# Oracle® Application Server 10*g*

Upgrading to 10*g* (9.0.4)

10*g* (9.0.4) for UNIX

**Part No. B10429-01**

November 2003

ORACLE®

Oracle Application Server 10*g* Upgrading to 10*g* (9.0.4), 10*g* (9.0.4) for UNIX

Part No. B10429-01



Primary Author: Julia Pond

Contributing Authors: John Bellemore, Jane Bielawa, Kamalendu Biswas, Chung Cheng, Will Chin, Saheli Dey, Bob Donat, Joe Garcia, Mark Gizejewski, Nick Greenhalgh, Elizabeth Hanks, Susan Highmoor, Clara Jaeckel, Maneesh Joshi, Mark Kennedy, Sam Lee, Shaun Lin, Jeremy Lizt, Peter Lubbers, Sunil Marya, Valarie Moore, Oscar Naim, Bill Norcott, Lei Oh, Shane Potter, Tony Quan, Pavi Sandhu, Charlie Shapiro, Jianping Shi, Cheryl Smith, Debbie Steiner, Jeff Tang, Parthiban Thilagar, Deepak Thomas, Olaf van der Geest, Satishkumar Venkatasamy, Manoj Verma, Richard Wang, David Ward, Liujin Yu

# Contents

## 3 Upgrading the Middle Tier

## 4 Upgrading the Metadata Repository

## 5 Upgrading the Identity Management Services

## A Component Upgrade Process Reference

# Send Us Your Comments

**Oracle Application Server 10*g* Upgrading to 10*g* (9.0.4), 10*g* (9.0.4) for UNIX**

**Part No. B10429-01**

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appserverdocs_us@oracle.com
- FAX: 650.506.7365 Attn: Oracle Application Server
- Postal service:
  Oracle Corporation
  Oracle Application Server Documentation
  500 Oracle Parkway, M/S 1op6
  Redwood Shores, California 94065
  USA

If you would like a reply, please include your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

# Preface

This preface describes *Upgrading to 10g (9.0.4)*, including the intended audience, structure, and conventions for the document. It also identifies related Oracle documents.

## Intended Audience

This manual is intended for Oracle Application Server system administrators who are responsible for installing, maintaining, and upgrading Oracle Application Server instances. It is assumed that the readers of this manual have knowledge of the following:

- Oracle Application Server system administration and configuration
- The configuration and expected behavior of the system being upgraded

## Structure

*Upgrading to 10g (9.0.4)* contains the following chapters and appendixes:

### Chapter 1, "Overview of the Oracle Application Server Upgrade"

This chapter identifies terminology used in the guide and describes the upgrade processes for middle tier and Infrastructure installation types, including valid upgrade paths.

### Chapter 2, "Planning an Upgrade"

This chapter provides guidelines for planning an upgrade. It discusses compatibility between Oracle Application Server releases, devising an upgrade strategy, upgrade tasks, system downtime, and system availability during upgrade.

### Chapter 3, "Upgrading the Middle Tier"

This chapter contains instructions for starting and using the Oracle Application Server Upgrade Assistant, and instructions on completing the upgrade.

### Chapter 4, "Upgrading the Metadata Repository"

This chapter contains instructions for upgrading the Metadata Repository. It includes instructions on preparing to upgrade the Metadata Repository, and executing each component schema upgrade script.

### Chapter 5, "Upgrading the Identity Management Services"

This chapter contains instructions on upgrading the Identity Management Services (Oracle Application Server Single Sign-On and Oracle Internet Directory).

### Appendix A, "Component Upgrade Process Reference"

This appendix describes the processing performed by the OracleAS Upgrade Assistant and, if applicable, component schemas. It lists the upgrade items (files or schemas) involved in each process.

### Appendix B, "Files Reference"

This appendix lists all of the files affected by the Oracle Application Server upgrade. It provides the path to the file from the Oracle home and indicates whether the file is processed manually or by the OracleAS Upgrade Assistant. It also provides a brief description of each file.

## Related Documents

The following Oracle publications are referenced in this guide. They contain information that may be useful when performing an upgrade:

- *Oracle Application Server 10g Administrator's Guide*
- *Oracle Application Server 10g Installation Guide*
- *Oracle Application Server Discoverer Configuration Guide*
- *Oracle Application Server Wireless Developer's Guide*
- *Oracle Application Server Single Sign-On Administrator's Guide*
- *Oracle Internet Directory Administrator's Guide*
- *Distributed Configuration Management Reference Guide*

- *Oracle Application Server 10g Release Notes*
- *Oracle9i Database Administrator's Guide*
- *Step-By-Step Guide to Install Portal 9.0.2.3 in a Custom Database (document ID 238516.1)* This document is located at http://metalink.oracle.com.
- *Oracle Application Server 10g (9.0.4) Installation Guide*
- *Oracle Application Server Portal Error Messages Guide*
- *Oracle Application Server Portal Configuration Guide*
- *Oracle Application Server Portal User's Guide*
- *Oracle Application Server Web Cache Administrator's Guide*
- *Boosting Performance in Web Cache white paper*
- *How do I add additional Portal and Single Sign-On Targets to the Enterprise Manager Web Site technical note*

# Conventions

This guide uses the following conventions:

| Convention | Meaning |
|---|---|
| .<br>.<br>. | Vertical ellipsis points in an example mean that information not directly related to the example has been omitted. |
| . . . | Horizontal ellipsis points in statements or commands mean that parts of the statement or command not directly related to the example have been omitted |
| `monospace text` | File names, path names, command names, code, URLs |
| **`monospace bold text`** | Typed user input. |
| *`monospace italic text`* | Variables in text or code. |
| *<source_MT_OH>* | The full path to the Release 2 (9.0.2) or Release 2 (9.0.3) Middle Tier Oracle home. |
| *<destination_MT_OH>* | The full path to the 10*g* (9.0.4) Middle Tier Oracle home. |
| *<source_Infra_OH>* | The full path to the Release 2 (9.0.2) Infrastructure Oracle home. |
| *<destination_Infra_OH>* | The full path to the 10*g* (9.0.4) Infrastructure (Identity Management or Metadata Repository) Oracle home. |
| *<repCA_CD>* | The full path to the root directory of the Repository Configuration Assistant CD. |

# 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. Standards will continue to evolve over time, and Oracle Corporation 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 additional information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

**Accessibility of Code Examples in Documentation** JAWS, a Windows screen reader, 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, JAWS 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 Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

# 1

# Overview of the Oracle Application Server Upgrade

This chapter describes the process of upgrading the middle tier and Infrastructure installation types from Oracle9*i* Application Server Release 2 (9.0.2) or Release 2 (9.0.3), also referred to as the source Oracle home or source instance, to Oracle Application Server 10*g* (9.0.4), also referred to as the destination Oracle home or destination instance. The middle tier comprises application server instances that process requests from the client tier, and may access or change data in the database to complete a request. The Infrastructure, usually installed on a different computer, provides centralized security and management services and a metadata repository to one or more application server instances on the middle tier.

This chapter contains these major sections:



## 1.1 The Oracle Application Server Upgrade Process

This section describes the process of upgrading a distributed Oracle Application Server environment. A distributed environment is one in which multiple middle tiers access an Infrastructure, across multiple Oracle homes. A detailed discussion of the middle tier and Infrastructure portions of the upgrade process is provided in this chapter.

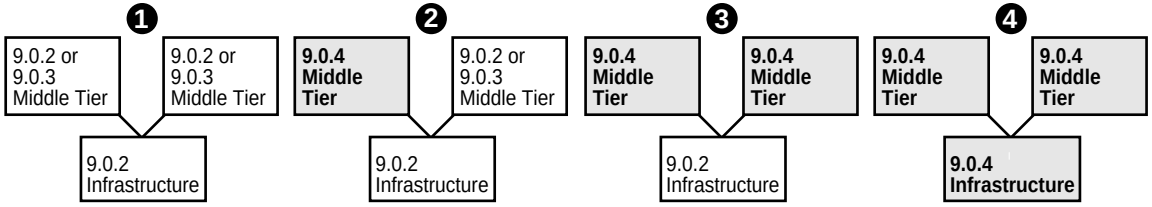
## 1.1.1 Upgrading a Distributed Environment

A distributed Oracle Application Server environment comprises one or more middle tiers and an Infrastructure, and is distributed across multiple Oracle homes. Upgrading such an environment involves upgrading each middle tier, and then upgrading the Infrastructure.

> **Note:** The Infrastructure itself comprises two units: the Metadata Repository and Identity Management (See Section 2.1, "Oracle Application Server Compatibility" on page 2-1 for valid upgrade sequences). Figure 1–1 illustrates the process of upgrading a distributed environment, depicting the Infrastructure as a single unit for simplicity's sake. Although you can upgrade Identity Management and the Metadata Repository in any order, the middle tier upgrade must precede the Metadata Repository upgrade.

1. Before the upgrade, middle tiers and Infrastructure are Release 2 (9.0.2) or Release 2 (9.0.3).
2. One middle tier is upgraded to 10*g* (9.0.4), and continues to use the Release 2 (9.0.2) infrastructure.
3. The remaining middle tiers are upgraded to 10*g* (9.0.4), and continue to use the Release 2 (9.0.2) Infrastructure.
4. The Release 2 (9.0.2) Infrastructure is upgraded to 10*g* (9.0.4).

***Figure 1–1 Upgrading a Distributed Oracle Application Server Environment***




## 1.1.2 Upgrading a Non-Distributed Environment

Upgrading a non-distributed Oracle Application Server environment (one that resides in a single Oracle home) involves upgrading the middle tier.

## 1.2 The Middle Tier Upgrade Process

The middle tier upgrade process is illustrated in Figure 1–2. The middle tier upgrade consists of the following steps:

1. The Release 2 (9.0.2) or Release 2 (9.0.3) middle tier installation resides on its computer, containing applications and configuration data.
2. A 10*g* (9.0.4) middle tier installation of the same installation type as the source installation is installed into a new Oracle home on the same computer.
3. The OracleAS Upgrade Assistant, a tool installed with Oracle Application Server that automates most middle tier component upgrade tasks, is executed. It copies the applications and configuration data from the source middle tier installation to the 10*g* (9.0.4) installation. Manual tasks may be necessary to complete the upgrade of some configurations.

   **See Also:** Chapter 3, "Upgrading the Middle Tier"

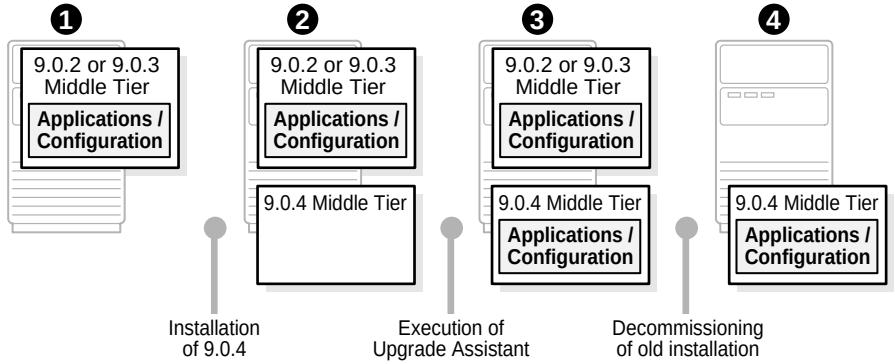
4. (Optional) The source installation is decommissioned.

---

**Note:** The installation of the 10*g* (9.0.4) middle tier will use some of the same port numbers as the Release 2 (9.0.2) or (9.0.3) middle tier. If you plan to use both middle tiers, you must re-assign port numbers in one of the middle tiers to avoid port conflicts. The components affected are Oracle Enterprise Manager, Oracle HTTP Server, and Oracle Application Server Web Cache.

---

***Figure 1–2 Middle Tier Upgrade***



## 1.3 The Infrastructure Upgrade Process

The Oracle Application Server Infrastructure provides centralized security and management services and a metadata repository to one or more application server instances. It is installed into a separate Oracle home from the application server instances, typically on a separate computer.

**See Also:** *Oracle Application Server 10g Administrator's Guide*

The Infrastructure comprises these parts:

- Identity Management — Oracle Application Server Single Sign-On, Oracle Internet Directory, Oracle Delegated Administration Services, Oracle Directory Integration and Provisioning, and Oracle Application Server Certificate Authority.
- Metadata Repository — Product metadata schemas (used by middle tier components such as OracleAS Portal and Oracle Application Server Wireless) Identity Management schemas (used by Identity Management components) and management schemas (used by components such as DCM)

The Oracle Application Server Infrastructure comprises these pieces: Identity Management (the Oracle Internet Directory and Oracle Application Server Single Sign-On) and the Metadata Repository (comprising multiple schemas within the underlying database).

The Oracle Application Server Upgrade Assistant is not used to upgrade the Infrastructure.

**Notes:** The illustrations below show an abstraction of Identity Management and the Metadata Repository, depicting the phases of the overall Infrastructure upgrade.

However, Identity Management consists of the Oracle Internet Directory and Oracle Application Server Single Sign-On binaries and their schemas in the Metadata Repository. In practice, Identity Management schemas in the Metadata Repository are upgraded in the Identity Management upgrade process, whereas all other schemas comprising the Metadata Repository are upgraded using individual scripts. See Section 5.1.2, "Understanding the Identity Management Upgrade Processes" on page 5-5 for illustrations.

Identity Management and the database containing the Metadata Repository can be on different computers.

Phases of the Infrastructure upgrade are listed below. At each phase, the configuration is operable. The Infrastructure can operate in the hybrid states described in Steps 2 and 3, and Steps 3 and 4 may be reversed.

1. The Release 2 (9.0.2) Oracle Application Server Infrastructure contains a database of version 9.0.1.3 by default.
2. A patch is applied to upgrade the database to version 9.0.1.5.
3. The Metadata Repository (schemas other than those for Oracle Internet Directory and Oracle Application Server Single Sign-On) is upgraded to 10*g* (9.0.4).
4. Identity Management is upgraded to 10*g* (9.0.4).

***Figure 1–3 Infrastructure Upgrade***

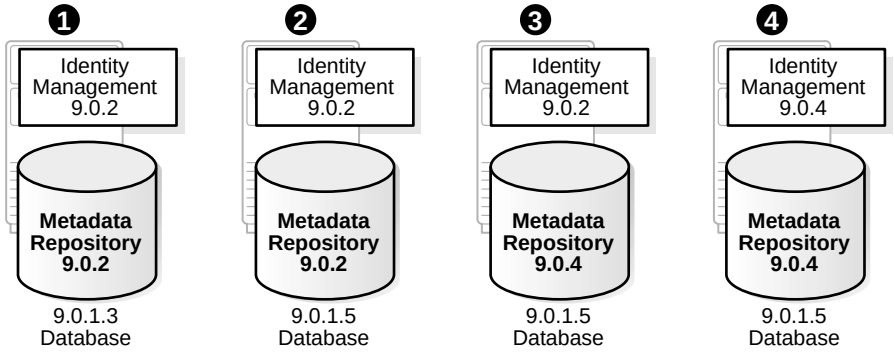
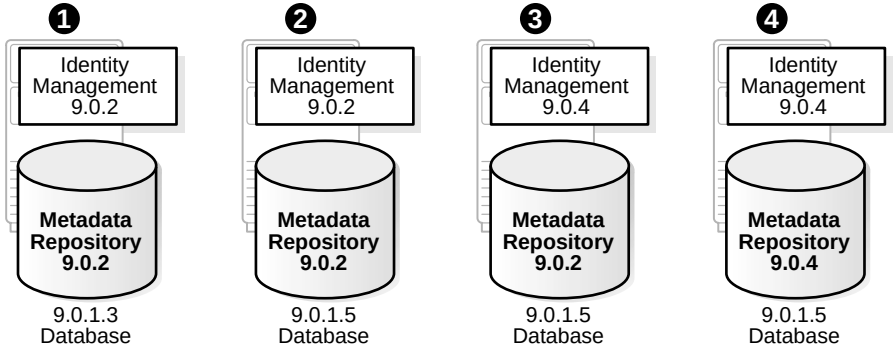



***Figure 1–4 Alternate Infrastructure Upgrade (Steps 3 and 4 Reversed)***



## 1.4 OracleAS Upgrade Assistant Upgrade Paths

This section describes the middle tier upgrade paths and requirements, and lists the components upgraded for each installation type. Alternative upgrade paths and exception cases are also discussed.

### Middle Tier Upgrade Paths

The OracleAS Upgrade Assistant can upgrade a middle tier from one Oracle home to another on the same computer, between these releases:

- Release 2 (9.0.2) to 10*g* (9.0.4)
- Release 2 (9.0.3) to 10*g* (9.0.4)

The OracleAS Upgrade Assistant will not display selections for Oracle homes of releases other than these.

The 10*g* (9.0.4) middle tier must be associated with the same Metadata Repository and Oracle Internet Directory as the Release 2 (9.0.2) middle tier.

### Middle Tier Components Upgraded by the OracleAS Upgrade Assistant

The OracleAS Upgrade Assistant upgrades these components within each installation type:

***Table 1–1 Components Upgraded by the OracleAS Upgrade Assistant (in processing order)***

| Component Upgraded by OracleAS Upgrade Assistant | Included in J2EE & Web Cache Installation Type? | Included in Portal & Wireless Installation Type? | Included in Business Intelligence & Forms Installation Type? |
|---|---|---|---|
| Oracle Process Management and Notification | Yes | Yes | Yes |
| Instance Configuration (iasschema.xml) | Yes | Yes | Yes |
| Oracle Application Server Containers for J2EE | Yes | Yes | Yes |
| Oracle HTTP Server | Yes | Yes | Yes |
| Oracle Application Server Web Cache | Yes | Yes | Yes |
| mod_plsql | Yes | Yes | Yes |
| Oracle Enterprise Manager (targets.xml) | Yes | Yes | Yes |
| Oracle Application Server Web Services UDDI Registry | No | Yes | Yes |

***Table 1–1 Components Upgraded by the OracleAS Upgrade Assistant (in processing order)***

| Component Upgraded by OracleAS Upgrade Assistant | Included in J2EE & Web Cache Installation Type? | Included in Portal & Wireless Installation Type? | Included in Business Intelligence & Forms Installation Type? |
|---|---|---|---|
| Oracle Ultra Search | No | Yes | Yes |
| Oracle Application Server Portal (Middle Tier) | No | Yes | Yes |
| Oracle Application Server Wireless | No | Yes | Yes |
| Oracle Application Server Forms Services | No | No | Yes |
| Oracle Application Server Discoverer | No | No | Yes |
| Oracle Application Server Reports Services | No | No | Yes |

### Source and Destination Installation Type Requirements

The OracleAS Upgrade Assistant can perform upgrades between Oracle Application Server instances of the same installation type only. For example, if you have a Release 2 (9.0.2) J2EE and Web Cache installation, you must upgrade to J2EE and Web Cache in 10*g* (9.0.4).

Upgrading the Unified Messaging installation type from Release 2 (9.0.2) UNIX systems is a special case, because Unified Messaging is not part of Oracle Application Server 10*g* (9.0.4). If you upgrade a Unified Messaging installation type, the OracleAS Upgrade Assistant applies the configuration to the Business Intelligence and Forms installation type in 10*g* (9.0.4).

See the Oracle Collaboration Suite to determine the upgrade path for a Unified Messaging installation.

### Alternative Upgrade Paths

If you have applications or configuration elements whose upgrade requirements are not addressed by the OracleAS Upgrade Assistant, you may use one of the following alternative paths:

- If there are applications deployed in Oracle Application Server Containers for J2EE (OC4J) instances of the"standalone" type—that is, the OC4J that is available for download from Oracle Technology Network in zip format—you must manually redeploy and reconfigure them.

**Note:** Standalone OC4J applications should not be configured in an Oracle Application Server Oracle home. Standalone applications should be redeployed in a managed OC4J instance in the 10*g* (9.0.4) Oracle home. See Section 3.3, "Upgrading a Standalone OC4J Instance" on page 3-6.

- To upgrade a standalone instance of OracleAS Web Cache Release 2 (9.0.2 or 9.0.3) to Oracle Application Server 10g (9.0.4), perform the steps described in Section 3.4, "Upgrading a Standalone OracleAS Web Cache Instance" on page 3-7 before you invoke the OracleAS Upgrade Assistant.
- To upgrade from Release 1 to Release 2, follow the instructions in *Migrating from Oracle9iAS Release 1 (1.0.2.2.x) to Release 2 (9.0.2)* in the Oracle9*i*AS Release 2 (9.0.2) documentation library.
- To upgrade from Release 1 (1.0.2.2) to 10*g* (9.0.4), you must first upgrade from Release 1 (1.0.2.2) to Release 2 (9.0.2), and then upgrade from Release 2 (9.0.2) to 10*g* (9.0.4).
- To upgrade Oracle Application Server Discoverer from Release 1 (1.0.2.2) to 10*g* (9.0.4), follow the instructions in the *Oracle Application Server Discoverer Configuration Guide* in the Oracle Application Server documentation library.
- The upgrade of applications that use JServ is not discussed in this guide. The material in the section titled "Migrating JServ to OC4J" in *Migrating from Oracle9iAS Release 1 (1.0.2.2.x) to Release 2 (9.0.2)*, in the Oracle9*i*AS Release 2 (9.0.2) documentation library, may be of some help in adapting such applications to the Oracle Application Server Containers for J2EE servlet environment in Oracle Application Server 10*g* (9.0.4).

### Expanding the Middle Tier Installation Type

You can expand to a larger (i.e., containing more components) middle tier installation type. For example, you can expand a J2EE and Web Cache middle tier to a Portal and Wireless middle tier.

**See Also:** *Oracle Application Server 10g Installation Guide*

To expand a middle tier in conjunction with an upgrade:

1. Upgrade the existing middle tier from Release 2 (9.0.2) or Release 2 (9.0.3) to 10*g* (9.0.4).
2. Expand the upgraded middle tier.

**See Also:** *Oracle Application Server 10g Installation Guide*

## 1.5 OracleAS Portal Repository Upgrade Paths

The Oracle9iAS Portal Repository can be upgraded using the instructions in this guide from any of the following versions: 9.0.2.0, 9.0.2.2, 9.0.2.3 or 9.0.2.6. For instructions on upgrading from other versions, see the the PortalCenter upgrades page at:

`http://portalcenter.oracle.com/upgrades`

# 2

# Planning an Upgrade

This chapter provides guidelines for planning an upgrade. It consists of the following sections:



## 2.1 Oracle Application Server Compatibility

As discussed in Chapter 1, a distributed Oracle Application Server configuration comprises one or more middle tiers and an Infrastructure. The Infrastructure comprises Identity Management and the Metadata Repository, each of which can be Release 2 (9.0.2) or 10*g* (9.0.4).

The upgrade process provides for compatibility between these units. Section 1.3, "The Infrastructure Upgrade Process" on page 1-4 discusses supported hybrid configurations, in which the Infrastructure combines Metadata Repository and Identity Management units of 10*g* (9.0.4) and Release 2 (9.0.2). The implication is, of course, that certain combinations of units are not supported.

This section identifies salient compatibility rules and, based on these, presents a sampling of supported and unsupported configurations. Understanding these, and the possible transition states between them, will help you plan a successful Oracle Application Server upgrade.

## 2.1.1 Compatibility Rules

Oracle Application Server compatibility is based on the following rules:

- A 10*g* (9.0.4) middle tier interoperates with a 10*g* (9.0.4) Metadata Repository.
- A 10*g* (9.0.4) middle tier interoperates with a 10*g* (9.0.4) Identity Management.
- A 10*g* (9.0.4) middle tier interoperates with a Release 2 (9.0.2) Metadata Repository.
- A 10*g* (9.0.4) middle tier interoperates with a Release 2 (9.0.2) Identity Management.
- All version rules are platform independent: two or more of the version units may reside on multiple operating systems without impact on their compatibility relationship.
- A 10*g* (9.0.4) Metadata Repository does not interoperate with a Release 2 (9.0.2) or Release 2 (9.0.3) middle tier.
- Clusters only host instances of the same release.

## 2.1.2 Oracle Application Server Configuration Examples

This section provides examples of configurations that are supported, and not supported, based on application of the compatibility rules.

### 2.1.2.1 Single Middle Tier and Single Infrastructure

Table 2–1 lists configurations that combine middle tier and Infrastructure units of different releases, and indicates whether they are supported.

***Table 2–1 Single Middle Tier and Single Infrastructure Compatibility***

| Configuration | Middle Tier Release | Metadata Repository[1] Release | Identity Management[2] Release | Supported? |
|---|---|---|---|---|
| **A** | 9.0.2 or 9.0.3 | 9.0.2 | 9.0.2 | Yes |
| **B** | 9.0.2 or 9.0.3 | 9.0.2 | 9.0.4 | Yes |
| **C** | 9.0.2 or 9.0.3 | 9.0.4 | 9.0.2 | No |
| **D** | 9.0.2 or 9.0.3 | 9.0.4 | 9.0.4 | No |
| **E** | 9.0.4 | 9.0.2 | 9.0.2 | Yes |
| **F** | 9.0.4 | 9.0.2 | 9.0.4 | Yes |

*Table 2–1 Single Middle Tier and Single Infrastructure Compatibility*

| Configuration | Middle Tier Release | Metadata Repository[1] Release | Identity Management[2] Release | Supported? |
|---|---|---|---|---|
| **G** | 9.0.4 | 9.0.4 | 9.0.2 | Yes |
| **H** | 9.0.4 | 9.0.4 | 9.0.4 | Yes |

[1] Excludes Oracle Internet Directory and Oracle Application Server Single Sign-On schemas. Consists of schemas for: Integration Platform, Oracle Certificate and Authentication, Oracle Ultra Search, Oracle Application Server Syndication Server, Oracle Application Server Web Services, Oracle Application Server Portal, Oracle Application Server Wireless, and Web Clipping.

[2] Identity Management comprises Oracle Internet Directory and Oracle Application Server Single Sign-On binaries and schemas.

#### 2.1.2.2 Multiple Middle Tiers and Single Infrastructure

Compatibility in multiple middle tiers and a single Infrastructure is very similar to the single middle tier and single Infrastructure case. The fact that a 10*g* (9.0.4) Metadata Repository does not operate with a Release 2 (9.0.2) or Release 2 (9.0.3) middle tier is the determining factor in compatibility; it is the most likely reason for incompatibility in this configuration.

