Unavailable" error when you try to access OracleAS Portal.

See Section K.1.1, "Unable to Access OracleAS Portal" in the *Oracle Application Server Portal Configuration Guide* for information about the causes and solutions to this problem.

J.2.17 Unable to Log into OracleAS Portal

You can access the public home page but are unable to log in. Common symptoms of this problem are the following:

- The login page does not appear after you click Login.
- You get an error after you enter your credentials on the OracleAS Single Sign-On login page.
- You get errors on OracleAS Portal pages after you have been authenticated.

See Section K.1.2, "Unable to Log In to OracleAS Portal" in the *Oracle Application Server Portal Configuration Guide* for information about the causes and solutions to this problem.

J.2.18 Oracle Internet Directory Server Does Not Start

Either `oidctl` or `opmnctl` fails to start an Oracle Internet Directory server instance.

See Section J.1.11.1, "Oracle Internet Directory Server Does Not Start" in the *Oracle Internet Directory Administrator's Guide* for information about the causes and solutions to this problem.

J.2.19 Poor LDAP Search Performance

LDAP search performance is poor.

See Section J.1.5.1, "Poor LDAP Performance" in the *Oracle Internet Directory Administrator's Guide* for information about the causes and solutions to this problem.

J.2.20 Authentication Failed

Users may see an Authentication Failed error after logging in to OracleAS Single Sign-On.

See Section A.1.4, "Authentication Failed" in the *Oracle Application Server Single Sign-On Administrator's Guide* for information about the causes and solutions to this problem.

J.2.21 Logging into OracleAS Single Sign-On Takes a Long Time

Logging into OracleAS Single Sign-On might take a long time depending on your configuration.

See Section A.2.1, "Logging into OracleAS Single Sign-On" in the *Oracle Application Server High Availability Guide* for information about the causes and solutions to this problem.

J.2.22 Standby Site Not Synchronized

In the OracleAS Disaster Recovery standby site, you may find that the site's OracleAS Metadata Repository is not synchronized with the OracleAS Metadata Repository in the primary site.

See Section A.3.1, "Standby Site Not Synchronized" in the *Oracle Application Server High Availability Guide* for information about the causes and solutions to this problem.

J.2.23 Failure to Bring Up Standby Instances After Failover or Switchover

Standby instances are not started after a failover or switchover operation.

See Section A.3.2, "Failure to Bring Up Standby Instances After Failover or Switchover" in the *Oracle Application Server High Availability Guide* for information about the causes and solutions to this problem.

J.2.24 Diagnosing OracleAS Forms Services FRM-XXXXX Errors

For information about diagnosing FRM-xxxxx errors from OracleAS Forms Services, see Section A.2, "Diagnosing FRM-XXXXX Errors" in the*Oracle Application Server Forms Services Deployment Guide*.

J.2.25 Resolving OracleAS Forms Services Memory Problems

For information about resolving memory problems for OracleAS Forms Services, see Section A.6, "Resolving Memory Problems" in the*Oracle Application Server Forms Services Deployment Guide*.

J.2.26 Hanging Report Requests

When running report requests with the OracleAS Reports Server, the report request may "hang" for various reasons. This can lead to stability issues if not noticed in time.

See the section "Hanging Report Requests" in *Oracle Application Server Reports Services Publishing Reports to the Web* for information about the causes and solutions to this problem.

J.2.27 List of Values (LOV) Too Long for a Discoverer Portlet URL

A list of values (LOV) might be greater than the URL limit.

See Section D.1.14, "List of Values (LOV) Is Too Long for a Discoverer Portlet URL" in the *Oracle Business Intelligence Discoverer Configuration Guide* for information about the causes and solutions to this problem.

J.2.28 Out of Memory Problems for the OC4J_BI_forms JVM Process

Discoverer end users might encounter errors if the OC4J_BI_forms JVM process runs out of memory.

See Section D.1.18, "Out of memory problems for the OC4J_BI_forms JVM process" in the *Oracle Business Intelligence Discoverer Configuration Guide* for information about the causes and solutions to this problem.

J.2.29 Problems Editing or Creating Discoverer Portlets

If Discoverer Portlet Provider is not correctly registered in OracleAS Portal, you might

encounter errors when creating or editing Discoverer portlets.

See Section D.1.20, "Discoverer Portlet Provider issue" in the *Oracle Business Intelligence Discoverer Configuration Guide* for information about the causes and solutions to this problem.

J.2.30 Previously Working Application Using ADF Business Components Starts Throwing JDBC Errors

An application that previously successfully retrieved data suddenly starts throwing JDBC errors such as Connection Reset By Peer, Connection Closed, or Socket Reset By Peer.

See Section B.1.1, "Previously Working Application Using ADF Business Components Starts Throwing JDBC Errors" in the *Oracle Application Development Framework Development Guidelines Manual* for information about the causes and solutions to this problem.

J.3 Troubleshooting Application Server Control

This section describes problems that you might encounter when using Application Server Control and explains how to solve them. It contains the following topics:

- Application Server Control General Problems and Solutions
- OC4J Management Problems and Solutions

> **See Also:** Chapter 23, "Troubleshooting the Backup and Recovery Tool" for information about any troubleshooting backup and recovery operations performed within the Application Server Control Console

J.3.1 Application Server Control General Problems and Solutions

This section describes problems and solutions. It contains the following topics:

- Resetting the Administrator (ias_admin) Password
- Unavailable Metric and Chart Data in the Application Server Control Console
- Application Server Status Is Down When Server Components Are Up
- Errors When Starting Application Server Control
- Problems Connecting to an Application Server Instance from Farm or Cluster Page
- Application Server Home Page Indicates That the Farm Is Unavailable
- Error Connecting to the Directory Server
- Browser Displays "SMISession has been invalidated" Error
- Memory Errors Generated by the Oracle Management Agent
- Administration Tasks Performed Using the Command Line Are Not Reflected in Application Server Control Console
- SSL Timeout Issues with Microsoft Internet Explorer Browsers
- Session Has Expired Message When Using Multiple Browser Windows
- Topology Viewer Applet Not Loading
- No Propagation Between Grid Control and Application Server Control When Creating a New OC4J Instance
- Problems Viewing Metrics When Configured for Secure Sockets Layer (SSL)
- Problems Displaying the Date Selection Window When Searching the Log Repository

J.3.1.1 Resetting the Administrator (ias_admin) Password

To manage an instance of Oracle Application Server, you must log in to the Application Server Control Console using the current Administrator (`ias_admin`) password.

Problem

If you forget or do not know the `ias_admin` password, then you cannot monitor or administer the application server or its components with the Application Server Control Console.

Solution

Reset the `ias_admin` password using the following procedure while you are logged in as the user who installed the Oracle Application Server instance:

1. Stop the Application Server Control.

On UNIX systems, enter the following command in the Oracle home of the application server instance:

```
ORACLE_HOME/bin/emctl stop iasconsole
```

On Windows systems, use the Services control panel to stop the Application Server Control service.