#### 2.1.2.3 Multi-Platform Installations

A unit's operating system does not affect its compatibility relationships with units on other platforms. Any of the distributed configurations will run on a variety of platforms. For example, a 10*g* (9.0.4) middle tier on a Linux system may access Release 2 (9.0.2) Identity Management and a Metadata Repository on a Windows system. A 10*g* (9.0.4) middle tier on a Windows system may access a 10*g* (9.0.4) Identity Management and a Release 2 (9.0.2) Metadata Repository on a Solaris system, and so on.

## 2.1.3 Transitions Between Configurations

This section identifies possible transitions between configurations during upgrade. Figure 2–1 depicts the transitions. Table 2–2, Table 2–3, and Table 2–4 list the releases and the order in which the upgrade must be performed.

***Figure 2–1 Supported Transitions Between Configurations***

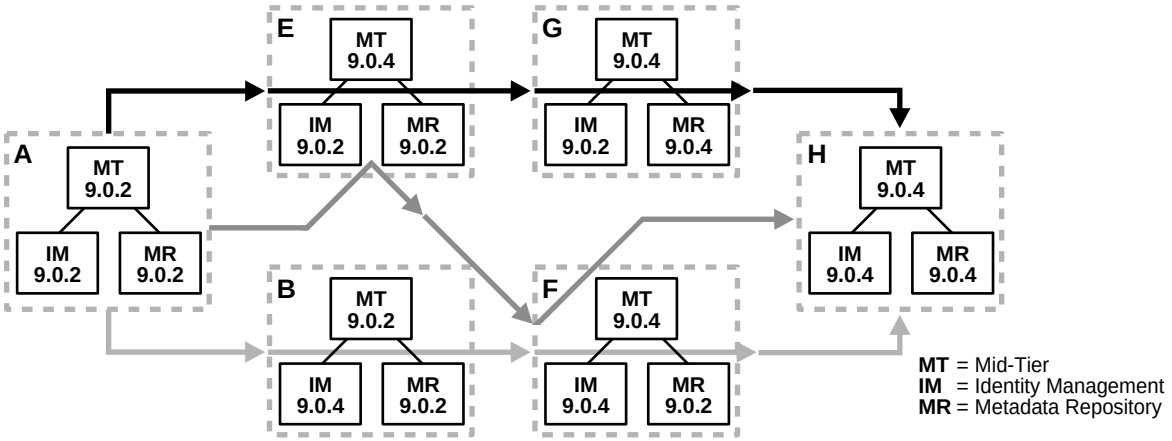



***Table 2–2 Transition from A to E to G to H[1]***

| Configuration | Middle Tier Release | Metadata Repository Release | Identity Management Release |
|---|---|---|---|
| **A** | 9.0.2 or 9.0.3 | 9.0.2 | 9.0.2 |
| **E** | 9.0.4 | 9.0.2 | 9.0.2 |
| **G** | 9.0.4 | 9.0.4 | 9.0.2 |
| **H** | 9.0.4 | 9.0.4 | 9.0.4 |

[1] The upgrade must be performed in this order: Middle tier, Metadata Repository, Identity Management.

*Table 2–3 Transition from A to E to F to H[1]*

| Configuration | Middle Tier Release | Identity Management Release | Metadata Repository Release |
|---|---|---|---|
| **A** | 9.0.2 or 9.0.3 | 9.0.2 | 9.0.2 |
| **E** | 9.0.4 | 9.0.2 | 9.0.2 |
| **F** | 9.0.4 | 9.0.4 | 9.0.2 |
| **H** | 9.0.4 | 9.0.4 | 9.0.4 |

[1] The upgrade must be performed in this order: Middle tier, Identity Management, Metadata Repository.

*Table 2–4 Transition from A to B to F to H[1]*

| Configuration | Identity Management Release | Middle Tier Release | Metadata Repository Release |
|---|---|---|---|
| **A** | 9.0.2 | 9.0.2 or 9.0.3 | 9.0.2 |
| **B** | 9.0.4 | 9.0.2 or 9.0.3 | 9.0.2 |
| **F** | 9.0.4 | 9.0.4 | 9.0.2 |
| **H** | 9.0.4 | 9.0.4 | 9.0.4 |

[1] The upgrade must be performed in this order: Identity Management, Middle tier, Metadata Repository.

# 2.2 Determining an Upgrade Strategy

This section provides information that will help you determine how to approach the Oracle Application Server upgrade. It contains these topics:



## 2.2.1 Upgrade Process Overview

This section describes the upgrade process in entirety and identifies the parts of the upgrade that are automated by the Upgrade Assistant or a script, and those that are manual. It contains these topics:



### 2.2.1.1 Upgrade Tasks

Table 2–5, "Oracle Application Server Upgrade Tasks", is a task list for the entire upgrade process. Some steps do not apply all configurations; for example,Step 6 and Steps 8 through 11 are not applicable to configurations that do not use an Infrastructure.

***Table 2–5 Oracle Application Server Upgrade Tasks***

| Step | Task | Instructions |
|---|---|---|
| 1 | Install the Oracle Application Server 10*g* (9.0.4) middle tier, which must use the same Metadata Repository and Oracle Internet Directory as the Release 2 (9.0.2) middle tier being upgraded. | *Oracle Application Server 10g Installation Guide* |
| 2 | Stop the source middle tier instance. | Section 3.1.2, "Stopping OracleAS Instances" on page 3-4 |
| 3 | Ensure that the Infrastructure is running and accessible. | Section 3.5.2, "Starting the Infrastructure" on page 3-10 |

***Table 2–5 Oracle Application Server Upgrade Tasks***

| Step | Task | Instructions |
|---|---|---|
| 4 | Execute the OracleAS Upgrade Assistant. | Section 3.5, "Using the OracleAS Upgrade Assistant" on page 3-8 |
| 5 | Perform the manual steps required to complete the middle tier upgrade. | Section 3.8, "Completing the Upgrade" on page 3-30 |
| 6 | Repeat Steps 1, 2, 3, 4 and 5 for each middle tier associated with the same Infrastructure. | *Oracle Application Server 10g Installation Guide*<br>Section 3.1.2, "Stopping OracleAS Instances" on page 3-4<br>Section 3.5.2, "Starting the Infrastructure" on page 3-10<br>Section 3.5, "Using the OracleAS Upgrade Assistant" on page 3-8<br>Section 3.8, "Completing the Upgrade" on page 3-30 |
| 7 | Apply the patch to upgrade the database to Release 9.0.1.5. | Section 4.2, "Preparing to Upgrade the Metadata Repository" on page 4-3 |
| 8 | Install or upgrade the Oracle Application Server 10*g* (9.0.4) Infrastructure. | Section 5.1, "Upgrading Identity Management" on page 5-2 |
| 9 | Execute the scripts to upgrade the Infrastructure database schemas for the Metadata Repository Container, Integration Platform, Oracle Certificate and Authentication, Syndication Server, Web Clipping, Oracle Ultra Search, and Oracle Application Server Web Services. | Section 4.4, "Executing Metadata Repository Upgrade Scripts" on page 4-12 |
| 10 | Upgrade the Portal repository. | Section 4.5, "Upgrading the OracleAS Portal Repository" on page 4-31 |
| 11 | Perform the manual tasks (if applicable) to complete the upgrade of Oracle Application Server Single Sign-On and Oracle Internet Directory, which occurred during Infrastructure installation. | Section 5.4, "Performing Infrastructure Post-Upgrade Tasks" on page 5-48 |
| 12 | Execute the scripts to upgrade schemas in customer databases, if applicable. | Section 4.6, "Upgrading Schemas in Customer Databases" on page 4-64 |
| 13 | Modify middle tier components to adopt 10*g* (9.0.4) Infrastructure functionality. | Section 4.7, "Activating 10g (9.0.4) Functionality for UDDI Applications" on page 4-65 |

#### 2.2.1.2 Middle-Tier Components Upgraded by the Oracle Application Server Upgrade Assistant

The OracleAS Upgrade Assistant upgrades the following middle-tier component configurations.

> **Note:** Oracle9*i*AS Clickstream Intelligence is not included in 10*g* (9.0.4), so its upgrade is not documented in this guide.

All upgrade processing performed by the OracleAS Upgrade Assistant is described in Appendix A, "Component Upgrade Process Reference", in the following sections:

- Section A.1.1, "The Oracle Process Manager and Notification Server (OPMN) Upgrade Process" on page A-2
- Section A.1.2, "The Instance Configuration Data Upgrade Process" on page A-3
- Section A.1.3, "The Oracle Application Server Containers for J2EE (OC4J) Upgrade Process" on page A-4
- Section A.1.4, "The Oracle HTTP Server Upgrade Process" on page A-5
- Section A.1.5, "The Oracle Application Server Web Cache Upgrade Process" on page A-8
- Section A.1.6, "The mod_plsql Upgrade Process" on page A-10
- Section A.1.7, "The Oracle Enterprise Manager Upgrade Process" on page A-10
- Section A.1.8, "The Oracle Application Server Web Services UDDI Registry Upgrade Process" on page A-11
- Section A.1.9, "The Oracle Ultra Search Upgrade Process" on page A-12
- Section A.1.10, "The OracleAS Portal Middle Tier Upgrade Process" on page A-12
- Section A.1.11, "The Oracle Application Server Wireless Upgrade Process" on page A-13
- Section A.1.12, "The Oracle Application Server Forms Services Upgrade Process" on page A-15
- Section A.1.13, "The Oracle Application Server Discoverer Upgrade Process" on page A-16
- Section A.1.14, "The Oracle Application Server Reports Services Upgrade Process" on page A-17

#### 2.2.1.3 Infrastructure Schemas Upgraded by Scripts

Some components have schemas in the Infrastructure database that contain product metadata and user data. These are upgraded by the installer, or by a script provided on the Repository Creation Assistant CD or on MetaLink. There may also be manual processes associated with these upgrades.

All processing by the schema upgrade scripts on the Repository Creation Assistant CD or is described in Appendix A, "Component Upgrade Process Reference", in the following sections:

- Section A.2.1, "The Identity Management Upgrade Process" on page A-19
- Section A.2.2, "The Metadata Repository Container Schema Upgrade Process" on page A-20
- Section A.2.3, "The Process Connect Upgrade Process" on page A-21
- Section A.2.4, "The Oracle Application Server Certificate Authority Upgrade Process" on page A-21
- Section A.2.5, "The Oracle Ultra Search Schema Upgrade Process" on page A-21
- Section A.2.6, "The OracleAS Portal Schema Upgrade Process" on page A-22
- Section A.2.7, "The Oracle Application Server Syndication Server Schema Upgrade Process" on page A-24
- Section A.2.8, "The Oracle Application Server Web Services UDDI Registry Schema Upgrade Process" on page A-24
- Section A.2.9, "The Web Clipping Upgrade Process" on page A-25
- Section A.2.10, "The Oracle Application Server Wireless Schema Upgrade Process" on page A-26

#### 2.2.1.4 Manual Upgrade Tasks

Manual tasks and their location in this guide are listed below:

##### Middle Tier Manual Upgrade Tasks

Section 3.8.1, "Completing the Oracle HTTP Server Upgrade" on page 3-31

Section 3.8.2, "Completing the Oracle Application Server Containers for J2EE (OC4J) Upgrade" on page 3-33

Section 3.8.3, "Completing the OracleAS Web Cache Upgrade" on page 3-43

Section 3.8.4, "Completing the OracleAS Portal Middle Tier Upgrade" on page 3-45

Section 3.8.5, "Completing the Oracle Application Server Discoverer Viewer Upgrade" on page 3-58

Section 3.8.6, "Completing the Oracle Application Server Reports Services Upgrade" on page 3-60

Section 3.8.7, "Completing the OracleAS Wireless Upgrade" on page 3-62

Section 3.8.8, "Completing the OracleAS Forms Services Upgrade" on page 3-66

Section 3.8.9, "Upgrading the tnsnames.ora File" on page 3-68

Section 3.9, "Upgrading OracleAS InterConnect" on page 3-69 (no automated upgrade exists for this component)

**Infrastructure Manual Upgrade Tasks**

Section 4.5.8, "Completing the OracleAS Portal Repository Upgrade" on page 4-42

Section 5.4.1, "Completing the Oracle Internet Directory Upgrade" on page 5-48

Section 5.4.2, "Completing the Oracle Application Server Single Sign-On Upgrade" on page 5-52

Section 5.4.3, "Completing the Oracle Application Server Wireless Upgrade" on page 5-57

## 2.2.2 Planning for System Downtime

This section contains information that will help you answer the following questions as you plan the Oracle Application Server upgrade:

- How much downtime should be allocated to upgrade? to troubleshooting?
- What parts of the system are subject to downtime?
- When will the downtime occur?

### 2.2.2.1 How Long Will the Upgrade Take?

The duration of upgrade preparation tasks and upgrade processing is of concern when considering downtime. This section provides estimates of the duration of the upgrade of a basic configuration. The estimates do not account for troubleshooting time; they are based on an error-free upgrade.

*Table 2–6 Middle Tier Upgrade Duration Estimates*

| Operation | J2EE & Web Cache | Portal & Wireless | Business Intelligence & Forms |
|---|---|---|---|
| **10*g*** (9.0.4) **middle tier installation**:A 10*g* (9.0.4) middle tier must be installed on the same computer as the Release 2 (9.0.2) or Release 2 (9.0.3) middle tier. | 30 minutes | 60 minutes[1] | 90 minutes[1] |
| **Pre-upgrade:** The source and destination instances must be stopped. Backing up the instances is optional and is not included in these estimates. | 20 minutes | 20 minutes | 20 minutes |
| **OracleAS Upgrade Assistant execution:** Execution time depends on source configuration; for example, the number and size of J2EE applications deployed may affect the duration significantly. This estimate assumes a basic configuration. | 20 minutes | 30 minutes | 40 minutes |
| **Post-upgrade:** This includes starting the upgraded instance and performing basic verification tests. | 20 minutes | 30 minutes | 40 minutes |
| **Total** | 1 hour, 45 minutes | 2 hours, 35 minutes | 3 hours, 25 minutes |

[1]The first 10*g* (9.0.4) instance that configures Oracle Application Server Wireless against a Release 2 (9.0.2) Metadata Repository upgrades the schema in that repository. This may increase the length of this operation significantly. If Oracle Application Server Wireless is running on multiple middle tiers, Oracle Application Server Wireless must be stopped on all of those middle tiers before performing this operation. See Appendix A.1.11, "The Oracle Application Server Wireless Upgrade Process" on page A-13 for more information.

*Table 2–7 Infrastructure Upgrade Duration Estimates*

| Operation | Metadata Repository | Identity Management | Full Infrastructure[1] |
|---|---|---|---|
| **Database backup:** The database should be backed up with the user's preferred procedure (for example, `ufsdump`). | 1 hour | Not applicable. | Not applicable. |
| **Oracle home backup:** The Infrastructure Oracle home should be backed up. | Not applicable. | 1 hour | 1 hour |
| **Database patch:** If the database containing the Metadata Repository is version 9.0.1.3, it must be patched to version 9.0.1.5. | 2 hours | 2 hours | 2 hours |
| **Metadata Repository Container upgrade:** The Metadata Repository Container upgrade script is executed from the Repository Configuration Assistant CD-ROM. [2] | 5 minutes | 5 minutes | 5 minutes |
| **Component schema and OracleAS Portal upgrade:**Component schemas in the Metadata Repository and OracleAS Portal are upgraded via scripts. | 2 hours | Not applicable | 3 hours |
| **Total:** | 5 hours, 35 minutes | 5 hours, 10 minutes | 8 hours, 40 minutes |

*Table 2–7 Infrastructure Upgrade Duration Estimates*

| Operation | Metadata Repository | Identity Management | Full Infrastructure[1] |
|---|---|---|---|
| **Metadata Repository Container post-upgrade:** The middle tier components relying on upgraded schemas are re-started. | 30 minutes | Not applicable | 30 minutes |
| **Installer preparation:** The Release 2 (9.0.2) Infrastructure software must be in the correct state (instance stopped, database started, database listener stopped, Oracle Internet Directory started, etc.) | Not applicable | 20 minutes | 20 minutes |
| **Installation:** The Oracle Universal Installer is executed in upgrade mode, installing the 10*g* (9.0.4) Infrastructure into a new Oracle home and executing all post-installation configuration tools. | Not applicable | 45 minutes | 45 minutes |
| **Identity Management post-upgrade:** Post-upgrade tasks, such as Oracle Application Server Single Sign-On DAD upgrade, as required | Not applicable | 1 hour | 1 hour |
| **Total:** | 5 hours, 35 minutes | 5 hours, 10 minutes | 8 hours, 40 minutes |

[1] The upgrade duration of the Metadata Repository and Identity Management may be shorter than that of the sum of the durations required to upgrade each piece individually, since common tasks need only be executed once.

[2] Although this estimate is based on execution of all scripts, it is not mandatory that all components in the Metadata Repository are upgraded; thus, the fewer components upgraded, the shorter the duration of this operation.

## 2.2.3 System Availability During Upgrade

This section outlines the steps involved in the upgrade process when two middle tier instances use a single Infrastructure instance. The upgrade process was designed for high availability. As shown in Figure 2–2, full system downtime occurs only in step 6 of the process (if the system relies on an Infrastructure).

The progression of system states during the upgrade process is detailed below:

1. The Release 2 (9.0.2) system is functioning at full capacity, with clients connecting through both middle tiers.
2. The first middle tier is stopped, in preparation for upgrade. Clients can no longer connect through the first middle tier, but continue to connect through the second middle tier.
3. The first middle tier is upgraded to 10*g* (9.0.4). (See Section 1.2, "The Middle Tier Upgrade Process" on page 1-3.) Clients can now lconnect through both middle tiers.

4. The second middle tier is stopped, in preparation for upgrade. Clients can no longer connect through the second middle tier, but continue to connect through the first middle tier.
5. The second middle tier is upgraded to 10*g* (9.0.4). Clients can now connect through both middle tiers.
6. The Infrastructure is stopped in preparation for upgrade. Applications that are dependent on the Infrastructure are unavailable now.
7. The Infrastructure is upgraded to 10*g* (9.0.4). (See Section 1.3, "The Infrastructure Upgrade Process" on page 1-4.) Clients can now connect to the fully upgraded system.

---

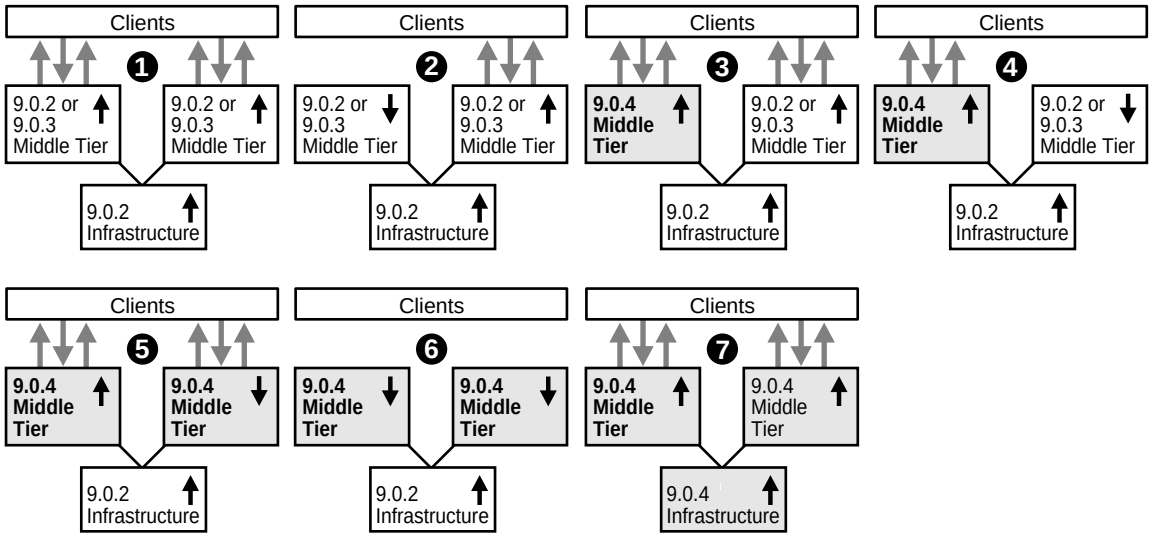
**Notes:** Only two middle tiers are shown in the figure for simplicity's sake, but in practice there may be many more. The more middle tiers in service, the lower the system capacity loss in downtime during upgrade. If there are two middle tiers, 50% capacity is lost when one is stopped for upgrade. If there are four middle tiers, only 25% capacity is lost when one is stopped for upgrade.

In the figure, "Clients" may refer to a load balancer. Users need not be aware of middle tier downtime.

For middle tiers running Oracle Application Server Wireless, there are exceptions to this upgrade process. Appendix A.1.11.1, "Upgrade of Oracle Application Server Wireless Middle Tiers and Wireless Schema" on page A-14.

---

***Figure 2–2 System Availability During Upgrade***




## 2.2.4 Creating Backups

The upgrade process only alters 10*g* (9.0.4) installations; the source instances are always left unchanged. You may want to create a backup of a 10*g* (9.0.4) Oracle home so that it can be restored to a pre-upgrade (that is, newly installed) state. Restoring from backups may be an efficient alternative to reinstalling the entire instance, in the event that upgrade results are unsatisfactory. A useful backup might include:

- Directories for specific components. See Table B–2, "Files Containing Upgrade Data (Sorted by Path)" on page B-7.
- The entire Oracle home. You can use the Oracle Application Server Backup and Recovery tool and documentation to do this.

**See Also:** *Oracle Application Server 10g Administrator's Guide*

# 3

# Upgrading the Middle Tier

This chapter explains how to upgrade the middle tier of your OracleAS installation. It includes instructions on how to prepare your system for a successful upgrade, and how to start and use the Oracle Application Server Upgrade Assistant. It also details the manual tasks you may have to perform on individual components after the OracleAS Upgrade Assistant has finished processing. Complete manual upgrade procedures are included for components, or parts of components, that are not upgraded by the OracleAS Upgrade Assistant. All of these steps must be performed for each middle tier in the installation.

---

**Note:** Only the 10*g* (9.0.4) Oracle home is changed in the upgrade process. This makes it possible to revert to using the source Oracle home.

---

**See Also:** Section 1.2, "The Middle Tier Upgrade Process" on page 1-3 for a general overview of the middle tier upgrade process.

The chapter is divided into the following major sections:

# 3.1 Preparing to Upgrade: Common Requirements for All Components

This section provides the steps you must take to prepare the 10*g* (9.0.4) middle tier installation for upgrade.



## 3.1.1 Requirements for Installing an OracleAS 10*g* (9.0.4) Middle Tier

If you are upgrading a middle tier that uses an Infrastructure, certain pre-installation requirements exist. You must ensure that these are met before you install an Oracle Application Server 10*g* (9.0.4) middle tier. Otherwise, the 10*g* (9.0.4) installation will not function with a Release 2 (9.0.2) Infrastructure. Specifically, before you can use the Release 2 (9.0.2) Infrastructure, you must update an entry in the Release 2 (9.0.2) Oracle Internet Directory.

> **See Also:** *Oracle Application Server 10g Installation Guide*, "Update an Entry in the 9.0.2 Oracle Internet Directory".

> **See Also:** *Oracle Application Server 10g Installation Guide*, Chapter 3.

Install OracleAS 10*g* (9.0.4) middle tier, adhering to the following requirements:

- The Oracle Application Server 10*g* (9.0.4) middle tier must be on the same computer as the Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) middle tier.
- The Oracle Application Server 10*g* (9.0.4) middle tier must be installed by the same operating system user that installed the Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) middle tier.
- The Oracle Application Server 10*g* (9.0.4) middle tier must be in a separate Oracle home from the Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) middle tier.
- The Oracle Application Server 10*g* (9.0.4) middle tier must be of the same installation type as the Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) middle tier. (The exception is that an Oracle Application Server 10*g* (9.0.4) Business Intelligence and Forms installation type must be installed to upgrade an Oracle9*i*AS Release 2 (9.0.2) Unified Messaging installation type.)
- If Identity Management is used, the Oracle Application Server 10*g* (9.0.4) middle tier must use the same Identity Management as the Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) middle tier.
- The Oracle Application Server 10*g* (9.0.4) middle tier must use the same Metadata Repository as the Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) middle tier. (The Metadata Repository will always be Oracle9*i*AS Release 2 (9.0.2)).
- Only components that are configured in both the source and the destination Oracle homes will be upgraded. The exception to this is that OracleAS Portal will be upgraded if it is configured in the destination instance. Note that the OracleAS Portal component will not be fully configured in the Oracle Application Server 10*g* (9.0.4) instance until after the OracleAS Upgrade Assistant is run. This means that the Oracle9*i*AS Portal must only be accessed using the Release 2 (9.0.2) middle tier until after the middle tier upgrade process is complete.

For information on compatibility with other software and related installation requirements:

**See Also:** *Oracle Application Server 10g Installation Guide*

### 3.1.2 Stopping OracleAS Instances

Stop all processes in the source and destination Oracle homes with these commands shown below.

- In the source Oracle home:
  1. Stop the Oracle Enterprise Manager Application Server Control:

     **<*source_MT_OH*>/bin/emctl stop**

  2. Stop OPMN and processes managed by it with this command:

     **<*source_MT_OH*>/opmn/bin/*opmnctl stopall***

  3. Stop OracleAS Web Cache with this command:

     **<*source_MT_OH*>/bin/*webcachectl stop***

  4. Stop all other running processes in the source Oracle home.
- In the destination Oracle home:
  1. Stop the Oracle Enterprise Manager Application Server Control with this command:

     **<*destination_MT_OH*>/bin/emctl stop iasconsole**

  2. Stop OPMN and processes managed by it with this command:

     **<*destination_MT_OH*>/opmn/bin/*opmnctl stopall***

  3. Stop all other running processes in the destination middle tier Oracle home.

### 3.1.3 Preparing to Upgrade a Middle Tier that Uses an Infrastructure

Before you upgrade any middle tier that uses an Infrastructure, the Infrastructure must be started and accessible. If the Infrastructure is stopped, certain upgrade processes will fail (Oracle Application Server Containers for J2EE, OracleAS Portal, and OracleAS Wireless).

# 3.2 Preparing to Upgrade: Requirements for Specific Components

This section outlines pre-upgrade steps you need to perform for specific components of the middle tier.