2. Locate and open the following file in a text editor:

```
ORACLE_HOME/sysman/j2ee/config/jazn-data.xml
```

3. Locate the line that defines the credentials property for the `ias_admin` user.

   The following example shows the section of `jazn-data.xml` with the encrypted `credentials` entry in boldface type:

```
<realm>
  <name>enterprise-manager</name>
    <users>
      <user>
        <name>ias_admin</name>
          <credentials>{903}buG0lUsQqTq0nQjdaKQRECL1kbs192mP</credentials>
      </user>
.
.
```

4. Replace the existing encrypted password with the new password.

   Be sure to prefix the password with an exclamation point (!). For example:

```
<credentials>!mynewpassword123</credentials>
```

   The password for the `ias_admin` user should conform to following guidelines:

   - The minimum length is five alphanumeric characters.
   - At least one of the characters must be a number.
   - Passwords must be shorter than 30 characters.
   - Passwords can contain only alphanumeric characters from your database character set, the underscore (_), the dollar sign ($), and the number sign (#).
   - Passwords must begin with an alphabetic character. It cannot begin with a number, the underscore (_), the dollar sign ($), or the number sign (#).

     See Also: "The ias_admin User and Restrictions on its Password" in the *Oracle Application Server Installation Guide*

5. Start the Application Server Control.

   After the restart, the Application Server Control will use your new Administrator (`ias_admin`) password, which will be stored in encrypted format within the `jazn-data.xml` file.

   See Also: Section A.1.1, "Starting and Stopping the Application Server Control Console on UNIX"

J.3.1.2 Unavailable Metric and Chart Data in the Application Server Control Console

The performance metrics for a particular component show up as "Unavailable" in the Application Server Control Console.

Similarly, instead of a chart, one of the following messages (or a similar message) appears on the component Home page:

- `The underlying data for the CPU usage graph is not yet available`
- `The underlying data for the Memory usage graph is not yet available`

This problem often occurs immediately after the component is started.

Problem

Some metric data must be collected multiple times before the data can be displayed in the Application Server Control Console.

Solution

Verify that the component is up and running. If the component is down, restart the component.

If the component is up and running, wait at least five minutes to allow the necessary metrics to initialize, and then click the **Refresh Data** icon to refresh the data on the page.

J.3.1.3 Application Server Status Is Down When Server Components Are Up

From the Oracle Application Server Home page, you can quickly review the status of individual application server components, as well as the status of the overall application server instance itself.

Problem

Sometimes the Application Server Control Console indicates that the application server instance is down when components of the application server are up and running.

Solution

An Oracle Application Server instance is considered down when any one of its enabled components is down. For example, if one of your OC4J instances is down, the application server is considered down, even if the other components, such as Oracle HTTP Server, are up and running.

You can fix this problem by disabling components that are not in use. When a component is disabled, its status does not affect the status of the Application Server target. You can always enable the component at a later time.

To disable a component, click **Enable/Disable Components** on the Application Server Home page. Use the resulting page to determine which components you want to enable or disable for this application server instance.

> **See Also:** "Disabling and Enabling Components" in the Enterprise Manager online help

J.3.1.4 Errors When Starting Application Server Control

Before you can perform application server administrative tasks with Enterprise Manager, you must start Application Server Control.

Problem

When you attempt to start Application Server Control—or when the Enterprise Manager configuration assistant in the installation procedure attempts to start Application Server Control—an error occurs and the necessary processes are not started.

Solution

Problems starting Application Server Control are often caused by port conflicts on the host computer. In other words, a specific port number that the Application Server Control requires is in use by another application on the machine.

The first step in troubleshooting port conflicts is to identify which ports are in conflict. Some of the more common port conflicts that affect the Application Server Control include:

- The port used in the Application Server Control Console URL
- The port used by the Oracle Management Agent
- The port used by OC4J Remove Method Invocation (RMI)

You can often identify a port conflict (or other startup problem) by reviewing the log files associated with these technologies or components. Table J–1 describes some of these component log files, which are installed in the `sysman/log` directory of the Oracle Application Server home directory.

Table J–1 Log Files to Review When Troubleshooting Application Server Control Port Conflicts

Log File	Information To Look For in the Log File
`emiasconsole.nohup`	Information and errors generated during startup of the Application Server Control
`rmi.log`	Information and errors generated by OC4J RMI
`emagent.log`	Information and errors generated by the Oracle Management Agent
`em-application.log`	Additional information and errors generated by the OC4J instance used to deploy the Application Server Control

See Also: Chapter 5, "Managing Log Files"

After you identify a port conflict, you can modify the Application Server Control port number (if the port number can be reassigned).

See Also: Section 4.3.1, "Changing Oracle Enterprise Manager Ports"

J.3.1.5 Problems Connecting to an Application Server Instance from Farm or Cluster Page

From the Oracle Application Server Farm home page, you can view a list of the individual application server instances and OracleAS Cluster that are part of the Farm. To navigate to the Application Server Home home page for an instance, click the name of the application server target on the Farm home page. Similarly, the Cluster page provides a list of the application server instances that belong to the selected OracleAS Cluster.

Problem

When you click the name of the application server instance on the Farm home page, you receive one of the following errors in your Web browser:

- In your Netscape browser:

  ```
  There was no response. The server could be down or is not responding.
  ```

- In your Internet Explorer browser:

  ```
  The page cannot be displayed.
  ```

Solution

This error is most often displayed when the Application Server Control for the instance you selected is not running.

> **Note:** In previous versions of Oracle Application Server, one Application Server Control was used to manage all the application server instances installed on a host. Starting with 10*g* (9.0.4), each application server instance, by default, requires an Application Server Control and Management Agent to be running from its Oracle home.

To fix the problem, you must start the Application Server Control for the instance you want to manage:

1. Log on to the host where the application server instance resides.

   Be sure to log in as the user who installed the Oracle Application Server instance.

2. Start the Application Server Control.

   On UNIX systems, use the following command to start the Application Server Control:

   ```
   ORACLE_HOME/bin/emctl start iasconsole
   ```

   On Windows systems, use the Services control panel to start the Application Server Control service.

   > **See Also:** Section A.1, "Starting and Stopping the Application Server Control"

J.3.1.6 Application Server Home Page Indicates That the Farm Is Unavailable

OracleAS Farm is a set of Oracle Application Server instances that share a common repository. If the instance you are managing is part of an OracleAS Farm, the Application Server Control Console URL displays the Farm page. The Farm page lists the application servers and OracleAS Cluster that are part of the Farm.

Problem

Sometimes the Application Server Control Console URL does not display the Farm page. Instead, Enterprise Manager displays the Application Server Home page. In the General section of the page, the **Farm** field indicates that the Farm is "Unavailable." Alternatively, the **Farm** field indicates that the "Infrastructure database is unavailable."

Solution

In most cases you can solve this problem by making sure that the OracleAS Metadata Repository database or Farm repository database is up and running. You can verify the status of the database by using one of two methods:

- Display the Oracle Enterprise Manager Database Control Console for the Infrastructure Oracle home. The Database Control Console provides you with a Web-based user interface for monitoring and administering the Infrastructure database. From the Database Control Console, you can obtain the status of the database.

  See Also: Section 2.5, "Managing the OracleAS Metadata Repository Database with Database Control"

- Use SQL*Plus to connect to the database and verify that it is up and running.

If the database is down, start the database and then navigate to the Application Server Control Console URL.

If the OracleAS Metadata Repository database is up and running, make sure the Oracle Internet Directory component of your OracleAS Identity Management installation is also available. Display the Application Server Control Console for the OracleAS Identity Management installation and check to be sure the Oracle Internet Directory component is up and running.