Section 3.2.1, "Preparing to Upgrade OracleAS Wireless Middle Tiers" on page 3-5

Section 3.2.2, "Optional: Increasing JVM Memory for Large OC4J Upgrades" on page 3-5

## 3.2.1 Preparing to Upgrade OracleAS Wireless Middle Tiers

If you are upgrading one or more OracleAS Wireless middle tiers, you must perform the steps below before starting the OracleAS Upgrade Assistant:

> **See Also:** Appendix A.1.11.1, "Upgrade of Oracle Application Server Wireless Middle Tiers and Wireless Schema" on page A-14.

1. Stop all OracleAS Wireless middle tiers in the farm, including any Oracle Collaboration Suite middle tiers that are configured to run OracleAS Wireless.
2. If you will be operating Release 2 (9.0.2) OracleAS Wireless middle tiers in combination with 10*g* (9.0.4) middle tiers, obtain the OracleAS Wireless 9.0.2.8.0 patch set from MetaLink and apply the patch to all of the Release 2 (9.0.2) OracleAS Wireless middle tiers.

   > **See Also:** *Oracle Application Server 10g Installation Guide*

3. Create a backup of the Oracle9*i*AS Wireless Release 2 (9.0.2) schema in the Infrastructure. The backup is needed because this schema is upgraded during the installation of the Oracle Application Server 10*g* (9.0.4) Wireless middle tier.

   > **See Also:** *Oracle Application Server 10g Installation Guide*

## 3.2.2 Optional: Increasing JVM Memory for Large OC4J Upgrades

If you are upgrading a large number of applications or a large number of OC4J instances, it might be helpful to increase the memory for the extract phase of the OC4J upgrade. The extract phase of the upgrade process starts a new Java process (that is, it has a new JVM), and you can configure its minimum and maximum memory. To do this, you configure the `JavaVM` property in the `<destination_MT_OH>/ias/upgrade/Oc4jPlugin.cfg` file.

***Example 3–1 JavaVM Property in Oc4jPlugin.cfg File***

```
<JavaVM>
    <JVMproperties property="Xms256m"/>
    <JVMproperties property="Xmx512m"/>
</JavaVM>
```

The default values of 256MB minimum and 512MB maximum are shown in Example 3–1; however, 1024MB is a plausible upper limit for upgrading several OC4J instances and many large applications.

## 3.3 Upgrading a Standalone OC4J Instance

Upgrading a standalone OC4J instance (for example, an instance created with the OC4J that is available for download from Oracle Technology Network in zip format) in Oracle9*i*AS Release 2 (9.0.2) or Release 2 (9.0.3) is a manual process. The OracleAS Upgrade Assistant does not upgrade such instances. You must re-deploy the applications in the Oracle Application Server 10*g* (9.0.4) Oracle home.

1. Create an OC4J instance to host the applications in the 10*g* (9.0.4) Oracle home, using the Oracle Enterprise Manager Application Server Control.
2. Configure the OC4J instance with any system properties or environment variables that are unique to the standalone instance.
3. Identify all command-line arguments from the `java -jar` command line in the standalone instance. In Oracle Application Server, the Oracle Process Management and Notification subsystem is responsible for starting and stopping OC4J.
4. Move all command-line arguments to `<destination_MT_OH>opmn/conf/opmn.xml`.
5. Move any properties specified in properties files in the standalone instance to `<destination_MT_OH>opmn/conf/opmn.xml`.
6. Re-deploy the Release 2 (9.0.2) or Release 2 (9.0.3) standalone applications (EARfiles) in the Oracle Application Server 10*g* (9.0.4) Oracle home.

## 3.4 Upgrading a Standalone OracleAS Web Cache Instance

You can upgrade a standalone instance of OracleAS Web Cache Release 2 (9.0.2 or 9.0.3) to Oracle Application Server 10g (9.0.4) using the OracleAS Upgrade Assistant. (A standalone instance is one installed from a kit that included only this product; it was not installed as part of an Oracle Application Server installation.)

However, you must perform the following steps *before* running the OracleAS Upgrade Assistant:

1. Edit the OracleAS Upgrade Assistant configuration file (iasua.xml), which is located at:

   ```
   <destination_MT_OH>/upgrade/iasua.xml
   ```

2. In the configuration file, delete the `<Plugin>` entries that apply to other Oracle Application Server components. Do not delete the OracleAS Web Cache `<Plugin>` entry, shown in the following example:

   ```
   <Plugin DisplayName="OracleAS Web Cache"
           ShortName="WebCache"
           Class="oracle.ias.upgrade.config.webcache.WebcacheConfigUpgrade">
   <Install Id="oracle.calypso"/>
   </Plugin>
   ```

3. Save the modified configuration file.

4. Use the OracleAS Upgrade Assistant as described in Section 3.5.4, "Performing an Upgrade with the OracleAS Upgrade Assistant (Graphical User Interface (GUI) Version)" on page 3-11 or Section 3.5.5, "Performing an Upgrade with the OracleAS Upgrade Assistant (Command-line Version)" on page 3-19.

# 3.5 Using the OracleAS Upgrade Assistant

This section provides instructions for using the graphical user interface or the command-line version to perform an upgrade, and describes properties you can configure to specify logging behaviors for the OracleAS Upgrade Assistant. These topics are included:

Section 3.5.1, "Specifying Logging Behaviors for the OracleAS Upgrade Assistant" on page 3-8

Section 3.5.2, "Starting the Infrastructure" on page 3-10

Section 3.5.3, "Starting the OracleAS Upgrade Assistant To Use Multiple Oracle Universal Installer Inventory Locations" on page 3-10

Section 3.5.4, "Performing an Upgrade with the OracleAS Upgrade Assistant (Graphical User Interface (GUI) Version)" on page 3-11

Section 3.5.5, "Performing an Upgrade with the OracleAS Upgrade Assistant (Command-line Version)" on page 3-19

## 3.5.1 Specifying Logging Behaviors for the OracleAS Upgrade Assistant

You can configure the logging behavior of the OracleAS Upgrade Assistant by setting properties in the `<destination_MT_OH>/upgrade/iasua.properties` file. The logging properties and their uses are:

- **log.level** — Use this property to specify the level of logging for the OracleAS Upgrade Assistant and all component plug-ins. For example, `log.level=NOTIFICATION` would set the logging level for all components upgraded by the OracleAS Upgrade Assistant to NOTIFICATION.
- **<plug-in name>.log.level** — Use this property to specify the level of logging for a specific component plug-in, used to override the log.level property for a given component upgrade. For example, `OC4J.log.level=TRACE` would set the logging level for the Oracle Application Server Containers for J2EE upgrade to TRACE, even if the log level for the OracleAS Upgrade Assistant was set to NOTIFICATION.
- **log.append** — Use this property to specify whether to append log entries to the existing log file or create a new log file. For example, `log.append=TRUE` would append log entries to the existing log file. (TRUE is the default.)

**Note:** Property names are case sensitive. Property values are case insensitive.

***Table 3–1 Logging Properties for the OracleAS Upgrade Assistant***

| Property Name | Description | Valid Values |
|---|---|---|
| **log.level** | Level of logging for the OracleAS Upgrade Assistant and all component plug-ins | WARNING<br>NOTIFICATION<br>ERROR<br>TRACE<br>DEBUG<br>OFF<br>INTERNAL_ERROR |
| **<plug-in name>.log.level** | Level of logging for a specific component plug-in, used to override the log.level property for a given component upgrade. | OPMN<br>InstanceConfig<br>OHS<br>OC4J<br>WebCache<br>modplsql<br>EM<br>UDDI<br>UltraSearch<br>Portal<br>Wireless<br>Forms<br>Discoverer<br>Reports |
| **log.append** | Specifies whether to append log entries to the existing log file or create a new log file. | TRUE<br>FALSE |

### 3.5.2 Starting the Infrastructure

If an Infrastructure is in use, it must be started in order for the OracleAS Upgrade Assistant to operate correctly.

> **See Also:** *Oracle Application Server 10g Administrator's Guide*, Chapter 3, "Starting an Infrastructure".

### 3.5.3 Starting the OracleAS Upgrade Assistant To Use Multiple Oracle Universal Installer Inventory Locations

The Oracle Universal Installer creates an inventory file, `/var/opt/oracle/oraInst.loc`, when it installs Oracle products. This file contains the location (full path) of the Oracle Application Server instance, and the group name of the user who installed it. The OracleAS Upgrade Assistant populates its source Oracle home drop-down list (shown in Figure 3–2, "OracleAS Upgrade Assistant Oracle Homes Screen") with the information from this file, the default inventory file. Additional inventory files are sometimes created after installation for the purpose of managing Oracle homes independently (thereby circumventing the Oracle Universal Installer features that track all Oracle homes in a single inventory).

If there are multiple inventory location files on the computer on which you are performing an upgrade, and you want to be able to select a non-default inventory location, you must start the OracleAS Upgrade Assistant with the `-invptrloc` argument, specifying the inventory location file(s) for the Oracle homes involved in the upgrade. The syntax for starting the OracleAS Upgrade Assistant for multiple inventory locations is provided below.

**Graphical User Interface (GUI) Version:**

```
iasua.sh [[-invptrloc <Oracle Universal Installer inventory
pointer file>]...]
```

**Command-line Version:**

```
iasua.sh -sourcehome <9.0.2 or 9.0.3 Oracle home path>
[[-invptrloc <Oracle Universal Installer inventory pointer
file>]...] [-verbose] [-noprompt]
```

### 3.5.4 Performing an Upgrade with the OracleAS Upgrade Assistant (Graphical User Interface (GUI) Version)

This section provides step-by step instructions for using the OracleAS Upgrade Assistant GUI version to perform an upgrade.

1. Start the OracleAS Upgrade Assistant with the command:

   **`<destination_MT_OH>/upgrade/iasua.sh`**

   > **Note:** Use the `-invptrloc` argument, described in Section 3.5.3, "Starting the OracleAS Upgrade Assistant To Use Multiple Oracle Universal Installer Inventory Locations" on page 3-10 if there are multiple inventories involved in the upgrade.

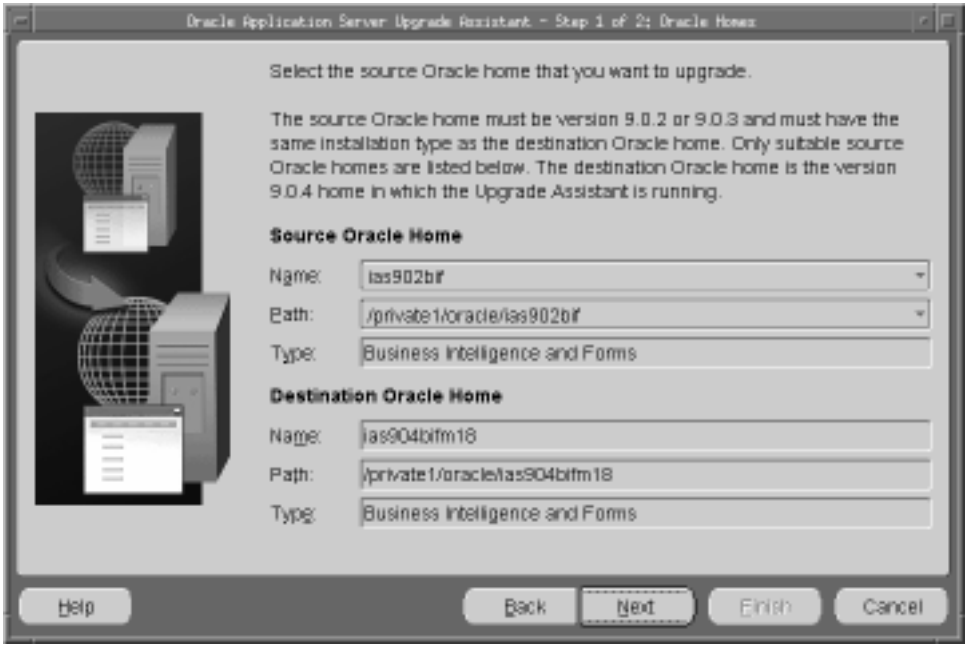
2. The Welcome screen appears as shown in Figure 3–1. Click **Next**.

***Figure 3–1 OracleAS Upgrade Assistant Welcome Screen***

3. The Oracle Homes screen appears as shown in Figure 3–2. The Source Oracle Home drop-down list contains the names of Release 2 (9.0.2) and Release 2 (9.0.3) Oracle homes found in the inventory of Oracle products on the current computer. The destination Oracle home is the 10*g* (9.0.4) Oracle home in which the OracleAS Upgrade Assistant is running.

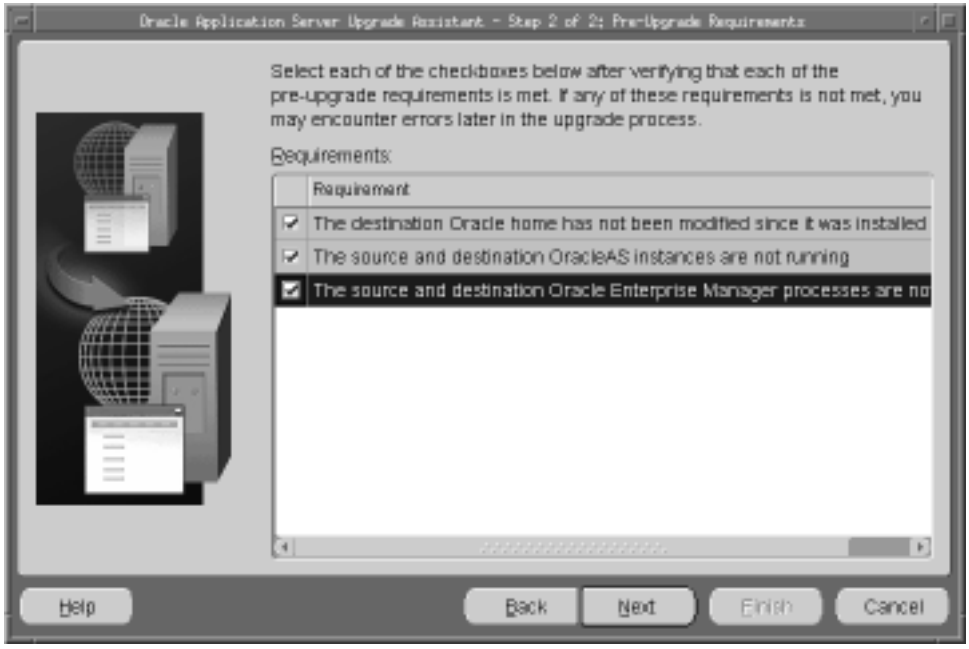
   Select the source Oracle home from the drop-down list. Then click **Next**.

***Figure 3–2 OracleAS Upgrade Assistant Oracle Homes Screen***




4. The Pre-Upgrade Requirements screen appears as shown in Figure 3–3. Ensure that all requirements are fulfilled, and check the box for each. The **Next** button is active only when all boxes are checked.

***Figure 3–3 OracleAS Upgrade Assistant Requirements Screen***

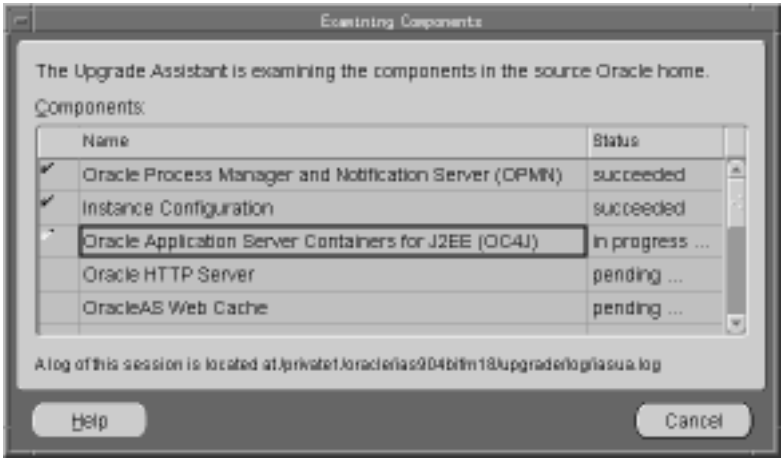



**5.** Click **Next**. The Examining Components dialog box appears as shown in Figure 3–4. The OracleAS Upgrade Assistant examines each component in the source Oracle home to determine whether it needs to be upgraded. The Status column for each component contains one of the following:

***Table 3–2 OracleAS Upgrade Assistant Component Examination Status***

| Status | Meaning |
|---|---|
| in progress... | The OracleAS Upgrade Assistant is examining the component's upgrade items. |
| pending... | The component will be examined when the OracleAS Upgrade Assistant finishes examining the current component. |
| succeeded | All of the component's upgrade items are valid for upgrade. |
| failed | The component has upgrade items that are missing or did not meet upgrade criteria. The OracleAS Upgrade Assistant cannot upgrade the component. |

***Figure 3–4 OracleAS Upgrade Assistant Examining Components Dialog Box***




**6.** If one or more components failed, the Examination Failure Warning dialog box appears as shown in Figure 3–5. Continue with Step 7.

If all components succeeded, the Summary screen appears as shown in Figure 3–6. Continue with Step 8.

***Figure 3–5 OracleAS Upgrade Assistant Examination Failure Warning Dialog Box***


**7.** Do one of the following:

- Remedy all conditions that caused the examination to fail, using the instructions in Section 3.6, "Resolving Errors" on page 3-21. Then, select the **Retry** option and click **OK**.
- Select the **Continue with an incomplete upgrade** option and click **OK**.
- Select the **Specify a different source Oracle home** option, click **OK**, then return to Step 3.
- Select the **Cancel the upgrade process** option and click **OK**. The OracleAS Upgrade Assistant stops.

  If the examination was successful, or you chose to continue with an incomplete upgrade, the Summary screen appears as shown in Figure 3–6.

**8.** The OracleAS Upgrade Assistant Summary screen appears. You can scroll to view the components, clicking the plus symbol (+) to expand a component's upgrade items. Review the components, then click **Finish**.

***Figure 3–6 OracleAS Upgrade Assistant Summary Screen***


---

**Note:** The Summary screen is the last screen before upgrade processing begins. Before you click Finish, verify that the choices on previous screens are correct and the upgrade items listed are ready to upgrade. See Section 3.1, "Preparing to Upgrade: Common Requirements for All Components" on page 3-2 for more information.

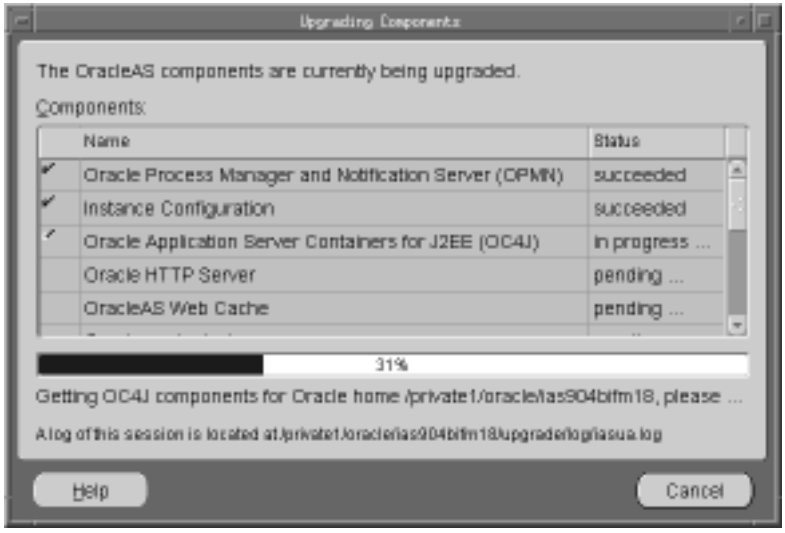
---

**9.** The Upgrading screen appears as shown in Figure 3–7. The Status column for each component contains one of the following:

***Table 3–3 OracleAS Upgrade Assistant Upgrading Status***

| Status | Meaning |
|---|---|
| in progress... | The OracleAS Upgrade Assistant is upgrading the component's upgrade items. |
| pending... | The component will be upgraded when the OracleAS Upgrade Assistant finishes upgrading the current component. |
| succeeded | The component was upgraded successfully. |
| failed | The OracleAS Upgrade Assistant could not upgrade the component. |

***Figure 3–7 OracleAS Upgrade Assistant Upgrading Screen***



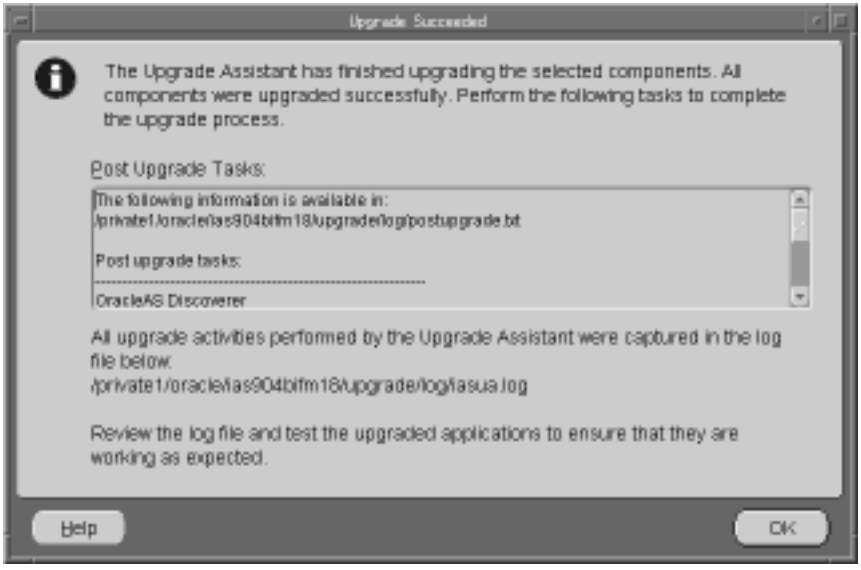
**10.** After the upgrade completes, the Upgrade Failed or Upgrade Succeeded screen (Figure 3–8) appears. Do one of the following.

- Click **OK** to close the Upgrade Failure screen and remedy the conditions that prevented the components from being upgraded. Start the OracleAS Upgrade Assistant again.

  **See Also:** Section 3.6, "Resolving Errors" on page 3-21 and Section 3.7, "Restarting the OracleAS Upgrade Assistant" on page 3-29.

- Click **OK** to close the Upgrade Succeeded screen.

***Figure 3–8 OracleAS Upgrade Assistant Upgrade Succeeded Screen***




The Upgrade Succeeded screen specifies the location of the upgrade log file and lists the post-upgrade tasks to be performed for various components.

### 3.5.5 Performing an Upgrade with the OracleAS Upgrade Assistant (Command-line Version)

This section explains how to start and use the OracleAS Upgrade Assistant command-line version to perform an upgrade.

---

**Note:** The OracleAS Upgrade Assistant examines components differently in the command-line version and the GUI version.

If the examination of a component fails in the command line version, then the upgrade is not performed.

If the examination of a component fails in the GUI version, the following choices are provided: retry, continue with an incomplete upgrade, specify another Oracle home, or cancel the upgrade.

---

1. Start the OracleAS Upgrade Assistant with the command:

   **<*destination_MT_OH*>/upgrade/iasua.sh -sourcehome <*source_MT_OH*>**

---

**Note:** The argument `-sourcehome` is required to start the command-line version of the OracleAS Upgrade Assistant (`iasua.sh` without this argument starts the GUI version). You can also use the following optional arguments when starting the command-line version:

`-verbose` to output detailed information to the screen during upgrade

`-noprompt` to turn off prompting and user confirmation during upgrade (by default, prompting and user confirmation are on)

---

---

**Note:** Use the `-invptrloc` argument, described in Section 3.5.3, "Starting the OracleAS Upgrade Assistant To Use Multiple Oracle Universal Installer Inventory Locations" on page 3-10 if there are multiple inventories involved in the upgrade.

---

A prompt appears listing all the pre-upgrade requirements and asking you to verify that they have been met.

```
Validating Oracle homes

Validating component plug-ins

Initializing component plug-ins

Pre-upgrade requirements:

The destination Oracle home has not been modified since
it was installed

The source and destination OracleAS instances are not
running

The source and destination Oracle Enterprise Manager
processes are not running

Verify that each of the pre-upgrade requirements above
have been met.

Have the pre-upgrade requirements been met?[No]Yes
```

2. Ensure that all the listed requirements are met. Then answer the prompt `[Y]` Yes to continue.

   Messages similar to the following appear (The messages vary according to components found in the Oracle home):

```
Examining component "Oracle Process Manager and
Notification Server (OPMN)"

Examining component "Instance Configuration"

Examining component "Oracle Application Server
Containers for J2EE (OC4J)"

Examining component "Oracle HTTP Server"

Examining component "OracleAS Web Cache"

Examining component "Oracle mod_plsql"

Examining component "Oracle Enterprise Manager"

Upgrading component "Oracle Process Manager and
Notification Server (OPMN)"

Upgrading component "Instance Configuration"
```

```
Upgrading component "Oracle Application Server
Containers for J2EE (OC4J)"

Upgrading component "Oracle HTTP Server"

Upgrading component "OracleAS Web Cache"

Upgrading component "Oracle mod_plsql"

Upgrading component "Oracle Enterprise Manager"

The command completed successfully
```

3. If any error messages are displayed in Step 2, correct the errors as explained in Section 3.6, "Resolving Errors" on page 3-21. Then restart the Upgrade Assistant and perform the upgrade process again.

## 3.6 Resolving Errors

If errors occur at either stage of the upgrade process, you must correct the conditions that caused them before you try the upgrade again.

### 3.6.1 Resolving Common Errors

Under certain conditions, the OracleAS Upgrade Assistant cannot perform an upgrade. Among these are that the starting configuration is unsupported, processes are running in the Oracle homes, the Infrastructure services are unavailable, or there is insuffcient memory for a large-scale OC4J application upgrade.

This section identifies each condition and its cause(s), and explains how to resolve it.

#### 3.6.1.1 Source Oracle Home Not Provided by OracleAS Upgrade Assistant

If the source Oracle home does not appear as expected in the drop-down list of the Oracle Homes screen when you execute the OracleAS Upgrade Assistant, suspect one of these conditions: wrong installation type, Oracle homes are on different computers, or the Oracle home is not identified in the default inventory. The solution for each of these is detailed below.