If both the OracleAS Metadata Repository database and Oracle Internet Directory are up and running, the problem may be related to port conflicts, which can affect the Oracle Process Manager and Notification Server (OPMN). Check the OPMN log files to identify the potential port conflict.

See Also: Chapter 5, "Managing Log Files"

If you identify a port conflict, modify the port number (if the port number can be reassigned).

See Also: Chapter 4, "Managing Ports"

J.3.1.7 Error Connecting to the Directory Server

If your application server instance is part of an OracleAS Farm, some components of your application server instance may require access to the Identity Management components. Specifically, they may need access to Oracle Internet Directory.

Problem

When you log in to the Application Server Control Console, the following error message appears:

```
Unable to Connect to Directory Server:javax.naming.CommunicationException
```

Solution

This problem is caused when the Oracle Internet Directory component is down or unavailable. Verify that Oracle Internet Directory is up and running and start it if necessary.

For example, log in to the Identity Management host and enter the following command in the Infrastructure Oracle home to start the Oracle Internet Directory:

```
opmnctl startproc ias-component=OID
```

See Also: *Oracle Process Manager and Notification Server Administrator's Guide* for more information about starting and stopping OPMN components, such as Oracle Internet Directory

J.3.1.8 Browser Displays "SMISession has been invalidated" Error

Using Application Server Control, you can manage Oracle Application Server from a browser. As a result, you can manage your application server instances remotely as long as you have access to the network. In addition, multiple administrators can manage your application server instances.

Problem

In some cases, you may see the following error message displayed in your browser window:

```
The SMISession has been invalidated. Resolution: Please close the current
SMISession, start another one and reapply the actions
```

Solution

To resolve this issue, click the **Refresh Data** icon located to the right of the time stamp, or close and reopen the browser to start a new session. This error can be the result of multiple users performing conflicting configuration actions on a single Enterprise Manager Application Server Control at the same time.

J.3.1.9 Memory Errors Generated by the Oracle Management Agent

Oracle Application Server includes a version of the Oracle Management Agent that gathers monitoring data for the Application Server Control Console.

Problem

The Management Agent generates "out of memory" errors while collecting application server metrics.

Solution

Use the following procedure to increase the amount of memory available to the Management Agent Java Virtual Machine (JVM). The default value is 64 MB:

1. Use a text editor to open the following configuration file in the application server Oracle home:

   ```
   (UNIX) $ORACLE_HOME/sysman/config/emd.properties
   (Windows) %ORACLE_HOME%\sysman\config\emd.properties
   ```

2. Locate the following entry in the emd.properties file:

   ```
   agentJavaDefines=-Doracle.dms.refresh.wait.time=1000
   ```

3. Add the following qualifier to the `agentJavaDefines` property to increase the available memory to 128 MB:

   ```
   -Xmx128M
   ```

4. Restart the Application Server Control.

See Also: Section A.1.1, "Starting and Stopping the Application Server Control Console on UNIX"

J.3.1.10 Administration Tasks Performed Using the Command Line Are Not Reflected in Application Server Control Console

Application Server Control is the preferred management tool for most of your Oracle Application Server management tasks. However, you can still accomplish your management tasks using various command line tools.

Problem

If you use command-line tools to make administration or configuration changes to an Oracle Application Server instance (for example, if you use the `dmctl applyarchiveto` command), the changes are not reflected in the Application Server Control Console until after the Application Server Control cache is cleared.

Solution

To clear the cache, click the **Refresh Data** icon, which is located to the right of the time stamp, or close and reopen the browser to start a new session.

J.3.1.11 SSL Timeout Issues with Microsoft Internet Explorer Browsers

You can use the `emctl secure iasconsole` command to configure the Application Server Control so it uses HTTPS secure communications.

> **See Also:** Section A.4, "Configuring Security for Application Server Control Console"

However, after you configure security for the Application Server Control, you may get intermittent problems when using Microsoft Internet Explorer 6.0 or a later release.

Problem

Microsoft Internet Explorer has known issues with trying to reuse SSL connections after they have timed out. Due to this limitation, users connecting to Application Server Control using Internet Explorer, may see intermittent errors. Some examples of the errors include the following:

- `500 Internal Server Error` when deploying a J2EE application
- `Error: Processing already completed` after responding to a confirmation message
- The graphics in the HTML version of Topology Viewers do not appear

Solution

To work around these SSL timeout errors, you can upgrade all browsers to use the correct Microsoft patches. For information about the Internet Explorer problem, its workarounds, and links to updates to Internet Explorer 6.0 and later, see the following:

`http://support.microsoft.com/default.aspx?kbid=831167`

J.3.1.12 Session Has Expired Message When Using Multiple Browser Windows

Browser displays a message saying that the `Session has expired`.

Problem

Opening multiple browser windows to view different Application Server Control Consoles on the same host may cause the browser to post a `Session has expired` message if you switch between the browser windows. For example, you are viewing one Application Server Control Console located at:

```
http://mgmthost1.acme.com:1156/
```

You then open another browser to view an Application Server Control Console located at:

```
http://mgmthost1.acme.com:18100
```

As you switch between the two browser windows, you might receive a `Session expired message`. This condition can occur with either Netscape Navigator or Internet Explorer.

Solution

To avoid this problem, start a new browser instance from the desktop and close any new windows opened from the original browser session. If you are using Netscape 7, you will need to create a new Netscape Profile for any additional browser windows.

J.3.1.13 Topology Viewer Applet Not Loading

You can specify whether you want to use the HTML Only version of the Topology Viewer or the Java applet version. To use the Java applet version, you must have the correct Java Plug-in support and proxy settings.

> **See Also:** "Setting the Topology Viewer Preferences" in the Enterprise Manager online help

Problem 1

The Java applet version of the Topology version requires Java Plug-in release 1.4 or later. Typically, your browser will prompt you to download the required version of Java Plug-in. However, in some browsers, you may not be prompted to download the plug-in, or you may be directed to an invalid URL. Without the correct plug-in support, the Topology Viewer applet does not load.

Solution 1

You must manually download and install the plug-in from `http://java.sun.com/products/plugin/`.

Problem 2

If the Topology Viewer applet does not load, then look in the Java Plug-in Console for errors. Typically, the cause of these errors is the Java Plug-in could not access the proxy server.

You can start the Java Plug-in Console in one of three ways:

- Windows system tray

  Right-click the **Java Console** icon from the system tray, and select **Open Console**.

- Microsoft Internet Explorer

  From the **Tools** menu, select **Sun Java Console**.

- Netscape

  From the **Tools** menu, select **Web Development** > **Java Console**.

Solution 2

Configure the proxy settings to automatically detect settings and disable the use of the automatic configuration script. The details of this configuration varies from browser to browser.

See Also:

- http://java.com/en/download/help/redximage.jsp for further information about the applet error and configuring proxy settings
- http://java.sun.com/products/plugin/index.jsp for information about the Java Plug-in technology
- http://java.sun.com/j2se/1.4.2/docs/guide/plugin/developer_guide/faq/troubleshooting.html for further information about troubleshooting the Java Plug-in technology

J.3.1.14 No Propagation Between Grid Control and Application Server Control When Creating a New OC4J Instance

You can use Grid Control to discover Oracle Application Server instances.

Problem

If Grid Control release 10.1.0.2 or 10.1.0.3 is used to discover a release 9.0.4 or 10.1.2 Oracle Application Server instance, configuration changes, such as enabling or disabling components and creating or deleting OC4J instances in the Oracle Application Server instance will not be propagated to the central Management Agent and the Grid Control Console.

Solution

In order to propagate these changes to the Grid Control Console, remove the Oracle Application Server and its components from Grid Control with the following steps. When you remove the components from Grid Control, any collected data in the Management Repository will be deleted.

1. Click the **Targets** tab, and then **All Targets** subtab.
2. Select the Application Server instance, and click **Remove**.
3. Select the BC4J component for the Application Server instance, and click **Remove**.
4. Ensure that all targets have been deleted:
   a. Click the **Management System** tab.
   b. In the Management Services and Repository Overview page, in the **General** section, click the link **Deleted Targets**.

Once the targets are deleted, perform the following steps in the Application Server Control:

1. Navigate to the Application Server Home page for the Oracle Application Server instance.
2. From the Application Server Home page, click the **Infrastructure** tab.
3. In the Grid Control Management section, click **Configure**.
4. In the Configure Grid Control page, select the appropriate Management Service, and then click **OK**.

The Oracle Application Server and its components will now appear in the Grid Control Console. You can now enable or disable components and create or delete OC4J instances in the Oracle Application Server.

J.3.1.15 Problems Viewing Metrics When Configured for Secure Sockets Layer (SSL)

When you use Application Server Control Console to monitor targets, such as an instance of OracleAS Portal, running in an environment configured for SSL, some performance metrics may not display.

To correct this problem you must allow the Application Server Control to recognize the Certificate Authority that was used by the Web site you are monitoring to support HTTPS. You must add the Certificate of that Certificate Authority to the list of Certificate Authorities recognized by the Application Server Control.

> **See Also:** Section 16.3.7, "Configuring SSL for Oracle Enterprise Manager 10g"

J.3.1.16 Problems Displaying the Date Selection Window When Searching the Log Repository

Problem

If you are searching the Log Repository for log entries that occurred during a specific time frame, you might have problems displaying the pop-up date selection window.

Solution

This problem can be caused by customizations that you made to Microsoft Internet Explorer browser. For example, if you installed third-party pop-up blocking software, the browser may not be able to display the date selection window.

To work around this problem, deinstall or disable the browser customizations. Alternatively, enter the date directly into the date field, using the following date format: `MM/DD/YY`.

> **See Also:** "Searching the Log Repository" in the Application Server Control Console online help

J.3.2 OC4J Management Problems and Solutions

The following sections describe problems and issues when using Application Server Control to manage an OC4J instance and the J2EE applications you deploy:

- Problems Using the OC4J Security Page
- Lookup Error When Deploying an OC4J Application
- Redeploying WAR Applications with Application Server Control
- Deployment Performance in Internet Explorer and Netscape Navigator 7.0
- Problems Deploying Large OC4J Applications
- Troubleshooting OC4J Out of Memory Errors

J.3.2.1 Problems Using the OC4J Security Page

You use the OC4J Security page in the Application Server Control Console to configure various security settings for your deployed J2EE applications.

Problem 1

After making changes on the OC4J Security page, the changes do not seem to affect the deployed application.

Solution 1

After you make changes on the OC4J Security page, you must restart the OC4J instance in order for the changes to take effect. For example, if you add the user `admin` user and the `administrators` group as described in Section J.3.2.2, you must restart the OC4J instance to complete the procedure.

Problem 2

OC4J security employs a user manager to authenticate and authorize users and groups that attempt to access a J2EE application. One of the user managers that can be used to designate the users and groups for an application is the JAZN user manager.

With Application Server Control, you can specify that the JAZN user manager be associated with an application. Using the Deploy Application: User Manager page of the Application Server Control Console, you can specify that the application use either a JAZN XML configuration or a JAZN LDAP configuration.

When you use Application Server Control Console to specify an XML-based JAZN configuration, the following line is entered into the `orion-application.xml` file:

```
<jazn provider="XML" location="./jazn-data.xml" />
```

When you use Application Server Control Console to specify an LDAP-based JAZN configuration, the following line is entered in the orion-application.xml file:

```
<jazn provider="LDAP" default-realm="sample_subrealm" />
```

Some applications may prefer to specify a JAZN configuration by providing a path to a `jazn.xml` file, but Enterprise Manager does NOT support this type of JAZN configuration. This type of JAZN configuration would be specified as follows in the orion-application.xml file:

```
<jazn config="jazn.xml"/>
```

If you manually specify this type of JAZN configuration in the `orion-application.xml` file, you will either be unable to use the Application Server Control Console OC4J Security page or you will experience problems even after apparently using the page successfully.

Solution 2

Do not manually configure JAZN by providing a path to the `jazn.xml` file.

For more information about user managers specifying users and groups for a J2EE application, see *Oracle Application Server Containers for J2EE Security Guide*.

J.3.2.2 Lookup Error When Deploying an OC4J Application

From the Application Server Control Console, you can deploy J2EE applications to Oracle Application Server Containers for J2EE (OC4J).

Problem

When you are attempting to deploy an OC4J application using the Application Server Control Console, you may receive the following error:

```
Deployment failed: Nested exception
```

```
Root Cause: Lookup error: javax.naming.NoPermissionException: Not allowed to
look up java:comp/ServerAdministrator, check the namespace-access tag setting
in orion-application.xml for details;
```

Solution

This error may appear if the user manager for the OC4J default application does not include the user `admin` and the group `administrators`.

To view or define the users and groups for the default application user manager:

1. Navigate to the OC4J home page for the OC4J instance you used to deploy your application.
2. Click **Applications** to display the list of application deployed in the selected OC4J instance.
3. Click the **Default Application Name**, which appears at the top of the Applications page.

   Enterprise Manager displays the OC4J Application home page for the default application.
4. Scroll to the bottom of the page and click **Security**.

   Enterprise Manager displays the Security page, which lists the Groups and Users.

J.3.2.3 Redeploying WAR Applications with Application Server Control

On the OC4J Applications Page in the Application Server Control Console, you can deploy EAR files (applications with a file type of `.ear`) and deploy WAR files (web applications with a file type of `.war`).

To deploy a WAR file using the Application Server Control Console, click **Deploy War file** on the OC4J Applications Page.

The first time you deploy a WAR file, Enterprise Manager launches a deployment tool that automatically wraps the WAR application into a J2EE application (`.ear` file) before deploying it. The `.ear` file that Enterprise Manager creates to deploy your WAR file contains an `application.xml` file that describes the application modules. The `.ear` file is given an application name that you supply when you step through the deployment tool. After the WAR application is deployed, the name of the new application (`.ear` file) appears in the **Deployed Applications** table.

Problem

After you have deployed a WAR file using the Application Server Control Console, it cannot be redeployed by selecting the application (`.ear` file) on the OC4J Applications Page and clicking **Redeploy**.

Solution

To redeploy a WAR file using Application Server Control Console, you must undeploy the application first, then deploy it again by following these steps:

1. In the **Deployed Applications** table on the OC4J Applications Page, select the application (`.ear` file) in which the WAR file was wrapped and deployed.
2. Click **Undeploy**.
3. Click **Deploy War file**. In the deployment tool, specify the same application name as you specified the first time for the application (`.ear` file) in which the WAR file was wrapped and deployed.