#### Wrong Installation Type

The source Oracle home will not appear if the source middle tier instance is not of the same installation type as the destination middle tier instance. If this is the case, you must do one of the following:

- Expand the installation type, as described in "Expanding the Middle Tier Installation Type" on page 1-10.
- Reinstall the destination middle tier with the same installation type as the source middle tier.

#### Oracle Homes on Different Computers

Another case in which the source middle tier will not appear as a selection is that the source middle tier instance is installed on a different computer from the destination middle tier instance. If this is the case, you must install the destination middle tier instance on the same computer as the source instance to be upgraded.

#### Oracle Home Not in Default Inventory

The OracleAS Upgrade Assistant uses the default inventory location to populate the drop-down list in the Oracle Homes screen. If the source Oracle home is not listed in the default Oracle Universal Installer inventory, then you need to provide the inventory file location to the OracleAS Upgrade Assistant. Start the OracleAS Upgrade Assistant with the `-invptrloc` option, described in Section 3.5.3, "Starting the OracleAS Upgrade Assistant To Use Multiple Oracle Universal Installer Inventory Locations" on page 3-10 to specify the inventory location.

### 3.6.1.2 Upgrade Fails During OPMN, OC4J, or Oracle HTTP Server Upgrade

If the upgrade fails during the OPMN, OC4J or Oracle HTTP Server upgrade, it is probably because OPMN is still running in one or both instances (source and destination). You must stop OPMN before starting the OracleAS Upgrade Assistant. Follow the instructions in Section 3.1.2, "Stopping OracleAS Instances" on page 3-4.

### 3.6.1.3 Upgrade Fails During OC4J Examination or Other Phase

If the upgrade fails during the OC4J examination phase, or another phase, it is probably because the Infrastructure is unavailable. The OracleAS Upgrade Assistant needs the Infrastructure services for certain operations, so the Infrastructure must be started before you start the OracleAS Upgrade Assistant. Follow the instructions in Section 3.5.2, "Starting the Infrastructure" on page 3-10.

#### 3.6.1.4 Upgrade Fails During Extensive OC4J Upgrade

If the upgrade fails while attempting to upgrade many OC4J applications, or large OC4J applications, suspect a memory shortage. You can configure a memory increase for the upgrade operation. Follow the instructions in Section 3.2.2, "Optional: Increasing JVM Memory for Large OC4J Upgrades" on page 3-5.

### 3.6.2 Examining the Log File

You can examine the `<destination_MT_OH>/upgrade/log/iasua.log` file and Table 3–4, "OracleAS Upgrade Assistant Error Messages" to determine the cause of examination and upgrade failures.

---

**Note:** By default, the OracleAS Upgrade Assistant logging function appends, so you should always look for the last instance of a message in the file. You can set `log.append=FALSE` in `<destination_MT_OH>/upgrade/iasua.properties` to overwrite entries instead of appending them.

---

#### 3.6.2.1 Recovering From Examination Failures

To determine the cause of an examination failure:

1. Note the name of the failed component in the OracleAS Upgrade Assistant dialog or command-line output.
2. Open `<destination_MT_OH>/upgrade/log/iasua.log`.
3. Search for the message `Starting to examine component_name`.
4. Investigate the messages between the `Starting...` message and the message `Finished examining component_name with status: Failure.`

#### 3.6.2.2 Recovering From Upgrade Failures

To determine the cause of an upgrade failure:

1. Note the name of the failed component in the OracleAS Upgrade Assistant dialog or command-line output.
2. Open `<destination_MT_OH>/upgrade/log/iasua.log`.
3. Search for the message `Starting to upgrade component_name`.
4. Investigate the messages between the `Starting...` message and the message `Finished upgrading component_name with status: Failure`.

***Table 3–4 OracleAS Upgrade Assistant Error Messages***

| Component | Message | Possible Cause and Solution |
|---|---|---|
| All | `Unable to upgrade file` *`filename`*. | The file was not found in the source Oracle home, or you do not have sufficient permissions to copy the file. Determine the permissions for the file in the source Oracle home and the destination Oracle home, and adjust them as necessary. |
| Instance Configuration | `INVALID_XML_CONFIG_FILE` | The `iasschema.xml` file is corrupted. Provide an uncorrupted version of the file. |
| Instance Configuration | `IOException` | The `iasschema.xml` file could not be accessed in the source or destination Oracle home. Ensure that the file is accessible in both locations. |
| Oracle Application Server Containers for J2EE | `J2eeDeploymentException` | An application EAR file is not 100% J2EE compliant.<br>Use the `validateEarFile` utility to identify the noncompliant characteristics, and correct them. Instructions for using the utility are provided in Section 3.6.3.2.1, "Validating EAR Files for J2EE Compliance" on page 3-28. |
| OracleAS Forms Services | `Save files operation failed.` | The copy operation failed. Some files are copied "as is" from <*`source_MT_OH`*> (i.e., registry.dat and ftrace.cfg). Verify that all of these files exist and that permissions and disk space are sufficient for a copy operation. |
| OracleAS Forms Services | `Invalid section in the <formsweb.cfg> <default.env> file.` | There is an invalid entry in the named file in <*`source_MT_OH`*>. Examine the file, and locate and correct any errors. |
| OracleAS Forms Services | `Invalid or missing configuration file.` | There is an invalid configuration file in <*`source_MT_OH`*>. Examine the file, and locate and correct any errors. |

***Table 3–4 OracleAS Upgrade Assistant Error Messages***

| Component | Message | Possible Cause and Solution |
|---|---|---|
| OracleAS Forms Services | `Invalid or missing Forms configuration file <file name>.` | The Upgrade Assistant is unable to locate the configuration files specified in the `formsweb.cfg` file (`*htm` and `*env` files), or the user-defined FormsServlet configuration file specified in `oc4j_bi_forms.properties`. Ensure that all files specified in the entries are valid and exist in the specified location. |
| OracleAS Forms Services | `Forms is not configured in the Source Oracle Home <version number>, Forms upgrade cannot proceed.` | If Forms services are not configured in the source OracleAS middle tier installation, then the Upgrade Assistant will not upgrade OracleAS Forms Services. Ignore this message; if OracleAS Forms Services is not configured in the source Oracle home, then upgrade is unnecessary. |
| OracleAS Forms Services | `Forms is not configured in the Destination Oracle Home <version number>, Forms upgrade cannot proceed.` | If Forms services are not configured in the destination middle tier installation, then the Upgrade Assistant will not upgrade OracleAS Forms Services. Configure OracleAS Forms Services in the destination Oracle home. |
| Oracle HTTP Server (mod_plsql) | `java.io.FileNotFoundException...Apache/modplsql/conf/dads.conf`<br>or<br>`java.io.FileNotFoundException...Apache/modplsql/conf/cache.conf` | The file was not found. Provide a file at the location specified. |

***Table 3–4 OracleAS Upgrade Assistant Error Messages***

| Component | Message | Possible Cause and Solution |
|---|---|---|
| Oracle Application Server Web Services UDDI Registry | `iAS/Upgrade/UddiPlugin Destination configuration file is not found at:`<br>*`path`*<br>`Look for backup at`<br>*`path`*<br>`Unable to upgrade file`<br>or<br>`iAS/Upgrade/UddiPlugin Destination configuration file is not found at:`<br>*`path`*<br>`and its backup is not found at`<br>*`path`*<br>`Upgrading cannot proceed` | There is no `uddiserver.config` file or `uddiserver.config.backup` file in the 10*g* (9.0.4) Oracle home.<br>Place a copy of the shipped 10*g* (9.0.4) `uddiserver.config` in `<`*`destination_MT_OH`*`>/ds/uddi/config/`. |
| Oracle Application Server Web Services UDDI Registry | `iAS/Upgrade/UddiPlugin Missing URL prefix definition for UDDI.` | The `uddiserver.config` file does not contain a definition for the property `oracle.uddi.server.db.urlPrefix`<br>Define the `uddiserver.db.urlPrefix` value in the `<`*`destination_MT_OH`*`>/ds/uddi/config/uddiserver.config` file. |
| Oracle Application Server Wireless | `Could not copy file.`<br>or<br>`Could not create directory.` | The disk is full, or you do not have sufficient permissions to copy the file. Determine the available disk space and permissions for the directory `<`*`destination_MT_OH`*`>/wireless/server/classes` and adjust these as necessary. |

## 3.6.3 Reasons for Oracle Application Server Containers for J2EE Upgrade and Deployment Failures

This section discusses reasons for which an Oracle Application Server Containers for J2EE upgrade may fail.

### 3.6.3.1 Configuration Change Requirements

If a configuration does not perform as expected after an upgrade, it might be because configuration changes were made to OC4J application files by means other than the Oracle Enterprise Manager Application Server Control. Only the changes made by the Oracle Enterprise Manager Application Server Control will be included in the OC4J upgrade performed by the OracleAS Upgrade Assistant. Manually edited files may not be in the scope of the managed configuration, and the edits may not be preserved in an upgrade.

If you use Distributed Configuration Management's `dcmctl` utility to perform configuration changes, see the *Distributed Configuration Management Reference Guide* for instructions and a complete discussion on the correct usage of the commands.

### 3.6.3.2 Application Deployment and J2EE Compliance Requirements

OC4J deployment enforces J2EE compliance rules, so the OracleAS Upgrade Assistant may not upgrade applications that are not fully J2EE compliant. The OracleAS Upgrade Assistant simply reads the files in the source Oracle home and attempts to deploy them to the destination Oracle home; if deployment fails, it could be because an application is not J2EE compliant. If the OracleAS Upgrade Assistant cannot deploy an application for any reason, it logs the exception in the *<destination_MT_OH>*`/upgrade/log/iasua.log`. The exception may not be explicitly described as a J2EE compliance issue, but that may be the reason for the failure. Knowledge of the J2EE and EJB specifications, and the EJB features used in applications will be helpful in preventing and troubleshooting deployment failures (10*g* (9.0.4) supports a higher version of the EJB specification than Release 2 (9.0.2)).

While the development of J2EE applications is standardized and portable, the XML configuration files are not. Multiple XML files may need to be configured for an OC4J application to be deployed, and the required configuration varies according to the services the application uses. For example, if the application uses a database, the DataSource object in the `data-sources.xml` file must be configured.

**3.6.3.2.1 Validating EAR Files for J2EE Compliance** The `dcmctl` utility provides a J2EE compliance validation command. It takes one input, the name of an EAR file, and then lists non-compliant characteristics of that file. The syntax is:

`<destination_MT_OH>/dcm/bin/dcmctl validateEarFile -f <full path and filename for ear file>`

You must provide the full path to the EAR file.

If you connect to the Internet using a proxy server, you must configure proxy settings so that the validation routine can access DTDs (for example, on the Sun Microsystems site). To do this, you define an environment variable called `ORACLE_DCM_JVM_ARGS`, which specifies a hostname and port for the proxy. For example, using `tcsh`, the command is:

```
setenv ORACLE_DCM_JVM_ARGS
"-DhttpProxy.host=www-proxy.hostname.com
-DhttpProxy.port=9999"
```

where hostname is the host name and 9999 is the port number. The method of defining this environment variable depends on the platform, so refer to system documentation for instructions on defining this variable.

If there is no firewall to connect to an external network, use the `-noproxy` flag with the command. For example:

`<destination_MT_OH>/dcm/bin/dcmctl validateEarFile -f <full path and filename for ear file> -noproxy`

***Example 3–2 validateEarFile Command and Output for J2EE-Compliant Application***

```
dcmctl validateEarFile -v -f simple.ear
No J2EE XML/DTD validation errors were found
```

***Example 3–3 validateEarFile Command and Output for non-J2EE-Compliant Application***

```
dcmctl validateEarFile -v -f petstore.ear
Warning: J2EE/DTD validation errors were found
ADMN-906001   {0}   Base Exception:
oracle.ias.sysmgmt.deployment.j2ee.exception.J2eeDeploymentException:Cannot get
xml   document by parsing /var/tmp/jar50152.tmp: Invalid element 'servlet' in
content of 'web-app',   expected elements '[servlet-mapping, session-config,
mime-mapping, welcome-file-list,   error-page, taglib, resource-ref,
security-constraint, login-config, security-role, env-entry,   ejb-ref]'.
```

It is a good idea to review all applications for overall J2EE compliance before upgrading them, since there are cases in which an application is deployable, but delivers unpredictable or undesirable server behavior. For example, ensure that each application has a unique context root defined in `application.xml`.

## 3.7 Restarting the OracleAS Upgrade Assistant

You can restart the OracleAS Upgrade Assistant after it has partially or completely processed an Oracle home. Follow these steps:

1. Start the **OracleAS Upgrade Assistant** GUI version as described in Section 3.5.4 on page 3-11, or the command-line version as described in Section 3.5.5 on page 3-19.

   The OracleAS Upgrade Assistant displays one of the following messages, depending on the outcome of the previous upgrade:

   If the previous upgrade was unsuccessful, then the message is:

   ```
   The OracleAS Upgrade Assistant has already processed this
   destination Oracle home directory, it didn't complete
   successfully.
   ```

   If the previous upgrade was successful, then the message is:

   ```
   The OracleAS Upgrade Assistant has already successfully
   processed this destination Oracle home directory.
   ```

2. Close the dialog and continue with the upgrade.

## 3.8 Completing the Upgrade

This section explains how to perform the manual tasks that may be necessary to make the newly upgraded 10*g* (9.0.4) instance functional after the OracleAS Upgrade Assistant has finished executing.

This section contains the following topics:

## 3.8.1 Completing the Oracle HTTP Server Upgrade

This section describes post-upgrade tasks for the Oracle HTTP Server.

### 3.8.1.1 Manual Upgrade Tasks You May Need to Perform

The OracleAS Upgrade Assistant upgrades the standard settings for the Oracle HTTP Server. If you have configuration files or documents that are in non-standard locations or referenced in non-standard ways, you must upgrade these manually. These, and other specific cases for manual upgrade, are detailed below.

- **If you want the Oracle HTTP Server to listen on a port numbered lower than 1024:** The HTTP server executable `apachectl` must have root user privileges to bind to ports numbered lower than 1024. Follow these steps to grant root privileges to the executable:

  1. Log in to the root account.
  2. Navigate to `<destination_MT_OH>/Apache/Apache/bin` and issue these commands:

     ```
     chown root .apachectl

     chmod 6750 .apachectl
     ```

  3. Exit the root account.

- **If mod_osso was configured:** If mod_osso was configured, after upgrade, the osso.conf file continues to use the Release 2 (9.0.2) partner entry in the Single Sign-On server. The 10*g* (9.0.4) partner entry in the Single Sign-On server is not being used, and will cause a broken link (invalid URL) when the application logs out. You should remove the 10*g* (9.0.4) partner entry. In addition, if the name of the entry in use is obsolete (in that it refers in some way to the source Oracle home), you may wish to rename it.

- **If there are configuration files in non-default locations:** If `httpd.conf`, `mod_oc4j.conf`, `mod_osso.conf` and `moddav.conf` files are not in the default location, you must upgrade them manually by applying the customizations in the files in the source Oracle home to the files in `the destination Oracle home`.

- **If there are custom files and directories referenced by Oracle HTTP Server configuration files:** Because the OracleAS Upgrade Assistant only upgrades the items listed in Section A.1.4.1, "OHS Upgrade Items" on page A-7, there could be files or directories referred to by directives such as `Alias`, `mod_rewrite`, and log directives, such as `ErrorLog`, that are not present after the upgrade.

Ensure that all such items are upgraded manually and exist in the locations expected by the directives. If these files or directives are missing after the upgrade, the Oracle HTTP Server may not start. You can identify errors by starting the Oracle HTTP Server individually after the upgrade, and examining the <*destination_MT_OH*>/Apache/Apache/logs/error_log for errors associated with these items.

- **If there are Dynamic Monitoring Service (DMS) configuration elements in the httpd.conf and mod_oc4j.conf files**: You must relocate these configuration elements into the dms.conf file.
- **If Oracle Application Server Web Cache is the first listener:** If OracleAS Web Cache is configured as the first listener, ensure that the Oracle HTTP Server directives listed in Table 3–5 have the same values as the corresponding OracleAS Web Cache elements. In particular, note that the Oracle HTTP Server Port directive specifies the port number of a front-end load balancer or reverse proxy. Thus, if Oracle Application Server Web Cache is used, then the Oracle HTTP Server Port directive should have the value of the port on which OracleAS Web Cache is listening.

***Table 3–5 Oracle HTTP Server and Oracle Application Server Web Cache Port Settings***

| Oracle HTTP Server Directive | Oracle Application Server Web Cache Element |
|---|---|
| VirtualHost | Site definitions |
| Listen | Origin server ports |
| VirtualHost, Listen | Site-to-server mappings |
| Port | Listen |

- **If you have static documents in the default DocumentRoot directory that you want to upgrade:** The OracleAS Upgrade Assistant locates static document files and directories for upgrade in the location specified in the DocumentRoot directive. The DocumentRoot directive defines the location for static documents and related directories. The base server has a document root location, and each virtual host has one. The OracleAS Upgrade Assistant copies files under these directories to the destination Oracle home. The default DocumentRoot directory <*source_MT_OH*>/Apache/Apache/htdocs contains demonstration programs and release notes placed there by the installer, so the OracleAS Upgrade Assistant does not upgrade this directory. You must upgrade this directory manually.

## 3.8.2 Completing the Oracle Application Server Containers for J2EE (OC4J) Upgrade

The OracleAS Upgrade Assistant performs many of the Oracle Application Server Containers for J2EE (OC4J) upgrade tasks. However, some components of OC4J may require manual adjustments, or may have characteristics of which you should be aware before using OracleAS 10*g* (9.0.4).

This section details upgrade considerations for some sub-components of OC4J. If you use Oracle JMS, Oracle JDBC, the XML Parser for JAXP / XDK, or Oracle JSP pages, some or all of the topics in this section may be useful to you.

### 3.8.2.1 Upgrading Oracle Application Server Java Authentication and Authorization Service (JAZN) LDAP Security Settings

The OracleAS Upgrade Assistant does not upgrade the JAZN settings (the `orion-application.xml` file). Therefore, if you upgraded OC4J applications that use the JAZN LDAP User Manager for security, to complete the upgrade, you must perform the following steps:

1. Using the Oracle Enterprise Manager Application Server Control, in the **General Properties** section of the OC4J application, under **User Manager**, select the **JAZN LDAP User Manager**.
2. In the **Security Settings** section of the OC4J application, under **Security Roles**, map **Users/Groups** to the same role defined in the source Oracle home.

### 3.8.2.2 Upgrading JAZN Library Path Entries

In Oracle Application Server 10*g* (9.0.4), the `jazn.jar` file has been split into two JAR files: `jazn.jar` and `jazncore.jar`. For this reason, after upgrading OC4J applications that use JAZN, both JAR file names must have library path entries in the `application.xml` file.

Ensure that the `application.xml` file contains both of the entries below:

```
<library path="904 J2EE HOME/jazn.jar"/>

<library path="904 J2EE HOME/jazncore.jar"/>
```

where

```
<904 J2EE HOME> = <destination_MT_OH>/j2ee/home
```

#### 3.8.2.3 Upgrading OC4J Instances Created by the Installer

Customizations that were made to OC4J instances in the opmn.xml file must be upgraded manually. This includes the instances created by the installer (home, OC4J_WIRELESS, OC4J_DEMOS, OC4J_PORTAL OC4J_BI_FORMS). The OracleAS Upgrade Assistant upgrades customizations to OC4J instances that were created by the user.

#### 3.8.2.4 Upgrading application.xml Entries

If you have customized entries in the application.xml file, such as library paths, Java options, and OC4J options, you must upgrade them manually.

#### 3.8.2.5 Upgrading the jms.xml File

The jms.xml file is not automatically upgraded from earlier versions. All queues, topics, and connection factories defined in the jms.xml file in the source Oracle home must be added to the jms.xml file in the destination Oracle home.

#### 3.8.2.6 Using the Compatibility Test Suite (CTS) Compatibility Flag for Backward Compatibility

In Oracle Application Server 10*g* (9.0.4), OC4J by default complies with the J2EE 1.3 specification. In some cases, this results in behavior that differs from that seen with previous OC4J implementations. To allow for backward compatibility, OC4J supports a CTS compliance flag that you can set to false to revert to previous OC4J behavior in the following components:

- Oracle JMS
- Oracle JDBC
- Oracle XML parser for JAXP/XDK

The compliance behavior of OC4J is determined by the flag `oracle.cts.useCtsFlags`, with a default value of `true`. If any of the upgrade issues are critical in a particular application, you can disable CTS compliance and revert to old behavior for an OC4J instance by setting the flag value to false in an OC4J properties file, and providing the location of the properties file to OC4J.

For example, the file `<destination_MT_OH>/j2ee/home/config/oc4j.properties` might contain the flag:

```
oracle.cts.useCtsFlags=false
```

Supply the name and location of a properties file to OC4J through an <oc4j-option> element in the <*destination_MT_OH*>/opmn/conf/opmn.xml file, as in the following example:

***Example 3–4 oc4j-option Element in opmn.xml File***

```
<oc4j>
...
 <oc4j-option value="-p <destination_MT_OH>/j2ee/home/config/oc4j.properties"/>
...
</oc4j>
```

This is equivalent to starting OC4J as follows in standalone mode (where % is a system prompt):

***Example 3–5 Starting OC4J in Standalone Mode***

```
% java -jar oc4j.jar -p <destination_MT_OH>/j2ee/home/config/oc4j.properties
```

**3.8.2.6.1 CTS Compatibility and OJMS** In the OracleAS 10*g* (9.0.4) implementation of Oracle JMS (OJMS), which complies with J2EE 1.3, some behavior differs from OJMS behavior in Oracle9iAS Release 1 (1.0.2.2). (There are no such upgrade considerations between Oracle9iAS releases 9.0.2 and 9.0.3.) The differences are as follows:

- JMSExpiration—In the OJMS 10*g* (9.0.4) J2EE 1.3-compliant implementation, the JMSExpiration header value in a dequeued message is the sum of the JMS timestamp when the message was enqueued, and the time-to-live. This value is expressed in milliseconds from midnight, January 1, 1970 to the current Greenwich Mean Time. If a message never expires, the value is 0.

  In the OJMS 1.0.2.2 implementation, the JMSExpiration header value in a dequeued message is the duration until expiration of the message, in milliseconds. If a message never expires, the value is -1.

- JMSPriority—In the OJMS Release 2 (9.0.4) 1.3-compliant implementation, 9 is the highest priority, 0 is the lowest priority, and 4 is the default priority.

  In the OJMS 1.0.2.2 implementation, `java.lang.Integer.MIN_VALUE` is the highest priority, `Integer.MAX_VALUE` is the lowest priority, and 1 is the default priority.

- Durable subscribers—In the OJMS 10*g* (9.0.4) J2EE 1.3-compliant implementation, durable Topic Subscribers with the same name are not allowed under any circumstances.

  In the OJMS 1.0.2.2 implementation, durable Topic Subscribers with the same name are allowed if they are subscribed to different topics.

- Strongly typed JMS selectors—In accordance with the JMS 1.02b specification and J2EE 1.3 compliance requirements, the OJMS 10*g* (9.0.4) implementation uses only a certain subset of SQL92 syntax for selector expression syntax, with the following mandated restrictions:

  - Selector expressions are strongly typed, meaning operators and operands in arithmetic comparisons must be of the same type. Automatic type conversions for the purpose of comparison, such as converting the string "1" to the integer 1, are prohibited.
  - String and boolean comparisons are restricted to "=", "<", and ">". Two strings are equal only if they contain the exact same sequence of characters.
  - The "!=" operator is prohibited.

  The OJMS 1.0.2.2 implementation is not subject to these restrictions or to the limited subset of SQL92 syntax for selector expression syntax.

**3.8.2.6.2 CTS Compatibility and JDBC** In the OracleAS 10*g* (9.0.4) implementation of Oracle JDBC, which complies with J2EE 1.3, some behavior differs from JDBC behavior in Oracle9iAS Release 2 (9.0.2) and prior. The differences are as follows:

- Java types for NUMBER columns—In 10*g* (9.0.4), the `getObject()` method of a result set (`java.sql.ResultSet` instance) returns a `java.lang.Double` value for a NUMBER column with precision, or a `java.math.BigDecimal` value for a NUMBER column without precision.

  In Release 2 (9.0.2) and prior releases, `getObject()` returns a BigDecimal value for any NUMBER column.

- Metadata for NUMBER columns—In 10*g* (9.0.4), the `getColumnTypeName()` method of a result set metadata object (`java.sql.ResultSetMetaData` instance) returns `"FLOAT"` for a NUMBER column with precision, or `"NUMBER"` for a NUMBER column without precision. The `getColumnType()` method returns `java.sql.Types.FLOAT` for a NUMBER column with precision, or `Types.NUMBER` for a NUMBER column without precision.

  In Release 2 (9.0.2) and prior releases, `getColumnTypeName()` returns `"NUMBER"` for any NUMBER column, and `getColumnType()` returns `Types.NUMBER` for any NUMBER column.

- Java types for DATE and TIMESTAMP columns—In 10*g* (9.0.4), the `getObject()` method of a result set returns a `java.sql.Date` value for a DATE column, and a `java.sql.Timestamp` value for a TIMESTAMP column.

  In Release 2 (9.0.2) and prior releases, `getObject()` returns a `java.sql.Timestamp` value for a DATE column. (TIMESTAMP columns were not supported.)

- Exceptions for inappropriate SQL statements—In 10*g* (9.0.4), if an `executeQuery()` call in a statement object contains anything but a SELECT statement (such as if it instead contains an INSERT or UPDATE statement), the JDBC driver properly throws an exception. Similarly, if an `executeUpdate()` call contains a SELECT statement, the driver properly throws an exception. (An UPDATE, INSERT, or DELETE statement is expected.)

  In Release 2 (9.0.2) and prior releases, these situations did not result in exceptions.

**3.8.2.6.3 CTS Compatibility and the JAXP/XDK XML Parser** In the OracleAS 10*g* (9.0.4) implementation of the XML parser for JAXP / XDK, which complies with J2EE 1.3, some behavior differs from XML parser behavior in Oracle9iAS Release 2 (9.0.2) and prior. The differences are as follows:

- `getNamespaceURI()` null return values—In 10*g* (9.0.4), the `getNamespaceURI()` method returns 'null' if the namespace is not defined for an element or attribute.