After the WAR application is deployed, the name of the web application (`.ear` file) appears in the Deployed Applications table.

J.3.2.4 Deployment Performance in Internet Explorer and Netscape Navigator 7.0

Problem

If you attempt to deploy an OC4J application while using Microsoft Internet Explorer or Netscape 7.0, the file upload may take an extremely long time (for example, 10 minutes for a 45 MB `.ear` file as compared to 15 seconds with Netscape 7.1).

Solution

If you are using Netscape Navigator, upgrade to Netscape 7.1.

If you are using Internet Explorer, refer to the following Microsoft knowledge base article, which addresses this problem:

http://support.microsoft.com/default.aspx?scid=kb;en-us;329781

J.3.2.5 Problems Deploying Large OC4J Applications

Problem

When attempting to deploy a large application (greater than 50 MB EAR or WAR file), the default configuration of the Distributed Configuration Management (DCM) Daemon is insufficient. Attempting to deploy such a large application results in an "Out Of Memory" error.

Solution

Use the Application Server Control Console to increase the memory for the DCM Daemon component:

1. Navigate to the Application Server home page in the Application Server Control Console.
2. Click **Process Management** to edit the `opmn.xml` file.
3. Locate the `java-parameters` `<data>` tag in the DCM Daemon configuration section of the file:

```
<ias-component id="dcm-daemon" status="enabled" id-matching="true">
  <process-type id="dcm-daemon" module-id="DCMDaemon">
     <process-set id="dcm" numprocs="1">
        <module-data>
           <category id="start-parameters">
              <data id="java-parameters" value="-Xmx256m -Xrs
 -Doracle.ias.sysmgmt.logging.loglevel=ERROR
 -Djava.net.preferIPv4Stack=true
 -Djava.io.tmpdir=&amp;quot;$TMP&amp;quot;"/>
                 .
                 .
                 .
</ias-component>
```

4. Replace the string `-Xmx256m` with the string `-Xmx512m`.

   For example:

```
<data id="java-parameters" value="-Xmx512m
```

This new value increases the memory assigned to the DCM Daemon from 256 MB to 512 MB.

5. Click **Apply** to save your changes.
6. Open a terminal window (UNIX) or a DOS Command window (Windows) use the following commands to reload the OPMN configuration file, restart DCM, and restart the Application Server Control.

   On UNIX systems:

   ```
   ORACLE_HOME/opmn/bin/opmnctl reload
   ORACLE_HOME/opmn/bin/opmnctl restart ias-component="dcm-daemon"
   ORACLE_HOME/bin/emctl restart iasconsole
   ```

   On Windows systems:

   ```
   ORACLE_HOME\opmn\bin\opmnctl reload
   ORACLE_HOME\opmn\bin\opmnctl restart ias-component="dcm-daemon"
   ORACLE_HOME\bin\emctl restart iasconsole
   ```

7. Try deploying the application again.

J.3.2.6 Troubleshooting OC4J Out of Memory Errors

Problem

Depending upon the size and number of applications you deploy to your OC4J instance, you might experience "out of memory" errors.

Solution

Adjust the Java Virtual Machine (JVM) heap size for your OC4J processes.

See Also:

- "Setting the JVM Heap Size for OC4J Processes" in the chapter "Optimizing J2EE Applications In OC4J" in the *Oracle Application Server Performance Guide*
- "Administering OC4J Server Properties" in the Application Server Control Console online help

J.4 Need More Help?

You can find more solutions on Oracle*MetaLink*, http://metalink.oracle.com. If you do not find a solution for your problem, log a service request.

See Also: *Oracle Application Server Release Notes*, available on the Oracle Technology Network:

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

Glossary

access control

The ability of a system to grant or limit access to specific data for specific clients or groups of clients.

Access Control Lists (ACLs)

The group of access directives that you define. The directives grant levels of access to specific data for specific clients, or groups of clients, or both.

Advanced Encryption Standard

Advanced Encryption Standard (AES) is a cryptographic algorithm that has been approved by the National Institute of Standards and Technology as a replacement for DES. The AES standard is available in Federal Information Processing Standards Publication 197. The AES algorithm is a symmetric block cipher that can process data blocks of 128 bits, using cipher keys with lengths of 128, 192, and 256 bits.

AES

See **Advanced Encryption Standard**

attribute

An item of information that describes some aspect of an entry in an LDAP directory. An entry comprises a set of attributes, each of which belongs to an **object class**. Moreover, each attribute has both a *type*, which describes the kind of information in the attribute, and a *value,* which contains the actual data.

authentication

The process of verifying the identity of a user, device, or other entity in a computer system, often as a prerequisite to granting access to resources in a system. A recipient of an authenticated message can be certain of the message's origin (its sender). Authentication is presumed to preclude the possibility that another party has impersonated the sender.

authentication method

A security method that verifies the identity of a user, client, or server in distributed environments. Network authentication methods can also provide the benefit of **single sign-on (SSO)** for users. The following authentication methods are supported in Oracle Application Server:

- **Kerberos**
- **Secure Sockets Layer (SSL)**
- **Windows native authentication**

authorization

Authorization is the evaluation of security constraints to send a message or make a request. Authorization uses specific criteria to determine whether the request should be permitted. The criteria are **authentication** and restriction.

auto login wallet

An Oracle Wallet Manager feature that enables PKI- or password-based access to services without providing credentials at the time of access. This auto login access stays in effect until the auto login feature is disabled for that wallet. File system permissions provide the necessary security for auto login wallets. When auto login is enabled for a wallet, it is only available to the operating system user who created that wallet. Sometimes these are called "SSO wallets" because they provide single sign-on capability.

base

The root of a subtree search in an **LDAP**-compliant directory.

CA

See **certificate authority**

certificate

An ITU X.509 Version 3 standard data structure that securely binds an identity to a public key.

A certificate is created when an entity's public key is signed by a trusted identity, a certificate authority. The certificate ensures that the entity's information is correct and that the public key actually belongs to that entity.

A certificate contains the entity's name, identifying information, and public key. It is also likely to contain a serial number, expiration date, and information about the rights, uses, and privileges associated with the certificate. Finally, it contains information about the certificate authority that issued it.

certificate authority

A trusted third party that certifies that other entities—users, databases, administrators, clients, servers—are who they say they are. When it certifies a user, the certificate authority first seeks verification that the user is not on the certificate revocation list (CRL), then verifies the user's identity and grants a certificate, signing it with the certificate authority's private key. The certificate authority has its own certificate and public key which it publishes. Servers and clients use these to verify signatures the certificate authority has made. A certificate authority might be an external company that offers certificate services, or an internal organization such as a corporate MIS department.

certificate chain

An ordered list of certificates containing an end-user or subscriber certificate and its certificate authority certificates.

certificate request

A request, which consists of three parts: certification request information, a signature algorithm identifier, and a digital signature on the certification request information. The certification request information consists of the subject's distinguished name, public key, and an optional set of attributes. The attributes may provide additional information about the subject identity, such as postal address, or a challenge password by which the subject entity may later request certificate revocation. See **PKCS #10**

certificate revocation lists

(CRLs) Signed data structures that contain a list of revoked **certificate**s. The authenticity and integrity of the CRL is provided by a digital signature appended to it. Usually, the CRL signer is the same entity that signed the issued certificate.