  In Release 2 (9.0.2) and prior releases, the `getNamespaceURI()` method returns `'""'` in these circumstances.

- `getLocalName()` null return values—In 10*g* (9.0.4), the `getLocalName()` method returns `'null'` if the element or attribute was created using a DOM level 1 API call to `createElement()` or `createAttribute()`.

  In Release 2 (9.0.2) and prior releases, the `getLocalName()` method returns `'"Transfer interrupted!"'` in these circumstances.

- `getPrefix()` null return values—In 10*g* (9.0.4), the `getPrefix()` method returns `'null'` if the element or attribute was created using a DOM level 1 API call to `createElement()` or `createAttribute()`.

  In Release 2 (9.0.2) and prior releases, the `getPrefix()` method returns `'""'` in these circumstances.

---

**Note:** The `getNamespaceURI()`, `getLocalName()`, and `getPrefix()` methods exist with the above changes in the `XMLElement` and `XMLAttr` classes of the `oracle.xml.parser.v2` package.

---

- SAX exceptions—In 10*g* (9.0.4), registered error handlers throw a `SAXException` or `SAXParseException` in error conditions.

  In Release 2 (9.0.2) and prior releases, error handlers throw an `XMLParseException` in error conditions.

- I/O exceptions—In 10*g* (9.0.4), an `IOException` is thrown as is in I/O error conditions.

  In Release 2 (9.0.2) and prior releases, an `IOException` is wrapped in an `XMLParseException`.

### 3.8.2.7 Upgrade Considerations for Enterprise Java Beans

Beginning with Oracle9*i*AS Release 2 (9.0.3), Oracle Application Server Containers for J2EE complies with the J2EE 1.3 specification and implements the Enterprise Java Beans (EJB) 2.0 specification in entirety. Therefore, if you are upgrading from Release 2 (9.0.2) to 10*g* (9.0.4), applications using EJB features in the areas of container-managed persistence and container-managed relationships will require modification.

> **See Also:** *Oracle Application Server Containers for J2EE Enterprise JavaBeans Developer's Guide*, Appendix C.

### 3.8.2.8 Upgrade Considerations for the OC4J Java Server Pages (JSP) Container

This section describes JSP settings that are affected by the upgrade.

**3.8.2.8.1 Enabling Extra Imports** Beginning with Oracle9iAS Release 2 (9.0.3), the OC4J JSP container by default imports the packages listed below into any JSP page, in accordance with the JSP specification. No `page` directive import settings are required.

```
javax.servlet.*
javax.servlet.http.*
javax.servlet.jsp.*
```

In previous releases, the following packages were also imported by default:

```
java.io.*
java.util.*
java.lang.reflect.*
java.beans.*
```

For backward compatibility, you can use the JSP `extra_imports` configuration parameter as a workaround. Alternatively, you can add desired imports through `page` directives or global includes. See the *Oracle Application Server Containers for J2EE Support for JavaServer Pages Developer's Guide* for information about these topics.

**3.8.2.8.2 Setting Additional JSP Flags for Backward Compatibility** When upgrading to Oracle Application Server 10*g* (9.0.4) and using JSP pages, use appropriate settings for the following important JSP configuration parameters.

- `check_page_scope`
- `forgive_dup_dir_attr`

These are set as initialization parameters for the JSP front-end servlet, either in the global-web-application.xml file or in the application orion-web.xml file. Here is an example:

***Example 3–6 JSP Configuration Parameters for Upgrading to 10g (9.0.4)***

```
<servlet>
   <servlet-name>jsp</servlet-name>
   <servlet-class>oracle.jsp.runtimev2.JspServlet</servlet-class>
   <init-param>
      <param-name>check_page_scope</param-name>
      <param-value>true</param-value>
   </init-param>
   ...
</servlet>
```

See the *Oracle Application Server Containers for J2EE Support for JavaServer Pages Developer's Guide* for more information about JSP configuration parameters.

`check_page_scope` (boolean; default: `false`): This parameter was introduced in Oracle9iAS Release 2 (9.0.3). For OC4J environments, set it to `true` to enable Oracle-specific page-scope checking by the `JspScopeListener` utility.

This parameter is not relevant for non-OC4J environments. For JServ, Oracle-specific page-scope checking is always enabled. For other environments, the Oracle-specific implementation is not used and you must use the checkPageScope custom tag for JspScopeListener page-scope functionality. See the *Oracle Application Server Containers for J2EE JSP Tag Libraries and Utilities Reference* for information about JspScopeListener.

`forgive_dup_dir_attr` (boolean; default: `false`): This parameter was introduced in Oracle9iAS Release 2 (9.0.3). Set it to true to avoid translation errors in a JSP 1.2 environment such as OC4J if you have duplicate settings for the same directive attribute within a single JSP translation unit (a JSP page plus anything it includes through `include` directives).

The JSP 1.2 specification directs that a JSP container must verify that directive attributes, with the exception of the page directive `import` attribute, are not set more than once each within a single JSP translation unit.

The JSP 1.1 specification did not specify such a limitation. OC4J offers the `forgive_dup_dir_attr` parameter for backward compatibility.

### 3.8.2.9 Considering JDK 1.4 Issues: Cannot Invoke Classes Not in Packages

Among the migration considerations in moving to a Sun Microsystems JDK 1.4 environment, which is the environment that is shipped with Oracle Application Server 10*g* (9.0.4), there is one of particular importance to servlet and JSP developers.

As stated by Sun Microsystems, "The compiler now rejects import statements that import a type from the unnamed namespace." (This was to address security concerns and ambiguities with previous JDK versions.) Essentially, this means that you cannot invoke a class (a method of a class) that is not within a package. Any attempt to do so will result in a fatal error at compilation time.

This especially affects JSP developers who invoke JavaBeans from their JSP pages, as such beans are often outside of any package (although the JSP 2.0 specification now requires beans to be within packages, in order to satisfy the new compiler requirements). Where JavaBeans outside of packages are invoked, JSP applications that were built and executed in an OC4J 9.0.3 / JDK 1.3.1 or prior environment will no longer work in an OC4J 9.0.4 / JDK 1.4 environment.

Until you update your application so that all JavaBeans and other invoked classes are within packages, you have the alternative of reverting back to a JDK 1.3.1 environment to avoid this issue.

---

**Notes:**

The `javac -source` compiler option is intended to allow JDK 1.3.1 code to be processed seamlessly by the JDK 1.4 compiler, but this option does not account for the "classes not in packages" issue.

Only the JDK 1.3.1 and JDK 1.4 compilers are supported and certified by OC4J. It is possible to specify an alternative compiler by adding a `<java-compiler>` element to the `server.xml` file, and this might provide a workaround for the "classes not in packages" issue, but no other compilers are certified or supported by Oracle for use with OC4J. (Furthermore, do not update the `server.xml` file directly in an Oracle9*i*AS environment. Use the Oracle Enterprise Manager Application Server Control.)

---

For more information about the "classes not in packages" issue and other JDK 1.4 compatibility issues, refer to the following Web site:

`http://java.sun.com/j2se/1.4/compatibility.html`

In particular, click the link "Incompatibilities Between Java 2 Platform, Standard Edition, v1.4.0 and v1.3".

### 3.8.2.10 Considering Modified Servlet APIs and Behavior

When upgrading to Oracle Application Server 10*g* (9.0.4) and using servlets, consider the following changes in servlet APIs and behavior:

- Changes relating to getRequestURI()
- Changes regarding filtering of servlets that are forward or include targets

**3.8.2.10.1 Changes Relating to getRequestURI()** In previous Oracle9iAS releases, whenever Oracle HTTP Server received a request, it would unencode the URI before passing it to OC4J. Therefore, servlets making computations based on the response of `getRequestURI()` (a method on the request object) were implicitly getting a value that had been unencoded one time. As of the OC4J 9.0.4 implementation, Oracle HTTP Server will send OC4J an unaltered version of the URI, which in turn is used by OC4J as the return value of getRequestURI().

If the `mod_rewrite` module is being used in conjunction with `mod_oc4j` in communications between Oracle HTTP Server and OC4J, the rewritten URI that is sent to `mod_oc4j` is the same as what is sent to OC4J, and the return value of `getRequestURI()` will have had `mod_rewrite` rules applied to it.

The mod_rewrite and mod_oc4j modules are discussed in the *Oracle HTTP Server Administrator's Guide*. Further details about mod_rewrite are available in the Apache Server documentation.

**3.8.2.10.2 Filtering of Servlets That Are Forward or Include Targets** In previous OracleAS releases, if a filtered servlet forwards to or includes another servlet, the target servlet, by default, is also filtered. In Oracle Application Server 10*g* (9.0.4), this is no longer the default behavior. Having the target servlet not filtered by default matches the intention of the servlet specification.

This behavior is configurable: in the OC4J 9.0.4 implementation, it is according to the oracle.j2ee.filter.on.dispatch environment flag (false by default); in future implementations, it will be according to web.xml configuration as set forth in the servlet 2.4 specification.

## 3.8.3 Completing the OracleAS Web Cache Upgrade

This section outlines tasks you may need to perform to complete the OracleAS Web Cache upgrade after the OracleAS Upgrade Assistant has finished processing. These tasks include enabling the use of privileged ports, and upgrading a cluster.

### 3.8.3.1 Enabling the webcached Executable to Run as The Root User

OracleAS Web Cache will not start after upgrade if the port settings of 80 and 443 were upgraded from the Oracle9*i*AS Release 2 (9.0.2) Web Cache to the Oracle Application Server 10*g* (9.0.4) Web Cache.

Because port numbers under 1024 are reserved for privileged processes in UNIX, the `webcached` executable in 10*g* (9.0.4) must run as root in order to start the cache server process and bind to these ports.

A script is provided that enables the `webcached` executable to run as the root user:

`<destination_MT_OH>/webcache/bin/webcache_setuser.sh`

Log in to the system as the root user and run the script.

**See Also:** *Oracle Application Server Web Cache Administrator's Guide*

### 3.8.3.2 Upgrading an OracleAS Web Cache Cluster

If you have a OracleAS Web Cache cluster, you can upgrade one cache cluster member at a time. The caches will continue to function, but because the other cluster members have a different version of the configuration, the caches will not forward requests to cache cluster members operating with a different version.

For example, if you upgrade Cache_A to the current version, but have not yet upgraded Cache_B and Cache_C, Cache_A will not forward requests to the cache cluster members Cache_B and Cache_C.

In this situation, the Operations page in Web Cache Manager indicates that the Operation Needed is Incompatible software version.

---

**Note:** When the cache cluster members are not running the same version of OracleAS Web Cache, you can still invalidate documents and you can propagate the invalidation to other cluster members, but the invalidation request must originate with the cache that is operating with the earlier version, such as 9.0.2 or 9.0.3, of OracleAS Web Cache.

---

After you upgrade the cache cluster members, you must perform the following additional steps to synchronize the configuration for the members of the cluster:

1. If the caches have not been started, for each upgraded cache, start OracleAS Web Cache and OracleAS Web Cache Manager. On the command line, enter:

   ```
   opmnctl startproc ias-component=WebCache
   ```

   This command starts the OracleAS Web Cache cache server process and admin server process.

2. In a browser, enter the URL for the OracleAS Web Cache Manager for one of the upgraded caches, and, when prompted, enter the username and password for the ias_admin or administrator user.

3. In the navigator frame, select **Administration > Operations**.

   The **Operations** page appears.

4. In the Operations page, click **Retrieve**.

   Web Cache retrieves the cache-specific configuration information from the remote cache cluster members. Then, Web Cache Manager indicates that the Operation Needed is Propagate Configuration.

5. To propagate the configuration to all cache cluster members, select **All caches** and an **Interval** of **Immediate**. Then, click **Propagate**.

6. Restart the caches by selecting **All caches** and an Interval. Then, click **Restart**. (Note that you can perform this operation as you upgrade each cache, or you can perform this operation after all of the cache cluster members have been upgraded.)

### 3.8.3.3 Upgrading a Web Cache Cluster from Release 2 (9.0.2.x) to 10*g* (9.0.4)

A Release 2 (9.0.2.x) cache cannot accept invalidation messages from a 10*g* (9.0.4) cache. In a configuration that uses a OracleAS Web Cache cluster with a mixture of Release 2 (9.0.2.x) and 10*g* (9.0.4) cluster members, you must configure the Load Balancer to send invalidation messages only to the Release 2 (9.0.2.x) members.

When upgrading a cache cluster from Release 2 (9.0.2.x) to 10*g* (9.0.4), remove cluster members one at a time from the invalidation pool for the Load Balancer prior and upgrade them. Once all the cluster members are upgraded, add them back to the invalidation pool. As an example, assume a configuration with a Load Balancer in front of a cache cluster that is comprised of four members, webche1-host, webche2-host, webche3-host, and webche4-host, all running Release 2 (9.0.2.x).

To upgrade this cache cluster:

1. In the Load Balancer configuration, remove webche1-host from the pool that is responsible for invalidation.
2. Upgrade webche1-host from Release 2 (9.0.2.x) to 10*g* (9.0.4).
3. In the Load Balancer configuration, remove webche2-host from the pool that is responsible for invalidation.
4. Upgrade webche2-host from Release 2 (9.0.2.x) to 10*g* (9.0.4).
5. In the Load Balancer configuration, remove webche3-host from the pool that is responsible for invalidation.
6. Upgrade webche3-host from Release 2 (9.0.2.x) to 10*g* (9.0.4).
7. Upgrade webche4-host from Release 2 (9.0.2.x) to 10*g* (9.0.4). As this is the last cache member in the Load Balancer configuration, it is not necessary to remove it from the invalidation pool.
8. In the Load Balancer configuration, add webche1-host, webche2-host, and webche3-host back into the pool that is responsible for invalidation.

## 3.8.4 Completing the OracleAS Portal Middle Tier Upgrade

This section explains how to perform the manual procedures required to complete the Portal upgrade after the OracleAS Upgrade Assistant has finished processing. It discusses the following topics:

Section 3.8.4.1, "Enabling OracleAS Portal for the Secure Sockets Layer (SSL) Protocol" on page 3-46

Section 3.8.4.2, "Verifying Database Access Descriptor (DAD) Locations After Upgrade" on page 3-46

Section 3.8.4.3, "Upgrading the Parallel Page Engine" on page 3-47

Section 3.8.4.4, "Upgrading Portal Development Kit Services for Java (JPDK) Web Providers" on page 3-48

Section 3.8.4.5, "Completing the mod_plsql Upgrade" on page 3-54

Section 3.8.4.6, "Enabling Monitoring of a Release 2 (9.0.2) Portal Repository by the 10g (9.0.4) Oracle Enterprise Manager Application Server Control" on page 3-54

### 3.8.4.1 Enabling OracleAS Portal for the Secure Sockets Layer (SSL) Protocol

Enabling OracleAS Portal for SSL is a manual process that involves setting a property in a configuration file, and running a script. Follow these steps to SSL-enable Portal applications:

1. Edit the <*destination_MT_OH*>/portal/conf/iasconfig.xml file.
2. Modify the SSL property as follows:

   ```
   SSLEnabled="true"
   ```

3. Change directories to <*destination_MT_OH*>/portal/conf.
4. Execute the OracleAS Portal configuration by issuing this command:

   **ptlconfig -dad <*DAD name*> -site**

   where <*DAD name*> is the name of the Database Access Descriptor (DAD) that points to the Portal repository in use by the Release 2 (9.0.2) Oracle home. Typically, the name is "portal".

   The script uses the iasconfig.xml file to update the Oracle Application Server Web Cache, Oracle Application Server Single Sign-On, and Oracle Internet Directory settings in OracleAS Portal.

### 3.8.4.2 Verifying Database Access Descriptor (DAD) Locations After Upgrade

The Release 2 (9.0.2) instance can have a DAD with the same location as a DAD in the destination location. If this is the case, the OracleAS Upgrade Assistant renames the DAD in the destination instance, since it is illegal for two DADs in the same installation to have the same location. For example, if there is a DAD in Release 2 (9.0.2) with the location:

```
<Location /pls/portal>
```

and the same name is found in the 10*g* (9.0.4) installation, the location in 10*g* (9.0.4) is changed to:

***Example 3–7 Renamed DAD***

```
<Location /pls/portal_0>
```

After upgrade, ensure that all DADs are connecting to the intended resources.

### 3.8.4.3 Upgrading the Parallel Page Engine

If the Portal Parallel Page Engine in the source Oracle home has customized initialization arguments, you must manually upgrade these after the OracleAS Upgrade Assistant has finished processing.

The Parallel Page Engine initialization arguments are defined in the <*source_MT_OH*>/j2ee/OC4J_Portal/applications/portal/portal/WEB-INF/web.xml file.

1. Locate all customized arguments (arguments added to the file and default arguments whose value has been changed) in the source Oracle home web.xml file.
2. Copy the customized arguments from <*source_MT_OH*>/j2ee/OC4J_Portal/applications/portal/portal/WEB-INF/web.xml to <*destination_MT_OH*>/j2ee/OC4J_Portal/applications/portal/portal/WEB-INF/web.xml.

**3.8.4.3.1 Resetting the cacheDir Parameter** The cacheDir parameter in the web.xml file may need to be reset manually after the upgrade, because its default behavior is different in 10*g* (9.0.4). In 10*g* (9.0.4), the cacheDir parameter value is an absolute path. In Release 2 (9.0.2), the path value was the relative path to the <*source_MT_OH*>/Apache/modplsql/cache directory.

- If cacheDir was set to the default location <*source_MT_OH*>/Apache/modplsql/cache, then you can remove the cacheDir parameter entirely, since in 10*g* (9.0.4), it will default to this location if it is not configured.
- If cacheDir was set to a non-default location, then you must manually set it to the absolute path for the non-default location in the destination Oracle home.
- If cacheDir was set to the default location but expressed as a relative path, then you must either remove the parameter so that the default (<*destination_MT_OH*>/Apache/modplsql/cache) will be in effect, or provide the absolute path instead of the relative path.

### 3.8.4.4 Upgrading Portal Development Kit Services for Java (JPDK) Web Providers

The steps in this section are required if the source Oracle home has any of the following characteristics:

- Extra user JPDK web provider applications are deployed in the OC4J_Portal instance
- Provider Groups, Providers, and URL Portlets have been built using the OracleAS Portal user interface
- Configuration changes, customizations, or extensions have been made to web provider applications in the OC4J_Portal instance.

The upgrade process for Portal Development Kit Services for Java (JPDK) Web Providers consists of these steps.

1. Deploy user JPDK web provider applications in the destination Oracle home.

   **Note:** This step applies only to user applications. Applications such as `jpdk` and `portalTools` will already have been deployed in the destination Oracle home.

2. Identify the customized settings in the source Oracle home.
3. Apply the customizations to the destination Oracle home.
4. Update each OracleAS Portal instance that uses the provider with the new URL of the provider. After starting the instance as described in Section 3.12 on page 3-74, perform the steps in Section 3.12.3, "Updating the OracleAS Portal Provider Information" on page 3-76.
5. Refresh the Event/Parameter Passing Samples Provider. After starting the instance as described in Section 3.12 on page 3-76, perform the steps in Section 3.12.4, "Refresh the Event/Parameter Passing Samples Provider" on page 3-77.

The following sections describe the areas in which customizations are made, providing instructions on how to upgrade each area and register the upgraded web provider with its respective portal.

**3.8.4.4.1 User-created Providers, Portlets, and Provider Groups** Extra Providers, Portlets and Provider Groups may have been added to the "jpdk" or "portalTools/providerBuilder" web applications, manually or through the Oracle AS Portal user interface. Follow the steps below to upgrade these providers and provider groups:

1. Locate all extra deployment properties files in the source Oracle home that are not present in the destination Oracle home, and copy them to the destination Oracle home. Compare the following directories to locate the properties files:

   <*source_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/deployment/

   with

   <*destination_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/deployment/

   and

   <*source_MT_OH*>/j2ee/OC4J_Portal/applications/portalTools/providerBuilder/WEB-INF/deployment/

   with

   <*destination_MT_OH*>/j2ee/OC4J_Portal/applications/portalTools/providerBuilder/WEB-INF/deployment/

2. Open each deployment properties file copied in the above step and locate the line that defines the definition property. The definition property specifies the location of an XML provider definition file, relative to the parent `WEB-INF` directory. Copy this file to the same relative location in the destination Oracle home.

   For example, if:

   <*source_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/deployment/myservice.properties

   specified the following:

   ```
   definition=providers/mydir/provider.xml
   ```

   then you would copy

<*source_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/providers/mydir/provider.xml

to

<*destination_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/providers/mydir/provider.xml

3. Copy the `provider.xml` file for the Portlet Building Tools provider (to which URL portlets may have been added through the Portal UI) from

   <*source_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/PORTLETBLDGTOOLS/provider.xml

   to

   <*destination_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/PORTLETBLDGTOOLS/provider.xml

4. Open the Provider Groups registry in <*source_MT_OH*>/portal/pdkjava/providerGroups/iasProviders.xml.

5. Append the entries for all user-created providerGroups (generally, those whose name does not have an "oracle" prefix) to <*destination_MT_OH*>/portal/pdkjava/providerGroups/iasProviders.xml.

   For example, the entry for a user-created provider group might resemble the following:

```
<providerGroup name="my.providergroup" baseLanguage="en">
  <displayName language="en" translation="User Provider Group"/>
  <provider name="my.providers " baseLanguage="en">
    <displayName language="en" translation="User Provider"/>
    <timeout>60</timeout>
    <timeoutMessage language="en" translation="User Provider Timed Out"/>
    <loginFrequency>Once Per Session</loginFrequency>
    <httpURL>http://my.machine.com:7777/jpdk/providers</httpURL>
    <serviceId>urn:myservice</serviceId>
    <httpAppType>PORTAL</httpAppType>
    <cookieDomain>my.machine.com:7777</cookieDomain>
  </provider>
</providerGroup>
```

**3.8.4.4.2 Deployment Properties** If any deployment properties were added or changed from default values, you must apply them manually to the destination Oracle home. The location of the properties varies among web providers, and is indicated by the service identifier of the web provider. The service identifier identifies a provider within an application. The deployment property files are named according to the following convention:

```
<source_MT_OH>/j2ee/OC4J_Portal/applications/<application
name>/<web application name>/
WEB-INF/deployment/serviceidentifier.properties
```

For example, the deployment properties for the JPDK sample web provider, whose identifier is `sample`, reside in:

```
<source_MT_OH>/j2ee/OC4J_Portal/applications/jpdk/jpdk
WEB-INF/deployment/sample.properties
```

1. Locate all customized properties (properties added to the file and default properties whose value has been changed) in the source Oracle home.
2. Copy the customized properties from the source Oracle home to the destination Oracle home.

**3.8.4.4.3 JNDI Environment Variables** If any JNDI environment variables in the source Oracle home were added or changed from default values, you must apply them manually to the destination Oracle home. The location of the variables varies among web providers. The variables for the sample JPDK web provider are in:

```
<source_MT_OH>/j2ee/OC4J_
Portal/application-deployments/jpdk/jpdk/orion-web.xml
```

1. Locate all customized variables (variables added to the file and default properties whose value has been changed) in the source Oracle home.
2. Copy the customized variables from the source Oracle home to the destination Oracle home.

**3.8.4.4.4 File-based Portlet Personalization Manager** A file-based personalization manager my be used in the source Oracle home to personalize portlets. Upgrades to the personalization data will only function if the source and destination Oracle homes use the same Portal schema.

1. Copy the personalization data from the source Oracle home to the destination Oracle home.

The personalization data resides in a directory structure whose top level is in the same directory as the web provider's definition file, `provider.xml`. Ensure that all subdirectories at this location are copied to the destination Oracle home. The sample JPDK web provider is in:

`<source_MT_OH>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/providers/sample`

---

**Note:** You can change the location of personalization data by declaring a preference store `<rootDirectory>` tag. The location of the personalization data is specified in the `provider.xml` file.

---

**3.8.4.4.5 Database-based Portlet Personalization Manager** A database-based personalization manager may be used in the source Oracle home to personalize portlets. You can upgrade the personalization data only if the same Portal schema will be used after the upgrade. Personalization data cannot be copied from one Portal schema to another.

Database-based personalization managers use a data source to connect to the database schema that contains the personalization data. The data source is identified in the file:

`<source_MT_OH>/j2ee/OC4J_Portal/config/data-sources.xml`

1. Copy the personalization data from the source Oracle home to the destination Oracle home.

   The personalization data resides in:

   `<source_MT_OH>/j2ee/OC4J_Portal/config/data-sources.xml`

**3.8.4.4.6 Upgrading the oracle.http.configfile property and cache.xml** If the `oracle.http.configfile` property is present, you must upgrade it, and then upgrade the `cache.xml` file (the location of which is specified by the `oracle.http.configfile` property). Follow these steps to upgrade the property and the file.

1. Upgrade the `oracle.http.configfile` property.

   The property resides in:

   `<source_MT_OH>/j2ee/properties/oc4j_portal.properties`

   or

<*source_MT_OH*>/opmn/conf/opmn.xml

- If the oc4j_portal.properties file contains the oracle.http.configfile property, then copy the file from the source Oracle home to the destination Oracle home.
- If the opmn.xml file contains the oracle.http.configfile property, then copy the value from the opmn.xml file in the source Oracle home to the opmn.xml file in the destination Oracle home.

---

**Note:** The location of the cache.xml file, which you will upgrade in the next step, is specified by the oracle.http.configfile property.

---

2. Copy the cache.xml file from the source Oracle home to the destination Oracle home.

**3.8.4.4.7 Upgrading Provider UI Security Settings** If any security settings were added to, or changed from, default values, then you must apply them manually to the destination Oracle home. The provider UI security file is located at:

<*source_MT_OH*>/j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/deployment_providerui/provideruiacls.xml

and

<*source_MT_OH*>/j2ee/OC4J_Portal/applications/portalTools/providerBuilder/WEB-INF/deployment_providerui/provideruiacls.xml

Follow these steps to upgrade the provideruiacls.xml file:

1. Locate all changed security settings (settings added to the file, and default security settings whose value has been changed) in the source Oracle home.
2. Copy the changed security settings from the file in the source Oracle home to the file in the destination Oracle home.

### 3.8.4.5 Completing the mod_plsql Upgrade

If the mod_plsql configuration has customized settings (properties whose value has been changed from the default) in `<source_MT_OH>/Apache/modplsql/conf/plsql.conf`, then you must manually upgrade these after the OracleAS Upgrade Assistant has finished processing.

1. Locate all customized property values in `<source_MT_OH>/Apache/modplsql/conf/plsql.conf`.
2. Copy the customized values to `<destination_MT_OH>/Apache/modplsql/conf/plsql.conf`.

---

**Note:** The default values for all configuration parameters in `plsql.conf` are listed in Chapter 8 of the *Oracle HTTP Server Administrator's Guide*.

---

### 3.8.4.6 Enabling Monitoring of a Release 2 (9.0.2) Portal Repository by the 10*g* (9.0.4) Oracle Enterprise Manager Application Server Control

If an Oracle Application Server middle tier uses a Release 2 (9.0.2) Oracle9*i*AS Portal Repository, you must execute a script to enable the monitoring component in the Application Server Control to access the OracleAS Portal version, and the OracleAS Portal Metadata Repository database version and start time. If the monitoring component cannot get this information, missing package errors (`WWC_MONITORING`) will occur (these are recorded in the Oracle HTTP Server logs).

Follow the steps below to enable monitoring:

1. Connect to `SYS` in SQL*Plus.
2. Execute the script:

   ```
   <destination_MT_OH>/portal/admin/plsql/wwc/cfgvr902.sql
   <Oracle9iAS portal schema name>
   ```

   If the script was successful, the following text appears:

   ```
   If there were no errors, run the following grant in
   SQL*Plus when connected to the Portal schema:

   grant execute on WWC_MONITORING to PUBLIC
   ```

3. Connect to the OracleAS Portal schema in SQL*Plus. If needed, use the Oracle Directory Manager to obtain the randomized Oracle9*i*AS Portal schema password. Navigate to:

a. Entry Management

b. cn=OracleContext

c. cn=Products

d. cn=IAS

e. cn=Infrastructure Databases

f. OrclReferenceName=Infrastructure Database (for example: iasdb.server.domain.com)

g. OrclResourceName=PORTAL

h. Click on the above entry.

i. Look for the orclpasswordattribute value on the right panel. This is the schema password.

4. Execute the grant with this command:

```
grant execute on WWC_MONITORING to PUBLIC;
```

### 3.8.4.7 Restoring Portal Tools (OmniPortlet and Web Clipping) Providers

Perform the steps in this section if:

- The Portal Tools application from the Portal Development Kit (PDK) November 2002 (9.0.2.4.0) or PDK July 2003 (9.0.2.6.2) is installed on the Release 2 (9.0.2) middle tier, and
- You now want to migrate the Portal Tools application to the 10*g* (9.0.4) middle tier.

Since the OracleAS Portal Middle Tier Upgrade process deploys a new Portal Tools application, you must restore the old providers settings into the new providers. This section explains how to restore them. These steps are necessary only for those providers which you were using and had configured before the upgrade process. If the PDK version is not specified, you need to perform the step for all versions. The OracleAS Portal Tools application contains:

- Web Clipping Provider
- OmniPortlet Provider

**3.8.4.7.1 Restoring the Web Clipping Provider Settings** Follow these steps to restore the Web Clipping Provider settings:

1. **If Web Clipping was configured to use a proxy**: copy the <proxyInfo> tag information from <*source_MT_OH*>/j2ee/OC4J_Portal/applications/portalTools/webClipping/WEB-INF/providers/webClipping/provider.xml to the destination Oracle home.

   The <proxyInfo> tag is a child of the <provider> tag and appears in the file following the <preferenceStore>...</preferenceStore> tag. The <proxyInfo> tag is shown in the example below.

   ```
   <proxyInfo class="oracle.portal.provider.v2.ProxyInformation">
      <httpProxyHost>www-proxy.mycompany.com</httpProxyHost>
      <httpProxyPort>80</httpProxyPort>
   </proxyInfo>
   ```

2. Copy the <repositoryInfo> tag information from <*source_MT_OH*>/j2ee/OC4J_Portal/applications/portalTools/webClipping/WEB-INF/providers/webClipping/provider.xml to the destination Oracle home.

   The <repositoryInfo> tag is a child of the <provider> tag and appears in the file following the <preferenceStore>...</preferenceStore> tag. The <repositoryInfo> tag is shown in the example below.

   ```
   <repositoryInfo class="oracle.portal.wcs.provider.info.DatabaseInformation">
      <useRAA>false</useRAA>
      <databaseHost>mycompany.dbhost.com</databaseHost>
      <databasePort>1521</databasePort>
      <databaseSid>iasdb</databaseSid>
      <databaseUsername>webclip_user</databaseUsername>
      <databasePassword>AX3tR</databasePassword>
      <useASO>false</useASO>
   </repositoryInfo>
   ```

3. **For PDK 9.0.2.4.0 only:** After starting the instance as described in Section 3.12 on page 3-74, perform the steps in Section 3.12.1, "Accessing the Web Clipping Test Provider Page (PDK 9.0.2.4.0 Only)" on page 3-75.

**3.8.4.7.2 Restoring the OmniPortlet Provider Settings** Follow these steps to restore the OmniPortlet Provider settings:

1. **For PDK 9.0.2.4.0 only:** Copy the `<useHashing>` tag and value within the `<preferenceStore>` tag information from `<source_MT_OH>/j2ee/OC4J_Portal/applications/portalTools/omniPortlet/WEB-INF/providers/omniPortlet/provider.xml` to the destination Oracle home. If there is no `<useHashing>` tag in the source Oracle home, then omit it in the destination Oracle home as well.

   The `<preferenceStore>` tag is a child of the `<provider>` tag and appears in the file following the `<session>...</session>` tag. The `<preferenceStore>` tag is shown in the example below.

   ```
   <preferenceStore
   class="oracle.portal.provider.v2.preference.FilePreferenceStore">
      <name>omniPortletprefStore</name>
   </preferenceStore>
   ```

2. **If OmniPortlet was configured to use a proxy**: Copy the `<proxyInfo>` tag information from `<source_MT_OH>/j2ee/OC4J_Portal/applications/portalTools/omniPortlet/WEB-INF/providers/omniPortlet/provider.xml` to the destination Oracle home.

   The `<proxyInfo>` tag is a child of the `<provider>` tag and appears in the file following the `<preferenceStore>...</preferenceStore>` tag. The `<proxyInfo>` tag is shown in the example below.

   ```
   <proxyInfo class="oracle.portal.provider.v2.ProxyInformation">
      <httpProxyHost>www-proxy.mycompany.com</httpProxyHost>
      <httpProxyPort>80</httpProxyPort>
   </proxyInfo>
   ```

3. Copy all directories under `<source_MT_OH>/j2ee/OC4J_Portal/applications/portalTools/omniPortlet/WEB-INF/providers/omniPortlet/` to the destination Oracle home. This restores the customizations to the portlets you are using in OracleAS Portal from this provider.

4. If you had manually added any portlet definitions to OmniPortlet Provider's `provider.xml` file before upgrading, copy them to the destination Oracle home.

5. Configure the Secured data Repository. Ensure that you have copied the `repositoryInfo` tag information from the source Web Clipping `provider.xml` file to the destination Web Clipping `provider.xml` file.
6. **For PDK 9.0.2.4.0 only:** After starting the instance as described in Section 3.12 on page 3-74, perform the steps in Section 3.12.2, "Accessing the OmniPortlet Test Provider Page (PDK 9.0.2.4.0 Only)" on page 3-75.
7. **For PDK 9.0.2.6.2 only, if the OmniPortlet Secured Data Repository was configured:** Copy the `<vaultId>` tag information from `<`*`source_MT_OH`*`>/j2ee/OC4J_Portal/applications/portalTools/omniPortlet/WEB-INF/providers/omniPortlet/provider.xml` to the destination Oracle home.

   The `<vaultId>` tag is a child of the `<provider>` tag and appears in the file following the `<provider>` tag. The `<vaultId>` tag resembles the following: `<vaultId>2</vaultId>`

## 3.8.5 Completing the Oracle Application Server Discoverer Viewer Upgrade

The OracleAS Upgrade Assistant performs most of the Oracle Application Server Discoverer upgrade. The only exceptions are the style sheets used to customize the appearance of Discoverer Viewer, and entries in the `tnsnames.ora` file. This section explains how to perform these tasks to complete the Oracle Application Server Discoverer upgrade.

### 3.8.5.1 Upgrading the End User Layer

If you are upgrading from Oracle9*i*AS Discoverer version 9.0.2.52 or earlier, you must upgrade the End User Layer before you can use Oracle Application Server Discoverer 9.0.4 in 10*g* (9.0.4). See Section 4.6.1, "Upgrading the Oracle Application Server Discoverer End User Layer Schema" on page 4-64, for instructions.

### 3.8.5.2 Upgrading Style Sheets

If you have changed style sheets in the Release 2 (9.0.2) installation, and you want these changes to be in effect in the 10*g* (9.0.4) installation, then you must re-apply the changes manually to the 10*g* (9.0.4) installation. Style sheet files that may require manual upgrade are listed below.

The following files in <*source_MT_OH*>/j2ee/OC4J_BI_Forms/applications/discoverer/web/common/xsl may require manual upgrade:

- about.xsl
- add_connection_eul.xsl
- add_connection_main.xsl
- add_connection_responsibility.xsl
- blaf.xsl
- change_password.xsl
- delete_connection.xsl
- discoverer.xsl
- edit_connection_eul.xsl
- edit_connection_main.xsl
- edit_connection_responsibility.xsl
- errors.xsl
- functions.xsl
- gui_components.xsl
- list_of_connections.xsl
- render_table.xsl
- scripts.xsl

The following files in <*source_MT_OH*>/j2ee/OC4J_BI_Forms/applications/discoverer/web/viewer_files/xsl may require manual upgrade:

- page_layouts.xsl
- sorting.xsl
- style.xsl

### 3.8.5.3 Upgrading the tnsnames.ora File

The <*source_MT_OH*>/network/admin/tnsnames.ora file defines database instances referenced by Oracle Application Server Discoverer connections created in Oracle9iAS Discoverer Release 2 (9.0.2). The same database instances must appear in the <*destination_MT_OH*>/network/admin/tnsnames.ora file in order for connections upgraded to Oracle Application Server Discoverer 10*g* (9.0.4) to function. This file may require upgrade.

> **See Also:** Section 3.8.9, "Upgrading the tnsnames.ora File" on page 3-68.

## 3.8.6 Completing the Oracle Application Server Reports Services Upgrade

The OracleAS Upgrade Assistant performs most of the Oracle Application Server Reports Services upgrade. However, it does not process the script files <*source_MT_OH*>/bin/rw*.sh and the configuration file <*source_MT_OH*>/reports/conf/jdbcpds.conf. If you have customized any of these files, you must apply the customizations to the corresponding files in the destination Oracle home.

> **Note:** To apply the customizations, you must copy the customized entries from the source Release 2 (9.0.2) instance to the destination 10*g* (9.0.4) instance. It is not safe to replace the 10*g* (9.0.4) file with the Release 2 (9.0.2) file, because the files are different.

In addition, you may want to perform the following optional manual steps:

- To preserve the cache files and the cache directory from Release 2 (9.0.2), copy the reports server cache directory from the source Oracle home to the destination Oracle home.
- To monitor additional reports server instances with the Oracle Process Manager and Notification Server (OPMN), define the reports server instances in <*destination_MT_OH*>/opmn/conf/opmn.xml.

> **See Also:** *Oracle Application Server Reports Services Publishing Reports to the Web*, section entitled "Configuring Reports Server with Oracle Process Manager and Notification Server and Oracle Enterprise Manager"

- To manage additional reports server instances with the Oracle Enterprise Manager Application Server Control, define the reports server instances in <*destination_MT_OH*>/sysman/emd/targets.xml.

  **See Also:** *Oracle Application Server Reports Services Publishing Reports to the Web*, section entitled "Configuring Reports Server with Oracle Process Manager and Notification Server and Oracle Enterprise Manager"

### 3.8.6.1 Upgrading User-Defined OC4J Instances With Oracle Application Server Reports Services Deployments

The OracleAS Upgrade Assistant upgrades the Oracle9iAS Release 2 (9.0.2) Business Intelligence & Forms configuration to the Oracle Application Server 10*g* (9.0.4) Business Intelligence & Forms configuration. It is not aware of OC4J instances outside of these configurations that may contain deployed reports, or of customizations made to those instances in order to enable the deployed reports to run.

Therefore, if you are using OC4J instances other than the standard Business Intelligence and Forms OC4J instance, you must apply any manual deployment steps that you performed on those instances in Oracle9iAS Release 2 (9.0.2) to the equivalent instances in Oracle Application Server 10*g* (9.0.4).

You must also add the Java option below to the java-options tag in <*destination_MT_OH*>/opmn/conf/opmn.xml to these OC4J instances before you can use them for Oracle Application Server Reports Services.

```
-Xbootclasspath^/p:<destination_MT_
OH>/vbroker4/lib/vbjboot.jar
```

***Example 3–8 Default java-options Tag***

```
<data id="java-options" value="-Dhttp.proxyHost=www-proxy.us.oracle.com
-Dhttp.proxyPort=80"/>
```

***Example 3–9 java-options Tag Modified for Oracle Application Server Reports Services***

```
<data id="java-options" value="-Dhttp.proxyHost=www-proxy.us.oracle.com
-Dhttp.proxyPort=80"
-Xbootclasspath^/p:<destination_MT_OH>/vbroker4/lib/vbjboot.jar"/>
```

## 3.8.7 Completing the OracleAS Wireless Upgrade

This section provides information on upgrading the Oracle Application Server Wireless Middle Tier from Release 2 (9.0.2) to 10*g* (9.0.4).

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

### 3.8.7.1 Using the Oracle Application Server Wireless Notification Service Upgrade Script

This section explains how to upgrade notifications created by the OracleAS Wireless Notification Engine in the OracleAS Wireless System Manager. The architecture and functionality of the Notification Engine are not described here.

You can upgrade notifications from Release 2 (9.0.2) to 10*g* (9.0.4) with the `migrateNotifications.sh` script. To execute the script:

1. Navigate to `<destination_MT_OH>/wireless/bin`.
2. Set the ORACLE_HOME environment variable to the 10*g* (9.0.4) Oracle home.
3. Issue one of the commands below:
   - **`migrateNotifications.sh -name`** ***`deprecated master alert name(s)`*** **`-owner`** ***`owner user name`***
   - **`migrateNotifications.sh -oid`** ***`deprecated master alert oid`*** **`-owner`** ***`owner user name`***

   ---

   **Notes:** The `name` parameter enables you to upgrade alerts by name.

   The `oid` parameter enables you to upgrade a specific alert by object ID.

   You can use the `%` wildcard character to specify deprecated master alert names. All 9.0.4.x notification objects (master alert service, master service, link, and so on) will be owned by the user name specified.

   ---

   The script does the following:

   - Creates a new master alert service named ***`old master alert name`***`_New`. (This process involves converting the message template to a valid mobile xml, if necessary.)

- Creates the folder `/master/notifications`, if it does not exist.
- Creates the master service *`old master alert name`*`_MS`.
- Creates a mapping for the new master alert and the new master service based on the message template for the old master alert.
- Creates the folder `/Users Home/`*`username`*`/notifications`, if it does not exist.
- Discovers all associated 9.0.2.x `AlertService` objects and converts them to link objects. (The top-level authorization is flattened to link level authorization during this process.)
- Transforms all subscriptions for alert services converted.

The following command upgrades all 9.0.2.x master alert services whose name starts with StockAlert (e.g., StockAlertNews, StockAlertWarning, etc.). All new objects will be owned by the `orcladmin` user.

***Example 3–10 OracleAS Wireless Notification Service Upgrade script command with wildcard name (UNIX)***

```
migrateNotifications.sh -name StockAlert% -owner orcladmin
```

The following command upgrades the 9.0.2.x master alert service with the name StockAlert, and assigns all new objects to the `systemadmin` user.

***Example 3–11 OracleAS Wireless Notification Service Upgrade script command with specific name (UNIX)***

```
migrateNotifications.sh -name StockAlert -owner systemadmin
```

The following command upgrades the 9.0.2.x master alert service with the object ID 1973, and assigns all new objects to the `systemadmin` user.

### 3.8.7.2 Operating Oracle Application Server Wireless Release 2 (9.0.2) and 10*g* (9.0.4) Middle Tiers Together

You can operate an environment with Oracle9iAS Wireless Release 2 (9.0.2) and OracleAS Wireless 10*g* (9.0.4) middle tiers using the same Infrastructure services. However, this configuration is subject to some restrictions, as described below.

- J2ME download and XHTML/XForms based applications should not be used in a mixed environment. These features are new in OracleAS Wireless 10*g* (9.0.4), and would cause errors when attempting to access them from any of the Oracle9iAS Wireless Release 2 (9.0.2) middle tiers. If you wish to use these features, then it is necessary to upgrade all middle tiers to OracleAS Wireless 10*g* (9.0.4).
- The Notification Engine cannot be used in a mixed environment. Instead, you should use the Alert Engine.
- Service access point (service-level address) should be created through an OracleAS Wireless 10*g* (9.0.4) middle tier, in order for them to be visible to both the OracleAS Wireless 10*g* (9.0.4) middle tiers and Oracle9iAS Wireless Release 2 (9.0.2) middle tiers.
- OracleAS Wireless 10*g* (9.0.4) supports user name case sensitivity. However, this requires that you upgrade the Oracle Internet Directory to Oracle Application Server 10*g* (9.0.4).
- Changing (adding, deleting, or updating) a 10*g* (9.0.4) ASK Access point will not be reflected on the Release 2 (9.0.2) Enterprise Manager until the Release 2 (9.0.2) Enterprise Manager is restarted. The Release 2 (9.0.2) OC4J must also be bounced in order for ASK to pick up the changes.
- A driver account (for example, an e-mail account for an e-mail driver) which is removed from an instance and subsequently added to another instance of different release (for example, from Release 2 (9.0.2) to 10*g* (9.0.4)) may cause messages to be lost. To resolve this, restart the OC4J for the instance from which the account is removed.

The Notification Engine introduced in OracleAS Wireless 10*g* (9.0.4) replaces the Alert Engine, which was part of Oracle9iAS Wireless Release 2 (9.0.2). Although the Alert Engine is still available in OracleAS Wireless 10*g* (9.0.4), Oracle Corporation recommends that after all middle tiers have been upgraded to OracleAS Wireless 10*g* (9.0.4), you switch to the Notification Engine, as the Alert Engine may not be available in future versions of OracleAS Wireless. Upgrade scripts are available to help you with this task. See the *Oracle Application Server Wireless Developer's Guide* for details. The Oracle9iAS Wireless Release 2 (9.0.2) Alert APIs have been deprecated, and you must upgrade your applications to use the OracleAS Wireless 10*g* (9.0.4) APIs instead.

### 3.8.7.3 Configuring Site-Level Drivers in a Mixed Mode Environment

In a mixed mode environment, Oracle9*i*AS Wireless Release 2 (9.0.2) and OracleAS Wireless 10*g* (9.0.4) may have transport drivers configured to receive incoming messages. The two sets of entry points, Oracle9iAS Wireless Release 2 (9.0.2) and OracleAS Wireless 10*g* (9.0.4), should not be exposed to a device at the same time. A user issuing a request to the Release 2 (9.0.2) instance should not subsequently send another request, within an 3 hour period, to the entry point defined in the transport driver of the OracleAS Wireless 10*g* (9.0.4) instance. The same user may not receive any response for requests addressed to the latter entry point, if it is violated.

Since the driver configuration is different in Release 2 (9.0.2) and 10*g* (9.0.4), when a Oracle9*i*AS Wireless Release 2 (9.0.2) instance is upgraded to OracleAS Wireless 10*g* (9.0.4), the transport drivers must be managed such that requests are processed as expected.

In 10*g* (9.0.4), a site level driver can be enabled or disabled. By default, it is enabled. If a driver is disabled, it is not recognized by the routing algorithm, and therefore is not used by the messaging system. However, in Release 2 (9.0.2), all site level drivers are recognized by the routing algorithm.

If a Release 2 (9.0.2) instance has two middle tiers, after one of the middle tiers and the Infrastructure are upgraded to 10*g* (9.0.4), the upgraded middle tier may enable or disable a site level driver. However, middle tiers that are not yet upgraded recognize all drivers as enabled. For this reason, it is prudent to remove, rather than disable, a driver in this type of environment.

InRelease 2 (9.0.2), the transport mechanism can route a message to only one driver, and it does not matter whether there is an instance configured for it. This means that a message will not be delivered if it is indeed routed to a driver that has no instance configured. For this reason, the best practice is to remove all drivers that do not have an instance configured in any Release 2 (9.0.2) environment, including a Release 2 (9.0.2) and 10*g* (9.0.4) mixed environment.

## 3.8.8 Completing the OracleAS Forms Services Upgrade

The OracleAS Upgrade Assistant moves most of the OracleAS Forms Services configuration data from the source to the destination Oracle home. However, there may be manual tasks remaining after the upgrade. This section explains how to perform these tasks.

---

**Note:** After the upgrade, the `default.env` file contains the default OracleAS Forms Services environment variables and any user defined environment variables. The Upgrade Assistant upgrades any user defined environment variables to the destination Oracle Home `default.env` file. Any user modifications to the default variables in the source Oracle Home `default.env` file will be extracted and appended to the default environment variable values that the installer puts in the destination Oracle Home `default.env` file.

---

### 3.8.8.1 Upgrading the tnsnames.ora File

Entries may have been added to, or changed in, the `tnsnames.ora` file between the installation of Oracle9*i*AS Release 2 (9.0.2) and upgrade to 10*g* (9.0.4). If so, you must upgrade this file manually so that any added or changed entries are available in 10*g* (9.0.4).

> **See Also:** Section 3.8.9, "Upgrading the tnsnames.ora File" on page 3-68.

If an error occurs after upgrade, it may be because the `tnsnames.ora` file was overwritten by another component upgrade. A missing or incorrect entry yields the following error:

```
ORA-12154: TNS: Could not resolve service name.
```

#### 3.8.8.2 Upgrading Forms *.fmx Files

If you have deployed these files within the source Oracle home, you must manually copy them to the same location in the destination Oracle home. If the `*.fmx` files are not under the Oracle home on the file system, then no action is needed, as the `FORMS90_PATH` will be upgraded by the OracleAS Upgrade Assistant, and it will be valid after the upgrade.

#### 3.8.8.3 Upgrading User-defined Aliases for OracleAS Forms Services Servlets

If you defined any aliases for the OracleAS Forms Services servlets in:

`<`*`source_MT_OH`*`>/j2ee/OC4J_BI_FORMS/applications/forms90app/forms90web/WEB-INF/web.xml`, then you must manually copy these entries to:

`<`*`destination_MT_OH`*`>/j2ee/OC4J_BI_FORMS/applications/forms90app/forms90web/WEB-INF/web.xml`.

#### 3.8.8.4 Upgrading forms90app.ear Deployed in User-defined OC4J Instances

The `forms90app.ear` file is deployed by default into the OC4J_BI_Forms OC4J instance. Note that the Upgrade Assistant upgrades all user-defined OC4J instances and the applications deployed under these instances to the destination Oracle home.

Thus, if you have deployed the `forms90app.ear` file into one of the user-defined OC4J instances in the source Oracle home, the Upgrade Assistant will upgrade this deployment into the corresponding OC4J instance in the destination Oracle home.

As a result, the source Oracle home Release 2 (9.0.2) `forms90app.ear` file is deployed into the destination Oracle home. This causes the configuration of OracleAS Forms Services 10*g* (9.0.4) to be incorrect, because it requires the 10*g* (9.0.4) EAR file in order to function properly.

Therefore, you must undeploy the `forms90app.ear` file from these upgraded OC4J instances in the destination Oracle home, and then deploy `forms90app.ear` in the destination Oracle home again.

The OracleAS Forms Services 10*g* (9.0.4) `forms90app.ear` file is located in: `<`*`destination_MT_OH`*`>/forms90/j2ee`.

#### 3.8.8.5 Enabling the OracleAS Forms Services Oracle Application Server Single Sign-On and Oracle Internet Directory Configuration

The Oracle Application Server Single Sign-On and Oracle Internet Directory configuration for OracleAS Forms Services is disabled by default. If the configuration was enabled in Release 2 (9.0.2), the OracleAS Upgrade Assistant does not automatically enable it in 10*g* (9.0.4).

You can enable the configuration by setting the `ssoMode` entry value to `true` in the file:

`<destination_MT_OH>/forms90/server/formsweb.cfg`

In Release 2 (9.0.2), a servlet alias could be set up in the `forms90.conf` file in order to run Oracle9*i*AS Forms Services using Single Sign-On. In 10*g* (9.0.4), Single Sign-On can be configured on a per-application basis in the `formsweb.cfg` file. You can configure it for all applications as described above, by setting the ssoMode entry value to true in the default section of the file. To configure it for a specific application, set it to true in that application's configuration section of the file.

### 3.8.9 Upgrading the tnsnames.ora File

The `<source_MT_OH>/network/admin/tnsnames.ora` file contains connection information for various databases. It is shared among Oracle Application Server components.

You should examine this file to determine whether it contains any entries that will be needed to operate a component in the 10*g* (9.0.4) installation, but which do not exist in the `<destination_MT_OH>/network/admin/tnsnames.ora` file. For example, entries might have been added since the Oracle9*i*AS Release 2 (9.0.2) installation to provide access to additional databases, or the entry for the Infrastructure Services repository might have been changed in some way.

If new or changed entries exist, you have two choices for upgrading: append individual entries, or copy the entire file.

#### 3.8.9.1 Append New and Changed Entries to the 10*g* (9.0.4) tnsnames.ora File

Copy all new and changed entries to the `<destination_MT_OH>/network/admin/tnsnames.ora` file.

#### 3.8.9.2 Copy the Oracle9*i*AS Release 2 (9.0.2) tnsnames.ora File to the 10*g* (9.0.4) Installation

If you are certain that doing so will not eliminate entries added or changed by other components, you can copy the entire `tnsnames.ora` file from the source to the destination Oracle home, restoring the `EXTPROC_CONNECTION_DATA.US.ORACLE.COM` entry (which is introduced in 10*g* (9.0.4)).