Cipher Block Chaining (CBC)

An encryption method that protects against block replay attacks by making the encryption of a cipher block dependent on all blocks that precede it; it is designed to make unauthorized decryption incrementally more difficult. Oracle Advanced Security employs *outer* cipher block chaining because it is more secure than *inner* cipher block chaining, with no material performance penalty.

cipher suite

A set of authentication, encryption, and data integrity algorithms used for exchanging messages between network nodes. During an SSL handshake, for example, the two nodes negotiate to see which cipher suite they will use when transmitting messages back and forth.

cipher suite name

Cipher suites describe the kind of cryptographics protection that is used by connections in a particular session.

ciphertext

Message text that has been encrypted.

cleartext

Unencrypted plain text.

client

A user, software application (such as a browser), or computer that requests the services, data, or processing of another application or computer (the **server**). A client relies on a service.

cluster

A collection of application server **instance**s with identical configuration and application deployment. Clusters enforce homogeneity between member instances so that a cluster of application server instances can appear and function as a single instance. With appropriate front-end load balancing, any instance in an application server cluster can serve **client** requests. This simplifies configuration and deployment across multiple instances and enables fault tolerance among clustered instances.

confidentiality

A function of cryptography. Confidentiality guarantees that only the intended recipient of a message can view the message (decrypt the ciphertext).

connect descriptor

A specially formatted description of the destination for a network connection. A connect descriptor contains destination **service** and network route information. The destination service is indicated by using its service name for Oracle databases. The network route provides, at a minimum, the location of the **listener** through use of a network address. See **connect identifier**.

connect identifier

A **connect descriptor** or a name that maps to a connect descriptor. A connect identifier can be a **net service name**, database **service name**, or **net service alias**. Users initiate a connect request by passing a username and password along with a connect identifier in a connect string for the service to which they wish to connect:

```
CONNECT username/password@connect_identifier
```

connect string

Information the user passes to a **service** to connect, such as username, password and **net service name**. For example:

```
CONNECT username/password@net_service_name
```

credentials

A username, password, or certificate used to gain access to Oracle Database, Oracle Application Server 10g, or the Oracle Identity Management infrastructure.

CRL

See **certificate revocation lists**

CRL Distribution Point

(CRL DP) An optional extension specified by the X.509 version 3 certificate standard, which indicates the location of the Partitioned CRL where revocation information for a certificate is stored. Typically, the value in this extension is in the form of a URL. CRL DPs allow revocation information within a single **certificate authority** domain to be posted in multiple CRLs. CRL DPs subdivide revocation information into more manageable pieces to avoid proliferating voluminous CRLs, thereby providing performance benefits. For example, a CRL DP is specified in the certificate and can point to a file on a Web server from which that certificate's revocation information can be downloaded.

CRL DP

See **CRL Distribution Point**.

cryptography

The practice of encoding and decoding data, resulting in secure messages.

data dictionary

A set of read-only tables that provide information about a database.

Data Encryption Standard (DES)

The U.S. data encryption standard.

database alias

See **net service name**.

decryption

The process of converting the contents of an encrypted message (ciphertext) back into its original readable format (plaintext).

DES

See **Data Encryption Standard (DES)**

Diffie-Hellman key negotiation algorithm

A method that lets two parties communicating over an insecure channel to agree upon a random number known only to them. Though the parties exchange information over the insecure channel during execution of the Diffie-Hellman key negotiation algorithm, it is computationally infeasible for an attacker to deduce the random number they agree upon by analyzing their network communications. Oracle Advanced Security uses the Diffie-Hellman key negotiation algorithm to generate session keys.

digital signature

A digital signature is created when a public key algorithm is used to sign the sender's message with the sender's private key. The digital signature assures that the document is authentic, has not been forged by another entity, has not been altered, and cannot be repudiated by the sender.

directory naming

A **naming method** that resolves a database service, **net service name**, or **net service alias** to a **connect descriptor** stored in a central directory server.

directory naming context

A subtree which is of significance within a directory server. It is usually the top of some organizational subtree. Some directories only permit one such context which is fixed; others permit none to many to be configured by the directory administrator.

distinguished name (DN)

The unique name of an **LDAP**-based directory entry. A distinguished name comprises all of the individual names of the parent entries back to the root.

domain

Any tree or subtree within the **Domain Name System (DNS)** namespace. Domain most commonly refers to a group of computers whose host names share a common suffix, the domain name.

Domain Name System (DNS)

A system for naming computers and network services that is organized into a hierarchy of **domain**s. DNS is used in TCP/IP networks to locate computers through user-friendly names. DNS resolves a friendly name into an IP address, which is understood by computers.

encrypted text

Text that has been encrypted, using an encryption algorithm; the output stream of an encryption process. On its face, it is not readable or decipherable, without first being subject to **decryption**. Also called **ciphertext**. Encrypted text ultimately originates as **plaintext**.

encryption

The process of disguising a message rendering it unreadable to any but the intended recipient.

entry

In the context of a directory service, an entry is the building block of a directory. An entry is a collection of information about an object in the directory. Each entry is composed of a set of attributes that describe one particular trait of the object. For

example, if a directory entry describes a person, that entry can have attributes such as first name, last name, telephone number, or e-mail address.

external authentication

Verification of a user identity by a third party authentication service, such as **Kerberos**.

farm

A collection of **cluster**s and **instance**s that share the same Oracle Application Server Infrastructure. A farm can be file-based or database based. The repository for a file-based farm exists within the middle-tier instance Oracle home. The repository for a database-based farm exists within OracleAS Metadata Repository.

Federal Information Processing Standard (FIPS)

A U.S. government standard that defines security requirements for cryptographic modules—employed within a security system protecting unclassified information within computer and telecommunication systems. Published by the **National Institute of Standards and Technology (NIST)**.

FIPS

See **Federal Information Processing Standard (FIPS)**.

grid computing

A computing architecture that coordinates large numbers of servers and storage to act as a single large computer. Oracle Grid Computing creates a flexible, on-demand computing resource for all enterprise computing needs. Applications running on the Oracle 10*g* grid computing infrastructure can take advantage of common infrastructure services for failover, software provisioning, and management. Oracle Grid Computing analyzes demand for resources and adjusts supply accordingly.

HTTP

Hypertext Transfer Protocol. The underlying format used by the Web to format and transmit messages and determine what actions Web servers and browsers should take in response to various commands. HTTP is the protocol used between Oracle Application Server and clients.

HTTP server

A **server** that receives HTTP requests from remote browsers, converts the requested URL to a filename, and returns the file to the requester.

HTTPS

Secure Hypertext Transfer Protocol. A protocol that uses the **Secure Sockets Layer (SSL)** as a sublayer under the regular**HTTP** application layer to encrypt and decrypt user page requests as well as the pages that are returned by the origin server.

identity

The combination of the public key and any other public information for an entity. The public information may include user identification data such as an e-mail address. A user certified as being the entity it claims to be.

identity management

The creation, management, and use of online, or digital, entities. Identity management involves securely managing the full life cycle of a digital identity from creation (provisioning of digital identities) to maintenance (enforcing organizational policies regarding access to electronic resources), and, finally, to termination.

identity management realm

A subtree in Oracle Internet Directory, including not only an **Oracle Context**, but also additional subtrees for users and groups, each of which are protected with access control lists.