To upgrade by copying the entire file, follow the steps below:

1. Create a backup of *<source_MT_OH>*`/network/admin/tnsnames.ora` and *<destination_MT_OH>*`/network/admin/tnsnames.ora`.
2. Copy the entire `tnsnames.ora` file from the source to the destination Oracle home.
3. Copy the `EXTPROC_CONNECTION_DATA.US.ORACLE.COM` entry (used for executing external procedures in the database) from the 10*g* (9.0.4) backup file created in step 1 to the Oracle9*i*AS Release 2 (9.0.2) file.

## 3.9 Upgrading OracleAS InterConnect

This section explains how to upgrade Oracle9iAS InterConnect.

Since OracleAS InterConnect is not distributed on the OracleAS CD-ROM, and thus may not yet be installed, this section includes installation as a first step to upgrading. Installation instructions for each adapter are provided in the OracleAS InterConnect documentation in the OracleAS documentation library.

In Oracle9*i*AS Release 2 (9.0.2) topologies, OracleAS InterConnect components reside in the Infrastructure or middle tier Oracle home, or a different Oracle home. In 10*g* (9.0.4), these components do not reside in the Infrastructure or middle tier Oracle home. You must install them in a different Oracle home.

The upgrade process consists of the following tasks:

Section 3.9.1, "Installing and Upgrading Hub Components, Adapters, and Development Kit" on page 3-70

Section 3.9.2, "Upgrading Metadata" on page 3-70

Section 3.9.3, "Upgrading Management" on page 3-71

Section 3.9.4, "Upgrading Oracle Workflow" on page 3-71

### 3.9.1 Installing and Upgrading Hub Components, Adapters, and Development Kit

Follow the installation instructions in OracleAS InterConnect components to install the Hub, Adapters, and Development Kits. The information provided in the installation procedure should reflect the configuration of the Release 2 (9.0.2) components. Ensure that the architecture described in the 10*g* (9.0.4) installation is the same as the Release 2 (9.0.2) architecture unless you intend to change the architecture.

### 3.9.2 Upgrading Metadata

Follow these steps to upgrade metadata:

1. Install the most recent Release 2 (9.0.2) OracleAS InterConnect patch set.
2. Run the `oaiexport` script provided with the Release 2 (9.0.2) installation. Supply values for repository name, file name, system password, and connect string with the following command:

   ***<source_InterConnect_OH>*/oai/9.0.2/repository/oaiexport *<file name>* system/*<system password>* *<connect string>***

   The metadata is exported to a file in the current directory.
3. Run the `oaiimport` script provided with the 10*g* (9.0.4) installation. Supply values for repository name, file name, from user (the user ID of the user whose metadata is being imported), system password, oaihub904 schema password, and connect string with the following command:

   ***<destination_InterConnect_OH>*/oai/9.0.4/repository/oaiimport *<file name>* *<from user>* system/*<system password>* *<oaihub904 schema password>* *<connect string>***

   The file is imported into the 10*g* (9.0.4) Hub database.
4. Create the tables in the hub schema database by executing this command in the hub schema:

```
CREATE TABLE emdcomments (emdid NUMBER(10)
                                CONSTRAINT fk_emdcomments_emdid REFERENCES
                                           emd(id) ON DELETE CASCADE
                                CONSTRAINT nn_emdcomments_emdid NOT NULL,
                          name VARCHAR2(4000)
                               CONSTRAINT nn_emdcomments_name NOT NULL,
                          comments  LONG);
```

### 3.9.3 Upgrading Management

Follow the instructions in the 10*g* (9.0.4) OracleAS InterConnect installation documentation. The OracleAS InterConnect Management architecture has changed significantly since Release 2 (9.0.2). The upgraded Oracle Enterprise Manager Application Server Control cannot be used to manage OracleAS InterConnect components. See the *Oracle Application Server InterConnect Installation Guide* for more information.

### 3.9.4 Upgrading Oracle Workflow

Follow these steps to upgrade Oracle Workflow:

1. Install Oracle Workflow with 10*g* (9.0.4), including all pre- and post- installation steps as described in the *Oracle Application Server InterConnect Installation Guide*.

   The installation updates the Oracle Workflow server version in the database. See the *Oracle Application Server InterConnect Installation Guide* for detailed information.

2. Perform all Workflow setup steps for your 10*g* (9.0.4) installation as described in the "Setting Up Oracle Workflow" chapter of the Oracle Workflow Administrator's Guide. In particular, ensure that you:

   - Set your global Workflow preferences appropriately
   - Set up a directory service for Oracle Workflow
   - Configure the Workflow Notification Mailer using the Workflow Manager component of Application Server Control.

3. Copy all customized files from the source Oracle home to the destination Oracle home. Customized files may include:

   - Workflow process definition files (`.wft` files located in `<destination_MT_OH>/wf/res/lang`)
   - Business Event System definition files (`.wfx` files located in `<destination_MT_OH>/wf/res/lang`)
   - SQL scripts (`.sql` files located in `<destination_MT_OH>/wf/sql`)
   - Custom help files (`.htm` files located in `<destination_MT_OH>/wf/doc/lang/wfcust` or `.hlp` files located in `<destination_MT_OH>/wf/res/lang`)

## 3.10 Port Values and the portlist.ini File After Upgrade

After you upgrade a middle tier to OracleAS 10*g* (9.0.4), the upgraded instance is using the same ports that the Oracle9iAS Release 2 (9.0.2) instance used. For this reason, after upgrade, you cannot start the source and destination middle tier instances at the same time, because of port conflicts.

The `<destination_MT_OH>/install/portlist.ini` file does not reflect the upgraded port settings; it lists the port values assigned by the installer before the upgrade. Table 3–6 lists pre- and post-upgrade values for Oracle HTTP Server, Oracle Enterprise Manager Application Server Control, and Oracle Application Server Web Cache.

***Table 3–6 Port Values Before and After Upgrade***

| Component | Port in Source Oracle Home | Port Value in Destination Oracle Home Assigned by Installer and Recorded in portlist.ini File | Post-Upgrade Port Value |
|---|---|---|---|
| Oracle HTTP Server | Port: 7777<br>Listen: 7778 | Port: 7783<br>Listen: 7784 | Port: 7777<br>Listen: 7778 |
| Oracle Enterprise Manager Application Server Control | 1810 | 1812 | 1812 |
| Oracle Application Server Web Cache | Administration: 4000<br>Invalidation: 4001<br>Statistics: 4002 | Administration: 4003<br>Invalidation: 4004<br>Statistics: 4005 | Administration: 4000<br>Invalidation: 4001<br>Statistics: 4002 |

## 3.11 Upgrading Application Server Clusters

Upgrading an application server cluster is a two-stage process. First, you must upgrade the configuration of one of the instances in the cluster from the source Oracle home to the newly installed 10*g* (9.0.4) middle tier instance in the destination Oracle home. Then, you reconstruct the cluster by installing and clustering additional instances in new destination Oracle homes.

Follow these steps to upgrade an application server cluster:

1. Determine the instance name of the instance to upgrade with the command:

   **<*source_MT_OH*>/dcm/bin/dcmctl listinstances**

2. Stop all processes in the source instance with the command:

   **<*source_MT_OH*>/dcm/bin/dcmctl shutdown**

3. Stop all processes in the destination instance with the command:

   **<*destination_MT_OH*>/dcm/bin/dcmctl shutdown**

4. Follow the instructions in Section 3.5, "Using the OracleAS Upgrade Assistant" on page 3-8.

5. Follow the instructions that apply to the upgraded configuration in Section 3.8, "Completing the Upgrade" on page 3-30.

6. Start the Oracle Enterprise Manager Application Server Control with the command:

   **<*destination_MT_OH*>/bin/emctl start iasconsole**

7. Display the OracleAS Instance Home Page.

8. Start the upgraded instance by following the instructions in Section 3.12, "Starting the Upgraded Middle Tier Instance" on page 3-74.

9. Create a cluster with the command:

   **<*destination_MT_OH*>/dcm/bin/dcmctl createcluster <*name of cluster*>**

10. Join the upgraded instance to the cluster with the command:

    **<*destination_MT_OH*>/dcm/bin/dcmctl joincluster -cl <*name of cluster*>**

11. Join each additional instance to the cluster with the command:

    **<*destination_MT_OH*>/dcm/bin/dcmctl joincluster -cl <*name of cluster*>**

12. If the upgraded configuration used `mod_oc4j.conf` for request routing, do the following:

    a. View the <*destination_MT_OH*>/Apache/Apache/conf/mod_oc4j.conf file in one of the instances, noting the instance and cluster names in the `Oc4jMount` directives.

    b. Change the instance (and, if necessary) cluster names to the instance name of the upgraded instance.

    c. Copy the `Oc4jMount` directives to the `mod_oc4.conf` file in each instance in the new cluster.

    d. Verify that requests that match the URL patterns in the `Oc4jMount` directives are routed to the correct instances.

## 3.12 Starting the Upgraded Middle Tier Instance

After the OracleAS Upgrade Assistant has finished processing, and you have completed all of the applicable manual upgrade tasks, start the upgraded middle tier instance. If the middle tier instance uses an Infrastructure, ensure that the Infrastructure is running. (Since the OracleAS Upgrade Assistant requires that the Infrastructure is running, this should already be the case, if not, follow the instructions in Section 3.5.2, "Starting the Infrastructure" on page 3-10.)

Follow these instructions to start the middle tier instance:

1. Start the application server components by issuing this command:

   **<*destination_MT_OH*>/opmn/bin/opmnctl startall**

   Oracle Process Management and Notification and all of the processes it manages are started (i.e., Distributed Configuration Management, Oracle HTTP Server, OC4J instances, OracleAS Web Cache, OracleAS Forms Services, and Oracle Application Server Reports Services).

2. Start the Application Server Control by issuing this command:

   **<*destination_MT_OH*>/bin/emctl start iasconsole**

### 3.12.1 Accessing the Web Clipping Test Provider Page (PDK 9.0.2.4.0 Only)

If you restored the Web Clipping provider as described in Section 3.8.4.7, "Restoring Portal Tools (OmniPortlet and Web Clipping) Providers" on page 3-55, follow these steps to access the Web Clipping Provider Test Page:

1. Access the following URL:

   `http://<host>:<port>/portalTools/webClipping/providers/webClipping`

   where <*host*> and <*port*> are the OracleAS Portal middle tier host name and port.

   The Web Clipping Provider Test page appears. If you have specified the same database for the Repository Target as that previously used for a PDK 9.0.2.4.0 installation, a Repository upgrade is also necessary. In this case, the Web Clipping Provider Test Page provides an **Upgrade (from 9.0.2.4.0)** link that you can click to perform installation of new tables and migration of existing Clipping Definitions to the latest versions.

   **Note:** If you perform the upgrade, the Clipping Definitions stored in the Web Clipping Repository will no longer work with PDK 9.0.2.4.0.

2. Click the **Upgrade (from 9.0.2.4.0)** link.

   Tables are added to the database schema hosting the Web Clipping repository, and all Definitions are upgraded.

### 3.12.2 Accessing the OmniPortlet Test Provider Page (PDK 9.0.2.4.0 Only)

If you restored the OmniPortlet provider as described in Section 3.8.4.7.2, "Restoring the OmniPortlet Provider Settings" on page 3-57, follow these steps to access the OmniPortlet Provider Test Page:

1. Access the following URL:

   `http://<host>:<port>/portalTools/omniPortlet/providers/omniPortlet`

   where <*host*> and <*port*> are the OracleAS Portal middle tier host name and port.

   The OmniPortlet Provider Test page appears.

2. Click the **Edit** link adjacent to the status of the Secured Data Repository. In the Repository Settings section of the Edit Provider: webClipping page, the database connection information for Web Clipping Repository is shown.
3. Click **OK** to save the settings.

   The OmniPortlet's `provider.xml` file is updated with the correct `<vaultId>` tag to save secured data into the repository.

### 3.12.3 Updating the OracleAS Portal Provider Information

Portal instances access web providers via a URL. The process of specifying this URL is referred to as provider registration. If the destination Oracle home will be accessed using a hostname and/or port number different from that of the source Oracle home, or the web providers have been deployed to a different URL path, then you need to update the URLs used to access the upgraded web providers. Web providers can be referenced by multiple portal instances; all of these must be updated.

Follow these steps to update the web provider URL:

1. Log on to OracleAS Portal as an administrator.
2. Click the **Navigator** link.

   The **Portal Navigator** page appears.
3. Click the **Providers** tab.
4. Click **Registered Providers**.

   A sorted list of registered providers appears.
5. Locate the provider to update, using the **Next** and **Previous** links if necessary.
6. Click the **Edit Registration** link for the provider to update.

   The **Edit Provider** page appears.
7. Click the **Connection** tab.
8. Update the URL to reflect the new location of the provider.
9. Click **OK** or **Apply** to save the changes.

### 3.12.4 Refresh the Event/Parameter Passing Samples Provider

The Event / Parameter Passing Samples Provider definition has changed since Release 2 (9.0.2). Consequently, the provider must be refreshed in the OracleAS Portal repository. Repeat these steps for each OracleAS Portal instance that references this provider.

Follow these steps to update the web provider URL:

1. Log on to OracleAS Portal as an administrator.
2. Click the **Navigator** link.

   The **Portal Navigator** page appears.
3. Click the **Providers** tab.
4. Click **Registered Providers**.

   A sorted list of registered providers appears.
5. Locate the JPDK V2 Sample Event Web Provider, using the **Next** and **Previous** links, if necessary.
6. Click the **Refresh** link for the JPDK V2 Sample Event Web Provider.

### 3.12.5 Resolving Oracle Call Interface Component Errors

If the upgrade succeeds, but components that use the Oracle Call Interface do not function properly, it may be that the `tnsnames.ora` file was not upgraded, or was upgraded incorrectly. Review the instructions and upgrade strategies presented in Section 3.8.9, "Upgrading the tnsnames.ora File" on page 3-57 and verify that the file contains all necessary entries for the components that use it.

# 3.13 Validating the Middle Tier Upgrade

This section describes tasks you should perform after the middle tier upgrade to validate that the upgrade was successful.

## 3.13.1 Verify Operation of Middle Tier Components

Follow these steps to verify that the middle tier components that were upgraded are started:

1. In a browser, access the Oracle Enterprise Manager Application Server Control in the 10*g* (9.0.4) middle tier Oracle home by entering its URL. Ensure that you provide the correct host name and port number. For example:

   ```
   http://midtierhost.mycompany.com:1812
   ```

   The Oracle Enterprise Manager page appears. A link for the middle tier instance appears in the **Standalone Instances** section.

2. Click the link.

   The **System Components** page appears.

3. Verify that the components are running.

4. Verify that the configuration information for the components in use is reflected in the 10*g* (9.0.4) Oracle home.

## 3.13.2 Check Significant URLs

Follow these steps to verify that you can access the Oracle HTTP Server and application URLs:

1. Verify that you can access the Oracle HTTP Server on the same host and port that you did in the previous release by entering the URL. Ensure that you provide the correct host name and port number. For example:

   ```
   http://midtierhost.mycompany.com:7777
   ```

2. Verify that you can access the URLs for the applications you operated in the previous release, and ensure that the applications are functioning as they did in the previous release.

## 3.14 Considerations for the Source Oracle Home After Upgrade

The upgrade process leaves the source Oracle home unchanged. Depending on the type of installation you have, and your future needs, you may elect to remove the source Oracle home, or to retain it for specific reasons. This section discusses decommissioning the source Oracle home, as well as reasons for retaining it, including the steps you must perform to revert to using it.

---

**Note:** If you retain the source Oracle home, you cannot operate it simulataneously with the destination Oracle home, because certain components have the same port values after upgraded. See Section 3.10, "Port Values and the portlist.ini File After Upgrade" on page 3-72.

---

### 3.14.1 Decommissioning the Source Oracle Home

When you are certain that the upgrade was successful, you have all of the necessary backups, and have no plans to revert to the source Oracle home, you may elect to remove the files from the source Oracle home. Use the Oracle Universal Installer to deinstall the instance.

> **See Also:** *Oracle9i Application Server Installation Guide* in the Release 2 (9.0.2) or (9.0.3) documentation library

Deinstalling a Release 2 (9.0.2) or (9.0.3) instance when there is also a 10*g* (9.0.4) instance on the computer requires a patch, and there are issues associated with this deinstallation that may apply to your configuration.

> **See Also:** *Oracle Application Server 10g Installation Guide*

#### 3.14.1.1 Preserving Application Files and Log Files

If there are application files or log files in the source Oracle home that are being referenced or used by the destination Oracle home, you should move them to another location before you decommission the source Oracle home, and, in the destination Oracle home, change any references to the files to the new location.

### 3.14.2 Retaining the Source Home for Future Language Loading

If you continue to operate an Oracle9iAS Release 2 (9.0.2) Portal repository, you should not decommission the Oracle9iAS Release 2 (9.0.2) Oracle home if there is a possibility that you might later want to load additional languages into the Oracle9iAS Release 2 (9.0.2) Portal repository. The utilities for loading languages in Oracle Application Server 10*g* (9.0.4) are not compatible with Oracle9*i*AS Portal in Release 2 (9.0.2).

### 3.14.3 Reverting to the Source Oracle Home: Resetting the Portal Service Monitoring Link

If you decide to use a Oracle9*i*AS Release 2 (9.0.2) middle tier again after the upgrade, then you will need to reset the Portal Service Monitoring link in the Services portlet on the Portal Builder page. The `monseed.sql` script is provided in the source Oracle home to accomplish this. Follow the steps below to run the `monseed.sql` script:

1. Set the ORACLE_HOME environment variable to the Oracle9*i*AS Release 2 (9.0.2) middle tier Oracle home.
2. Navigate to the *<source_MT_OH>*`/portal/admin/plsql/wwc/` directory.
3. Using SQL*Plus, connect to the Portal schema.
4. Issue this command:

   **`@monseed.sql`** ***<EM host> <EM port> <Portal DAD> <middle tier host> <middle tier port> <iAS instance name>***

   where

   - *<EM host>* is the hostname of Release 2 (9.0.2) Oracle Enterprise Manager
   - *<EM port>* is the port of Release 2 (9.0.2) Oracle Enterprise Manager
   - *<Portal DAD>* is the name of the Portal DAD (default is `portal`)
   - *<middle tier host>* is the hostname of Release 2 (9.0.2) middle tier
   - *<middle tier port>* is the ort of Release 2 (9.0.2) middle tier
   - *<iAS instance name>* is the instance name given to the Release 2 (9.0.2) middle tier at installation time. This name is found in the *<source_MT_OH>*`/ sysman/emd/targets.xml` file, in the Composite Membership segment of the HTTP Server target that is running the Oracle9iAS Portal

middle tier. Determine the HTTP Server target by finding the HTTP Server target with the ORACLE_HOME property matching the home of the HTTP Server servicing the Oracle9*i*AS Portal middle tier.

# 4

# Upgrading the Metadata Repository

This chapter explains how to upgrade the metadata repository, including schemas in customer databases and the OracleAS Portal repository. It is divided into these major sections:



---

**Note:** As a convention in this chapter, <*Infra_OH*> will be used to reference the active infrastructure Oracle home. This should be set to the Release 2 (9.0.2) Oracle home if the Metadata Repository upgrade is being performed on the infrastructure instance before an Identity Management upgrade is being performed. It should be set to the 10g (9.0.4) Oracle home if the Identity Management upgrade has already been performed on the infrastructure instance.

Specifically, <*Infra_OH*> references the Oracle home containing the database installation that holds the Metadata Repository. If the Metadata Repository exists in a separate installation from your Identity Management, then <*Infra_OH*> refers to the installation that contains the Metadata Repository.

---

# 4.1 Understanding the Metadata Repository Upgrade Process

Component schemas are contained in the Metadata Repository, along with the Identity Management schemas. However, the upgrade process for the component schemas is different from the upgrade process for the Identity Management schemas. Component schemas are upgraded by individual scripts, as shown in Figure 4–1, "Metadata Repository Upgrade". The Identity Management schemas are upgraded by the Oracle Universal Installer.

---

**Note:** Before the Metadata Repository can be upgraded, the 9.0.1.5 patch must be applied to the database.

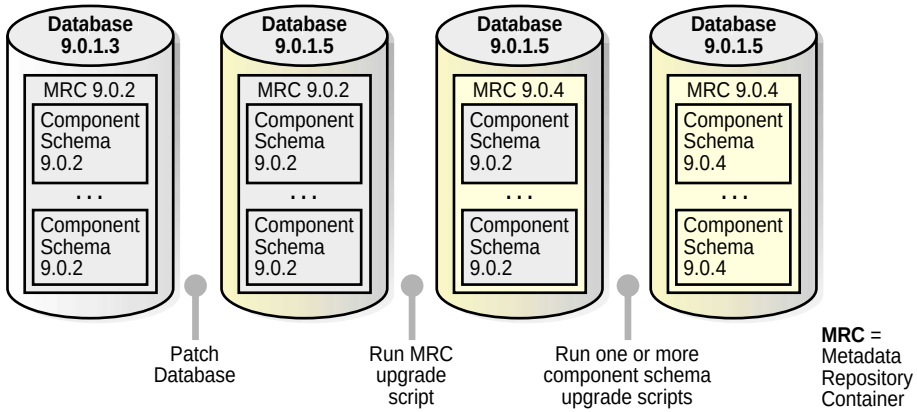
---

**See Also:** Section 5.1.2, "Understanding the Identity Management Upgrade Processes" on page 5-5

The Metadata Repository upgrade consists of these steps:

1. The Metadata Repository Container Upgrade script is run.
2. One or more component upgrade scripts are run. The component upgrade scripts that are run will depend on the components configured. See Section 4.3, "Performing the Metadata Repository Upgrade" on page 4-10 for a suggested approach for your installation type.

***Figure 4–1 Metadata Repository Upgrade***



## 4.2 Preparing to Upgrade the Metadata Repository

Before you begin any other Metadata Repository upgrade tasks, perform these steps in the Infrastructure Oracle home.

1. Stop all processes.
2. Back up the database.
3. Install the RDBMS 9.0.1.5 patch set against the Infrastructure Oracle home (if it has not already been installed as part of an Identity Management upgrade).
4. Ensure that there are no invalid objects in the database.

   **See Also:** Section 5.6.1, "Executing the utlrp.sql Utility" on page 5-63 and Section 5.6.2, "Checking for Invalid Database Objects" on page 5-63

5. Obtain the Repository Creation Assistant CD-ROM.
6. Start the database server and listener.
7. Install the DBMS_IAS_UPGRADE package.

   **See Also:** Section 4.2.2, "Loading the DBMS_IAS_UPGRADE Package" on page 4-4

8. Upgrade the Metadata Repository Container.

   **See Also:** Section 4.2.3, "Upgrading the Metadata Repository Container" on page 4-5

9. Ensure that there are no invalid objects in the database.

   **See Also:** Section 5.6.1, "Executing the utlrp.sql Utility" on page 5-63 and Section 5.6.2, "Checking for Invalid Database Objects" on page 5-63

### 4.2.1 Setting the Environment for Upgrading the Metadata Repository

In order to execute most steps in the Metadata Repository upgrade, it is necessary to set your environment to point to the infrastructure Oracle home. This means setting the ORACLE_HOME environment variable to <*Infra_OH*> and setting the ORACLE_SID environment variable to the instance name for the Infrastructure database. The easiest way to accomplish this is to execute one of the environment scripts, `coraenv` or `oraenv`. `coraenv` can be used to set the environment for `csh` shells. `oraenv` can be used for other shells.

### 4.2.2 Loading the DBMS_IAS_UPGRADE Package

Before you can upgrade to the 10*g* (9.0.4) Infrastructure, you must load a PL/SQL package called `DBMS_IAS_UPGRADE`. This package allows the schema upgrade scripts to grant permissions when they are run as user `SYS`.

Follow these steps to load the package:

1. Ensure that the database and listener are running.
2. Ensure that the ORACLE_HOME environment variable is set to <*Infra_OH*> and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
3. Change directories to <*repCA_CD*>`/repCA/rdbms/admin`.
4. Connect to SQL*Plus as user `SYS`.
5. Issue these commands:

   **`@dbmsiasu.sql`**

   **`@prvtiasu.plb`**

   The following messages appear:

   ```
   Package created.

   Package body created.
   ```

   The PL/SQL package `SYS.DBMS_IAS_UPGRADE` is installed.

### 4.2.3 Upgrading the Metadata Repository Container

The Metadata Repository Container upgrade process (the `mrc.pl` script) performs two functions:

- Creates new tablespaces and schemas in the metadata repository (`ias_meta`, `wcrsys_ts`, `ocats`, `ip_dt`, `ip_rt`, `ip_idx`, `ip_lob` `OLTS_SVRMGSTORE`, `oltsbattrstore`) tablespaces and `wcrsys`, `oca`, `oraoca_public`, `ip`, `wk_test` and `internet_appserver_registry` schemas).

  **Note:** If you manually created any of the items listed below after the Oracle9*i*AS Release 2 (9.0.2) Infrastructure installation, then you must move them to a different location, or remove them before you run the Metadata Repository Container upgrade:

  Tablespaces: `ias_meta`, `wcrsys_ts`, `ocats`, `ip_dt`, `ip_rt`, `ip_idx`, `ip_lob`, `OLTS_SVRMGSTORE, olts_battrstore`

  Schemas: `wcrsys`, `oca`, `oraoca_public`, `ip`, `wk_test` and `internet_appserver_registry`

  Otherwise, the Metadata Repository Container upgrade will fail. A similar issue is faced by users of the OracleAS RepCA, and is described in detail in the *Oracle Application Server 10g Installation Guide*, sections "Schema Name Already in Use" and Tablespace Name Already in Use".

- Updates the Oracle Internet Directory entry for the repository to accommodate the 10*g* (9.0.4) security architecture

Because the metadata repository and Oracle Internet Directory may reside on different computers, and require different access rights, the script is designed to perform only one of the functions, or both, depending on the credentials given when starting the script.

The Metadata Repository Creation script must be executed before any other schema upgrade scripts are executed, because the new schemas depend on the modifications made by `mrc.pl`.

Depending on the configuration to be upgraded, you will perform one of the following procedures:

- If you have DBA credentials for the metadata repository database, follow the instructions in Section 4.2.3.1, "Executing mrc.pl for New Schema Creation" on page 4-6.
- If you have administrative credentials for the Oracle Internet Directory, follow the instructions in Section 4.2.3.2, "Executing mrc.pl for Oracle Internet Directory Entry Update" on page 4-7.
- If you have DBA credentials for the metadata repository database and administrative credentials for the Oracle Internet Directory, and want to perform both upgrade functions, follow the instructions in Section 4.2.3.3, "Executing mrc.pl for New Schema Creation and Oracle Internet Directory Entry Update" on page 4-8.