IIOP

Internet inter-ORB protocol. An Internet transport protocol used by CORBA objects to communicate with each other. In the context of Oracle Application Server, IIOP is used by ECO/Java and EJB objects. IIOP is also used between Oracle Application Server components.

instance

The set of processes required to run the configured components within an application server installation. There can be only one application server instance for each application server installation. The terms installation and instance are sometimes used interchangeably; however, it is important to remember that an installation is the set of files installed into an Oracle home and an instance is a set of processes associated with those files.

integrity

The guarantee that the contents of the message received were not altered from the contents of the original message sent.

Java Database Connectivity (JDBC)

An industry-standard Java interface for connecting to a relational database from a Java program, defined by Sun Microsystems.

JDBC

See **Java Database Connectivity (JDBC)**.

Kerberos

A network authentication service developed under Massachusetts Institute of Technology's Project Athena that strengthens security in distributed environments. Kerberos is a trusted third-party authentication system that relies on shared secrets and assumes that the third party is secure. It provides **single sign-on (SSO)** capabilities and database link authentication (MIT Kerberos only) for users, provides centralized password storage, and enhances PC security.

key

When encrypting data, a key is a value which determines the **ciphertext** that a given algorithm will produce from given plaintext. When decrypting data, a key is a value required to correctly decrypt a ciphertext. A ciphertext is decrypted correctly only if the correct key is supplied.

With a symmetric encryption algorithm, the same key is used for both encryption and decryption of the same data. With an asymmetric encryption algorithm (also called a public-key encryption algorithm or public-key cryptosystem), different keys are used for encryption and decryption of the same data.

key pair

A **public key** and its associated **private key**. See **public and private key pair**.

LDAP

See **Lightweight Directory Access Protocol (LDAP)**.

ldap.ora file

A file created by Oracle Net Configuration Assistant that contains the following directory server access information:

- Type of directory server
- Location of the directory server
- Default identity management realm or Oracle Context (including ports) that the client or server will use

Lightweight Directory Access Protocol (LDAP)

A standard, extensible directory access protocol. It is a common language that LDAP clients and servers use to communicate. The framework of design conventions supporting industry-standard directory products, such as the Oracle Internet Directory.

listener

A process that resides on the server whose responsibility is to listen for incoming client connection requests and manage the traffic to the server. A listener can be an HTTP server that handles incoming requests and routes them to the dispatcher.

Every time a client requests a network session with a server, a listener receives the actual request. If the client information matches the listener information, then the listener grants a connection to the server.

listener.ora file

A configuration file for the Oracle Database listener that identifies the listener name, protocol addresses on which it is accepting connection requests, and services for which the listener is listening.

The `listener.ora` file typically resides in *`ORACLE_HOME`*`/network/admin` on UNIX platforms and *`ORACLE_HOME`*`\network\admin` on Windows.

man-in-the-middle

A security attack characterized by the third-party, surreptitious interception of a message, wherein the third party, the *man-in-the-middle*, decrypts the message, re-encrypts it (with or without alteration of the original message), and re-transmits it to the originally-intended recipient—all without the knowledge of the legitimate sender and receiver. This type of security attack works only in the absence of **authentication**.

message digest

Representation of text as a string of single digits. It is created using a formula called a one-way hash function, which is an algorithm that turns a message into a single string of digits. One-way means that it is almost impossible to derive the original message from the string of digits. The calculated message digest can be compared with the message digest that is decrypted with a **public key** to verify that the message has not been tampered with.

naming method

The resolution method used by a client application to resolve a **connect identifier** to a **connect descriptor** when attempting to connect to a service.

National Institute of Standards and Technology (NIST)

An agency within the U.S. Department of Commerce responsible for the development of security standards related to the design, acquisition, and implementation of cryptographic-based security systems within computer and telecommunication systems, operated by a Federal agency or by a contractor of a Federal agency or other organization that processes information on behalf of the Federal Government to accomplish a Federal function.

net service alias

An alternative name for a **directory naming** object in a directory server. A directory server stores net service aliases for any defined **net service name** or database service. A net service alias entry does not have connect descriptor information. Instead, it only references the location of the object for which it is an alias. When a client requests a directory lookup of a net service alias, the directory determines that the entry is a net service alias and completes the lookup as if it was actually the entry it is referencing.

net service name

The name used by clients to identify a database server. A net service name is mapped to a port number and protocol. Also known as a **connect string**, or **database alias**.

network authentication service

A means for authenticating clients to servers, servers to servers, and users to both clients and servers in distributed environments. A network authentication service is a repository for storing information about users and the services on different servers to which they have access, as well as information about clients and servers on the network. An authentication server can be a physically separate machine, or it can be a facility co-located on another server within the system. To ensure availability, some authentication services may be replicated to avoid a single point of failure.

network listener

A listener on a server that listens for connection requests for one or more databases on one or more protocols. See **listener**.

NIST

See **National Institute of Standards and Technology (NIST)**.

non-repudiation

Incontestable proof of the origin, delivery, submission, or transmission of a message.

obfuscation

A process by which information is scrambled into a non-readable form, such that it is extremely difficult to de-scramble if the algorithm used for scrambling is not known.

object class

A named group of **attribute**s. When you want to assign attributes to an entry, you do so by assigning to that entry the object classes that hold those attributes. All objects associated with the same object class share the same attributes.

Oracle Context

An entry in an LDAP-compliant internet directory called `cn=OracleContext`, under which all Oracle software relevant information is kept, including entries for checksumming security.

There can be one or more Oracle Contexts in a directory. An Oracle Context is usually located in an **identity management realm**.

Oracle Net Services

An Oracle product that enables two or more computers that run the Oracle server or Oracle tools to exchange data through a third-party network. Oracle Net Services support distributed processing and distributed database capability. Oracle Net Services is an open system because it is independent of the communication protocol, and users can interface Oracle Net to many network environments.

Oracle PKI certificate usages

Defines the purpose of the key contained in an **certificate**. Oracle PKI certificate usages are based on the key usages defined in the X.509 Version 3 standard.

PCMCIA cards

Small credit card-sized computing devices that comply with the Personal Computer Memory Card International Association (PCMCIA) standard. These devices, also called PC cards, are used for adding memory, modems, or as hardware security modules. PCMCIA cards that are used as hardware security modules securely store the private key component of a **public and private key pair** and some also perform the cryptographic operations as well.

peer identity

SSL connect sessions are between a particular client and a particular server. The identity of the peer may have been established as part of session setup. Peers are identified by **X.509 certificate chains**.

PEM

The Internet Privacy-Enhanced Mail protocols standard, adopted by the Internet Architecture Board to provide secure electronic mail over the Internet. The PEM protocols provide for encryption, authentication, message integrity, and key management. PEM is an inclusive standard, intended to be compatible with a wide range of key-management approaches, including both symmetric and public-key methods to encrypt data-encrypting keys. The specifications for PEM come from four Internet Engineering Task Force (IETF) documents: RFCs 1421, 1422, 1423, and 1424.

PKCS #10

An RSA Security, Inc., Public-Key Cryptography Standards (PKCS) specification that describes a syntax for certification requests. A certification request, also referred to as a **certificate request**, consists of a distinguished name, a public key, and optionally a set of attributes, collectively signed by the entity requesting certification.

PKCS #11

An RSA Security, Inc., Public-Key Cryptography Standards (PKCS) specification that defines an application programming interface (API), called Cryptoki, to hardware devices which hold cryptographic information and perform cryptographic operations. See also **PCMCIA cards**.

PKCS #12

An RSA Security, Inc., Public-Key Cryptography Standards (PKCS) specification that describes a transfer syntax for storing and transferring personal authentication credentials—typically in a format called a **wallet**.

PKI

See **public key infrastructure (PKI)**.

plaintext

Message text that has not been encrypted.

private key

In public-key cryptography, this key is the secret key. It is primarily used for decryption but is also used for encryption with digital signatures. See **public and private key pair**.

proxy authentication

A process typically employed in an environment with a middle tier such as a firewall, wherein the end user authenticates to the middle tier, which then authenticates to the directory on the user's behalf—as its *proxy*. The middle tier logs into the directory as a *proxy user*. A proxy user can switch identities and, once logged into the directory, switch to the end user's identity. It can perform operations on the end user's behalf, using the authorization appropriate to that particular end user.

public and private key pair

A set of two numbers used for **encryption** and **decryption**, where one is called the **private key** and the other is called the **public key**. Public keys are typically made widely available, while private keys are held by their respective owners. Though mathematically related, it is generally viewed as computationally infeasible to derive the private key from the public key. Public and private keys are used only with asymmetric encryption algorithms, also called public-key encryption algorithms, or public-key cryptosystems. Data encrypted with either a public key or a private key from a **key pair** can be decrypted with its associated key from the key pair. However, data encrypted with a public key cannot be decrypted with the same public key, and data enwrapped with a private key cannot be decrypted with the same private key.

public key

In public-key cryptography, this key is made public to all. It is primarily used for encryption but can be used for verifying signatures. See **public and private key pair**.

public key encryption

The process where the sender of a message encrypts the message with the public key of the recipient. Upon delivery, the message is decrypted by the recipient using its private key.

public key infrastructure (PKI)

Information security technology utilizing the principles of public key cryptography. Public key cryptography involves encrypting and decrypting information using a shared public and private key pair. It provides for secure, private communications within a public network.

realm

1. Short for **identity management realm**. 2. A **Kerberos** object. A set of clients and servers operating under a single key distribution center/ticket-granting service (KDC/TGS). Services in different realms that share the same name are unique.

realm Oracle Context

An **Oracle Context** that is part of an **identity management realm** in Oracle Internet Directory.

registry

A Windows repository that stores configuration information for a computer.

remote computer

A computer on a network other than the local computer.

restriction

A security scheme that restricts access to files provided by the server to client machines within certain groups of IP addresses or DNS domains.

root key certificate

See **trusted certificate**.

schema

1. Database schema: A named collection of objects, such as tables, views, clusters, procedures, packages, attributes, object classes, and their corresponding matching rules, which are associated with a particular user. 2. **LDAP** directory schema: The collection of attributes, object classes, and their corresponding matching rules.

Secure Sockets Layer (SSL)

An industry standard protocol designed by Netscape Communications Corporation for securing network connections. SSL provides authentication, encryption, and data integrity using public key infrastructure (PKI).

server

There are two types of servers relevant to this product. One is Oracle Application Server, which is a collection of middleware services and tools that provide a scalable, robust, secure, and extensible platform for distributed, object-oriented applications. Oracle Application Server supports access to applications from both Web clients (browsers) using HTTP and Common Object Request Broker Architecture (CORBA) clients, which use the CORBA and the Internet Inter-ORB (IIOP) protocols. The other is Oracle Database Server, which is a relational database server dedicated to performing data management duties on behalf of clients using any number of possible interfaces.

service

1. A network resource used by clients; for example, Oracle Application Server or Oracle database server.

2. An executable process installed in the Windows **registry** and administered by Windows. Once a service is created and started, it can run even when no user is logged on to the computer.

service name

A logical representation of a database, which is the way a database is presented to clients. A database can be presented as multiple services and a service can be implemented as multiple database instances. The service name is a string that is the global database name, that is, a name comprising the database name and domain name, entered during installation or database creation.

session key

A key shared by at least two parties (usually a client and a server) that is used for data encryption for the duration of a single communication session. Session keys are typically used to encrypt network traffic; a client and a server can negotiate a session key at the beginning of a session, and that key is used to encrypt all network traffic between the parties for that session. If the client and server communicate again in a new session, they negotiate a new session key.

single key-pair wallet

A **PKCS #12**-format **wallet** that contains a single user **certificate** and its associated **private key**. The **public key** is imbedded in the certificate.

single sign-on (SSO)

The ability of a user to *authenticate once*, combined with strong authentication occurring transparently in subsequent connections to other databases or applications. Single sign-on lets a user access multiple accounts and applications with a single password, entered during a single connection. *Single password, single authentication.*

smart card

A plastic card (like a credit card) with an embedded integrated circuit for storing information, including such information as user names and passwords, and also for performing computations associated with authentication exchanges. A smart card is read by a hardware device at any client or server.

A smartcard can generate random numbers which can be used as one-time use passwords. In this case, smartcards are synchronized with a service on the server so that the server expects the same password generated by the smart card.

sniffer

Device used to surreptitiously listen to or capture private data traffic from a network.

SSL

See **Secure Sockets Layer (SSL)**.

SSO

See **single sign-on (SSO)**.

system identifier (SID)

A unique name for an Oracle instance. To switch between Oracle databases, users must specify the desired SID. The SID is included in the `CONNECT DATA` part of the **connect descriptor** in a **tnsnames.ora** file, and in the definition of the **network listener** in a **listener.ora file**.

tnsnames.ora

A file that contains connect descriptors; each **connect descriptor** is mapped to a **net service name**. The file may be maintained centrally or locally, for use by all or individual clients. This file typically resides in the following locations depending on your platform:

- (UNIX) *ORACLE_HOME*`/network/admin`
- (Windows) *ORACLE_HOME*`\network\admin`

token card

A device for providing improved ease-of-use for users through several different mechanisms. Some token cards offer one-time passwords that are synchronized with an authentication service. The server can verify the password provided by the token card at any given time by contacting the authentication service. Other token cards operate on a challenge-response basis. In this case, the server offers a challenge (a number) which the user types into the token card. The token card then provides another number (cryptographically-derived from the challenge), which the user then offers to the server.

trusted certificate

A trusted certificate, sometimes called a root key certificate, is a third party identity that is qualified with a level of trust. The trusted certificate is used when an identity is being validated as the entity it claims to be. Typically, the certificate authorities you trust are called trusted certificates. If there are several levels of trusted certificates, a trusted certificate at a lower level in the certificate chain does not need to have all its higher level certificates reverified.

trusted certificate authority

See **certificate authority**.

trust point

See **trusted certificate**.

user search base

The node in the LDAP directory under which the user resides.

wallet

A wallet is a data structure used to store and manage security credentials for an individual entity. A **Wallet Resource Locator** (WRL) provides all the necessary information to locate the wallet.

wallet obfuscation

An **obfuscation** used to store and access an Oracle **wallet** without querying the user for a password prior to access (supports **single sign-on (SSO)**).

Wallet Resource Locator

(WRL) A locator that provides all necessary information to locate a **wallet**. It is a path to an operating system directory that contains a wallet.

Windows native authentication

An **authentication method** that enables a client single login access to a Windows server and a database running on that server.

WRL

See **Wallet Resource Locator**.

X.509

An industry-standard specification for digital **certificate**s.

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X