### 4.2.3.1 Executing mrc.pl for New Schema Creation

Follow these steps to create support for new schemas in the metadata repository:

1. Ensure that the database and listener are running.
2. Ensure that the ORACLE_HOME environment variable is set to *<Infra_OH>* and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
3. Change directories to *<repCA_CD>*/repCA/mrc/upgrade.
4. Ensure that there is an existing directory with write permission enabled in which to create new database files for the new tablespaces. (You will specify this directory as part of the command to start the script.)
5. Issue this command:

   **<*Infra_OH*>/perl/bin/perl mrc.pl d -dbpwd <*SYS user password*> -connstring <*conn string*> -dspace <*dir*>**

   where:

   - *<SYS user password>* is the dba password
   - *<conn string>* is the database connect string in the format host:port:SID
   - *<dir>* is the directory in which you want to create the new database files that contain the new tablespaces

   for example:

```
<Infra_OH>/perl/bin/perl mrc.pl d -dbpwd "change_on_
install" -connstring mycompany.com:1521:iasdb2 -dspace
/private1/mydir/ORA_IAS_902
```

You need not specify all of the above parameters on the command line. The script will prompt you to enter any parameters for which a default value is not available. For example, you may prefer not to type the password on the command line, but enter it only when prompted, so it is not displayed on the screen in clear text.

If creation of new schemas was successful, the following message appears:

```
Creation of new schemas was successful
```

#### 4.2.3.2 Executing mrc.pl for Oracle Internet Directory Entry Update

Follow these steps to update the Oracle Internet Directory entry:

1. Ensure that `mrc.pl` has been run to create new schemas, as described in Section 4.2.3.1, "Executing mrc.pl for New Schema Creation" on page 4-6.
2. Ensure that the database, listener, and Oracle Internet Directory are running.
3. Ensure that the ORACLE_HOME environment variable is set to *<Infra_OH>* and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
4. Change directories to *<repCA_CD>*`/repCA/mrc/upgrade`
5. Issue this command:

   **`<Infra_OH>/perl/bin/perl mrc.pl u -ousr <oid admin user name> -opwd <oid admin password> -connstring <conn string>`**

   where:

   - *<oid admin user name>* is the Oracle Internet Directory administrative user name
   - *<oid admin password>* is the Oracle Internet Directory administrative user password
   - *<conn string>* is the database connect string in the format host:port:SID

   for example:

   ```
   <Infra_OH>/perl/bin/perl mrc.pl u -ousr cn=orcladmin -opwd
   welcome1 -connstring "slocum.us.oracle.com:1521:iasdb2"
   ```

If creation of OID entries and update of randomized passwords was successful, the following message appears:

```
"Update of new OID entries successful"
```

### 4.2.3.3 Executing mrc.pl for New Schema Creation and Oracle Internet Directory Entry Update

This method of executing `mrc.pl` combines the `d` and `u` options in one procedure. You should not perform this combined procedure if you have run `mrc.pl` with either the `d` option or the `u` option (as described in Section 4.2.3.1 on page 4-6 and Section 4.2.3.2 on page 4-7). Follow these steps to create new schemas in the metadata repository and update the Oracle Internet Directory entry:

1. Ensure that the database, listener, and Oracle Internet Directory server are running.
2. Ensure that the ORACLE_HOME environment variable is set to *<Infra_OH>* and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
3. Change directories to *<repCA_CD>*`/repCA/mrc/upgrade`.
4. Ensure that there is an existing directory with write permission enabled in which to create new database files for the new tablespaces. (You will specify this directory as part of the command to start the script.)
5. Issue this command:

   ***<Infra_OH>*`/perl/bin/perl mrc.pl du -dbpwd <`*SYS user password*`> -dspace <`*tablespace directory*`> -ousr <`*oid admin user name*`> -opwd <`*oid admin user password*`> -connstring <`*database connect string*`>`**

   where:

   - *<SYS user password>* is the dba password
   - *<tablespace directory>* is an existing directory in which you want the files that contain the new tablespaces to be created
   - *<oid admin user name>* is the Oracle Internet Directory administrative user name

- *<oid admin password>* is the Oracle Internet Directory administrative user password
- *<conn string>* is the database connect string in the format host:port:SID

#### 4.2.3.4 Resolving Errors

Errors may occur during the upgrade process. Common errors returned by the upgrade script and their resolution are listed below.

**java.sql.SQLException: ORA-01034 ORACLE not available.**

**Cause:** The database is not running.

**Action:** Start the database.

**java.sql.SQLException: Io exception: The Network Adapter could not establish the connection.**

**Cause:** The listener is not running.

**Action:** Start the listener.

**Invalid OID password.**

**Cause:** The Oracle Internet Directory superuser password is incorrect.

**Action:** Provide the correct password.

## 4.3 Performing the Metadata Repository Upgrade

After you have performed all of the preparatory steps for upgrading the Metadata Repository, as described in Section 4.2, "Preparing to Upgrade the Metadata Repository" on page 4-3, you can proceed with executing the scripts to upgrade the schemas in the Metadata Repository for the components that you plan to use in Oracle Application Server 10*g* (9.0.4). Note that:

- You may elect to upgrade one component's schema, test that upgrade, and then upgrade another component's schema and test its upgrade, and so on, until the upgrade is complete. Alternatively, you may elect to upgrade the components' schemas in succession, and then test the entire upgrade at once.
- You may upgrade a subset of all components in the Metadata Repository, as best suits your needs, but the Metadata Repository is not considered to be at version 10*g* (9.0.4) until all component schemas have been upgraded.

Follow these steps to execute the component upgrade script(s) of your choice:

1. Stop all of the middle tiers that are using the Infrastructure in which the Metadata Repository resides, using these steps:
   a. Stop the Oracle Enterprise Manager Application Server Control with this command:

      ```
      <destination_MT_OH>/bin/emctl stop iasconsole
      ```

   b. Stop OPMN and the processes managed by it with this command:

      ```
      <destination_MT_OH>/opmn/bin/opmnctl stopall
      ```

   c. Stop all other running processes in the destination middle tier Oracle home.
2. Follow the instructions in the relevant sections to upgrade the component schemas of of your choice:
   - Section 4.4.1, "Executing the Process Connect Schema Upgrade Script" on page 4-13
   - Section 4.4.2, "Executing the Oracle Application Server Certificate Authority Schema Upgrade Script" on page 4-14
   - Section 4.4.3, "Executing the Oracle Ultra Search Schema Upgrade Script" on page 4-16
   - Section 4.4.5, "Executing the Oracle Application Server Syndication Server Upgrade Script" on page 4-24

- Section 4.4.6, "Executing the Oracle Application Server Web Services UDDI Registry Schema Upgrade Scripts" on page 4-25
- Section 4.4.7, "Executing the Web Clipping Schema Upgrade Script" on page 4-29
- Section 4.5, "Upgrading the OracleAS Portal Repository" on page 4-31

3. Start the middle tier application server components by issuing these commands:

   a. Start OPMN and the processes managed by it with this command:

      ***<destination_MT_OH>*/opmn/bin/opmnctl startall**

      Oracle Process Management and Notification and all of the processes it manages are started (i.e., Distributed Configuration Management, Oracle HTTP Server, OC4J instances, OracleAS Web Cache, Oracle Application Server Forms Services, and Oracle Application Server Reports Services).

   b. Start the Oracle Enterprise Manager Application Server Control by issuing this command:

      ***<destination_MT_OH>*/bin/emctl start iasconsole**

## 4.4 Executing Metadata Repository Upgrade Scripts

This section contains procedures for executing each component upgrade script provided on the Repository Creation Assistant CD-ROM to upgrade schemas in the Metadata Repository. It applies only to configurations that use the Infrastructure database, and have these components configured.

One or more of these procedures are intended to be followed as Step 2 of the procedure outlined in Section 4.3, "Performing the Metadata Repository Upgrade" on page 4-10.

---

**Note:** Each of the configured components has a set of verification steps to determine whether the upgrade of the component was successful. Ensure that you perform all verification steps for each component that you upgrade to 10*g* (9.0.4).

---

The following topics are included:

### 4.4.1 Executing the Process Connect Schema Upgrade Script

Follow the steps below to upgrade the Process Connect schema:

1. Create the directory <*Infra_OH*>/ip/admin.
2. Change directories to <*repCA_CD*>/repCA/ip/admin.
3. Ensure that the ORACLE_HOME environment variable is set to <*Infra_OH*> and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
4. Ensure that the PATH environment variable includes <*Infra_OH*>/bin.
5. Set the REPCA_ORACLE_HOME environment variable to point to <*repCA_CD*>/repCA, which is the path to the Oracle Application Server 10*g* (9.0.4) Repository Creation Assistant CD-ROM.
6. Start the upgrade script with the command below:

   **upgradeInfra.sh**

   A prompt for the SYS password appears.
7. Provide the SYS password.

   The script writes all output to the screen and to the <*Infra_OH*>/ip/admin/ipupgrade.log file. The process takes a few minutes to complete. When it is finished, the following message appears:

   ```
   PL/SQL procedure successfully completed.
   ```

   The log file should contain no error messages. If the upgrade was unsuccessful, the file will contain one or more of the messages described below.

**ORA-00942: table or view does not exist**

**Cause:** The connected database user does not have privileges to access IP schema.

**Action:** Run the script again with SYS user privileges.

**ORA-24010: QUEUE xxx_QUEUE does not exist**

**Cause:** Queue xxx_QUEUE does not exist in the repository.

**Action:** Ignore this error. It occurs because xxx_QUEUE was dropped before xxx_QUEUE was created.

**ORA-24002: QUEUE_TABLE xxx does not exist**

**Cause:** QUEUE TABLE xxx does not exist in the repository.

**Action:** Ignore this error. It occurs because xxx_QUEUE was dropped before xxx_QUEUE was created.

#### 4.4.1.1 Verifying the Upgrade Results

Follow these steps to verify the upgrade results:

1. Log in to SQL*Plus as `sysdba`.
2. Issue the following commands:

   ```
   SET SERVEROUTPUT ON
   execute ip.validate_components
   ```

   One of the following messages appears:

   ```
   SUCCESSFUL upgrade of schema: IP
   FAILED upgrade of schema: IP
   ORA-20000: VERIFICATION FAILED
   ```

   Other messages may appear, identifying missing columns or tables.

### 4.4.2 Executing the Oracle Application Server Certificate Authority Schema Upgrade Script

Follow the steps below to upgrade the Oracle Application Server Certificate Authority schema. Before you start the script, ensure that the `open_cursors` setting in `<destination_Infra_OH>/dbs/init<sid>.ora` has a value of 300 or more.

1. Ensure that the ORACLE_HOME environment variable is set to `<Infra_OH>` and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
2. Ensure that the database and the listener are running.
3. Change directories to `<repCA_CD>/repCA/oca/sql/`.
4. Start the upgrade script with the command below:

   ```
   sqlplus "system/<your system password as sysdba>" @ug_
   ocadocat.sql
   ```

where *<your system password as sysdba>* is the password of the sytem user.

#### 4.4.2.1 Verifying the Upgrade Results

Follow these steps to verify that the upgrade was successful:

1. Issue this command:

   ***sqlplus "system/<your system password as sysdba>"***

   where *<your system password as sysdba>* is the password of the system user.

2. Issue this command:

   **select table_name, owner from all_tables where owner='OCA';**

   If the upgrade was successful, the following appears:

***Example 4–1 Verification of Oracle Application Server Certificate Authority Schema Upgrade Results***

```
TABLE_NAME                                                OWNER
----------------------------------------------------------------
OCA_CONFIG                                                OCA
OCA_PARAMETER_CONFIG                                      OCA
OCA_TASKS                                                 OCA
OCA_TASKS_EXECUTION                                       OCA
OCM_CERTIFICATE                                           OCA
OCM_CERTIFICATE_META                                      OCA
OCM_CERTIFICATE_REQUEST                                   OCA
OCM_CERTIFICATE_REQUEST_META                              OCA
OCM_CERTIFICATE_SERIAL                                    OCA
OCM_ERROR_LOG                                             OCA
OCM_JAZN                                                  OCA
11 rows selected.
```

A detailed description of the processing performed by this script is included in Appendix A.2.4, "The Oracle Application Server Certificate Authority Upgrade Process" on page A-21.

## 4.4.3 Executing the Oracle Ultra Search Schema Upgrade Script

This section explains how to upgrade the Oracle Ultra Search schema in the metadata repository.

### 4.4.3.1 Preparing to Upgrade the Oracle Ultra Search Schema

Before you upgrade the schema, prepare the system by performing the steps below.

> **See Also:** *Oracle Ultra Search User's Guide* for instructions on using the Oracle Ultra Search Administration Tool.

1. Start the middle tier, if necessary, by following step 3 in Section 4.3, "Performing the Metadata Repository Upgrade" on page 4-10.
2. Disable all Oracle Ultra Search synchronization schedules.
   a. Log in to the Oracle Enterprise Manager Application Server Control.
   b. In the **Standalone Instances** section, select the desired middle tier instance.

      The system components of the instance are displayed.
   c. In the **System Components** section, select **OC4J_Portal** and click the **Start** button to start the OC4J instance.
   d. Navigate from the Portal Instance to Ultra Search instance using Oracle Enterprise Manager Application Server Control. Alternatively, you can log in to the Oracle Ultra Search Administration tool by accessing this URL:

      `http://<host>:<port>/ultrasearch/admin`
   e. In the **Schedules** tab, stop and disable all crawler synchronization schedules in all Oracle Ultra Search instances.
3. Back up the Oracle Ultra Search metadata in WKSYS and the user instance schemas. If unrecoverable errors occur during the upgrade, you will then be able to restore Oracle Ultra Search to the previous version.
4. Stop the middle tier by following Step 1 in Section 4.3, "Performing the Metadata Repository Upgrade" on page 4-10.

#### 4.4.3.2 Upgrading the Oracle Ultra Search Schema

The schema upgrade process consists of running the the following scripts:

- `upgradeInfra.sh` — the script that upgrades the schema
- `usca.sh` — the post-upgrade configuration script

Follow these steps to run the Oracle Ultra Search schema upgrade script:

1. Ensure that the ORACLE_HOME environment variable is set to <*Infra_OH*> and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
2. Set the REPCA_ORACLE_HOME environment variable to point to <*repCA_CD*>/repCA, which is the path to the Oracle Application Server 10*g* (9.0.4) Repository Creation Assistant CD-ROM.
3. Ensure that the PATH environment variable includes <*Infra_OH*>/bin.
4. Change directories to <*repCA_CD*>/repCA/ultrasearch/setup.
5. Ensure that the `upgradeInfra.sh` script is executable. Start the script with the command:

   **`./upgradeInfra.sh`**

   or

   **`./upgradeInfra.sh -l <log file path>.log`**

   where <*log file path*> is the path and filename to which the script will write log entries. For example:

   ```
   ./upgradeInfra.sh -l /tmp/log/upgradeInfra.log
   ```

   If you do not specify the absolute path, providing only a file name (e.g., `./upgradeinfra.sh -l upgradeInfra.log`), then the log file `upgradeInfra.log` will be placed in the <*Infra_OH*> directory.

   If you do not provide a file name, then the log file is written to <*Infra_OH*>/ultrasearch/setup/upgrade.log.

   A password prompt appears.
6. Provide the password.

   A verification prompt appears.
7. Answer the prompt.

The script echoes and prompts you to verify your answers. When the process finishes, this message appears:

```
End of Upgrade - Ultra Search
```

---

**Note:** If you do not answer the verification prompts, the script will continue by default. You can enter `n` to stop the script at any point, and then start the script again if you have entered an incorrect answer.

---

8. Log in to the 10*g* (9.0.4) middle tier computer and set the environment variables as indicated below. These are required by the `usca.sh` script.

---

**Note:** If you have multiple middle tiers, you need only perform this step and the remaining steps on one of them.

---

   - Set `ORACLE_HOME` to <*destination_MT_OH*>.
   - Set the library path variable to include the path <*destination_MT_OH*>`/lib`.

9. Change directories to <*destination_MT_OH*>`/ultrasearch/setup`.

10. Ensure that the `usca.sh` script is executable. Start the script with the command:

    **`./usca.sh -action backend_upgrade -user cn=orcladmin -password`** <***password of the OID user 'orcladmin'***>

11. Answer the prompts.

    The script echoes and prompts you to verify each answer.

### 4.4.3.3 Configuring the Oracle Ultra Search Schema Metadata

Configuring the schema metadata to use the Oracle Application Server 10*g* (9.0.4) Infrastructure Oracle home may require one, or both, of the procedures in this section, depending on which part of the Infrastructure is upgraded first:

- If you upgrade the Metadata Repository first, you need only perform the steps in Section 4.4.3.3.1, "Installing the Java Runtime Environment (JRE) 1.4 and Configuring Oracle Ultra Search to Use JRE 1.4" on page 4-19.

- If you upgrade the Identity Management services first, and the Metadata Repository is using the Oracle Application Server 10*g* (9.0.4) Infrastructure OracleHome, then you need only perform the steps in Section 4.4.3.3.2, "Executing the Schema Configuration Script" on page 4-20 to configure Ultra Search to use the Oracle Application Server 10*g* (9.0.4) Infrastructure Oracle home. If the Metadata Repository is still using the Oracle9*i*AS Release 2 (9.0.2) Infrastructure Oracle home, then you need only perform the steps in Section 4.4.3.3.1, "Installing the Java Runtime Environment (JRE) 1.4 and Configuring Oracle Ultra Search to Use JRE 1.4" on page 4-19.

#### 4.4.3.3.1 Installing the Java Runtime Environment (JRE) 1.4 and Configuring Oracle Ultra Search to Use JRE 1.4

Follow these steps to install the Java Runtime Environment (JRE) 1.4 and configure Oracle Ultra Search to use it.

1. Navigate to the directory `<repCA_CD>/repCA/ultrasearch/jre_upgrade`.
2. Start the Oracle Universal Installer with the following command:

   ```
   ./runInstaller
   ```

   The **Welcome** screen appears.
3. Click **Next**.

   The **Specify File Locations screen** appears.
4. Enter a new Oracle home name in the **Name** field and a full path in the **Path** field. (You cannot install JRE 1.4 into the Infrastructure Oracle home location.)
5. Click **Next**.

   The **Summary** screen appears.
6. Click **Install**.

   The installation begins. The progress bar and a message indicate when it is complete; then the **End of Installation** screen appears.
7. Click **Exit**.
8. Log in to the database as SYSDBA.

9. Issue the following commands:

```
alter session set current_schema = wksys;

update wksys.wk$crawler_config_default set ccd_pvalue
='<new java execution path>' where ccd_pname = 'CC_JAVA_
EXEC_PATH';

exec wk_util.populate_crawler_config;

commit;
```

where:

- <*new java execution path*> is:

  <*JRE 1.4 directory*>/jre/1.4.1/bin/java -ms16m -mx256m -Djava.library.path=<*destination_Infra_OH*>/lib

***Example 4–2 JRE Installation and Oracle Ultra Search Configuration Command***

```
update wksys.wk$crawler_config_default set ccd_pvalue = '/private/jre_
home/jre/1.4.1/bin/java -ms16m -mx256m  -Djava.library.path=/private/infra_
home/lib' where ccd_pname = 'CC_JAVA_EXEC_PATH';
```

---

**Note:** Completing this procedure resets the crawler cache directory and crawler log directory to the default values. You can use the Oracle Ultra Search Administration tool to update these values.

---

**4.4.3.3.2 Executing the Schema Configuration Script** Follow these steps to configure Ultra Search to use the Oracle Application Server 10*g* (9.0.4) Infrastructure OracleHome.

1. Set the ORACLE_HOME environment variable to the <*destination_Infra_OH*>.
2. Set the ORACLE_SID environment variable to the Oracle Application Server 10*g* (9.0.4) Infrastructure database.
3. Change directories to <*destination_Infra_OH*>/ultrasearch/admin>.
4. Issue this command:

```
sqlplus "sys/<SYS user password> as sysdba"
```

5. Issue this command:

   ```
   @wk0config.sql <WKSYSPW> <JDBC_CONNSTR> <LAUNCH_ANYWHERE>
   ""
   ```

   where:

   - <*WKSYSPW*> is the password for the WKSYS schema. If you don't know the password, then you can click the **Change Schema Password** link on the **Infrastructure** page of the Oracle Enterprise Manager Application Server Control.
   - <*JDBC_CONNSTR*> is the JDBC connection string, in net description (TNS keyword-value) format, for example:

     ```
     (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=...)
     (PORT=1521)...))
     ```

   - <*LAUNCH_ANYWHERE*> is the mode of the Metadata Repository. Setting it to TRUE indicates that the Metadata Repository is in Real Application cluster mode. For this procedure, you should set it to FALSE.

   ---

   **Note:** Completing this procedure resets the crawler cache directory and crawler log directory to the default values. You can use the Oracle Ultra Search Administration tool to update these values.

   ---

#### 4.4.3.4 Verifying the Upgrade Results

All errors are captured in the log file <*Infra_OH*>/ultrasearch/setup/*log file name*.log. If there are no errors, you can verify the upgrade results by starting Oracle Ultra Search. Follow the steps below:

1. Start the middle tier, if necessary, by following Step 3 in Section 4.3, "Performing the Metadata Repository Upgrade" on page 4-10.
2. Access the URL below to start the Oracle Ultra Search administration software:

   ```
   http://host:port/ultrasearch/admin/index.jsp
   ```

3. Use the 10*g* (9.0.4) features described in the *Ultra Search User's Guide*.
4. Use the Oracle Ultra Search sample search application:

   ```
   http://host:port/ultrasearch/query/search.jsp
   ```

5. Enable and resume the crawler synchronization schedules in all Oracle Ultra Search instances.

#### 4.4.3.5 Resolving Errors

Errors may occur during the upgrade process. Common errors returned by the upgrade script and their resolution are listed below.

**ERROR: Please set the environment variable ORACLE_HOME**

**Cause:** The ORACLE_HOME environment variable is not set.

**Action:** Set the ORACLE_HOME environment variable and run the script again.

**ERROR: Please set the environment variable ORACLE_SID**

**Cause:** The ORACLE_SID environment variable is not set.

**Action:** Set the ORACLE_SID environment variable and run the script again.

**ERROR: Please set the environment variable REPCA_ORACLE_HOME**

**Cause:** The REPCA_ORACLE_HOME environment variable is not set.

**Action:** Set the REPCA_ORACLE_HOME environment variable and run the script again.

**ERROR: Unable to open file "<file name>"**

**Cause:** The script cannot open the file.

**Action:** Verify the ORACLE_HOME environment variable setting and ensure that the file exists at that location.

A detailed description of the processing performed by these scripts is included in Appendix A.2.5, "The Oracle Ultra Search Schema Upgrade Process" on page A-21.

### 4.4.4 Upgrading Oracle Ultra Search from a Release 9.0.1 or 9.2 Database

If you have multiple database instances of version 9.0.1 or 9.2 in the same Oracle home, and then install the Oracle Application Server 10*g* (9.0.4) Metadata Repository on one of the database instances, the Repostiory Configuration Assistant will overwrite the Oracle home with the Oracle Application Server 10*g* (9.0.4) Oracle Ultra Search files. If another database instance is still hosting the Oracle Ultra Search version 9.0.1 or 9.2, then the Oracle Ultra Search running on that database instance will not function.

You can resolve this problem by upgrading the 9.0.1 or 9.2 Oracle Ultra Search to Oracle Application Server 10*g* (9.0.4). Follow these steps:

1. Mount the Repository Creation Assistant CD-ROM on the database computer.
2. Change directories to `<repCA_CD>/repCA/ultrasearch/admin`
3. Set the ORACLE_SID environment variable to the 9.0.1 or 9.2 database instance (not the 10*g* (9.0.4) Metadata Repository).
4. Use SQL*Plus to log in to the database as user SYSDBA.
5. Issue this command:

   **`@wkdbmig.sql`**

6. Install an Oracle Application Server 10*g* (9.0.4) middle tier of the Portal and Wireless installation type. When prompted for an Infrastructure to use, select the instance that was overwritten by the Repository Configuration Assistant (not a 9.0.1 or 9.2 instance).
7. Update the `data-sources.xml` and `ultrasearch.properties` files to connect to the database instance that hosts the upgraded Oracle Ultra Search.

**See Also:** *Oracle Ultra Search User's Guide*, Section 3.4.4, "Installing the Middle Tier with the Oracle Application Server Release" for instructions on updating the data-sources.xml and ultrasearch.properties files.

---

**Note:** The upgraded Oracle Ultra Search does not support Oracle Application Server Single Sign-On. The Ultra Search administrators can only be database users.

---

## 4.4.5 Executing the Oracle Application Server Syndication Server Upgrade Script

Follow the steps below to upgrade the Oracle Application Server Syndication Server schema.

1. Ensure that the ORACLE_HOME environment variable is set to <*Infra_OH*> and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
2. Determine whether the Oracle Application Server Syndication Server schema requires upgrade:
   a. Change directories to <*repCA_CD*>/repCA/syndication/admin
   b. Issue the command:

      **sqlplus "sys as sysdba" @ossupg902chk.sql**

      If the schema version is current, a message explains that the script can only upgrade from 9.0.2 to 9.0.4.
3. Ensure that the middle tier Syndication Services is stopped.
4. Start the upgrade script with the commands below:

   **cd <*repCA_CD*>/repCA/syndication/admin**

   **sqlplus "sys as sysdba" @ossupg902.sql**

### 4.4.5.1 Verifying the Upgrade Results

If the processing was successful, messages such as 'SUCCESSFUL install/upgrade of schema: DSGATEWAY' will appear on the screen. Otherwise, the message reads 'FAILED install/upgrade of schema: DSGATEWAY'. For all errors, a message is printed with the error code (e.g. ORA-XXXX) and the line number.

A detailed description of the processing performed by these scripts is included in Appendix A.2.7, "The Oracle Application Server Syndication Server Schema Upgrade Process" on page A-24.

## 4.4.6 Executing the Oracle Application Server Web Services UDDI Registry Schema Upgrade Scripts

This section explains how to upgrade the Oracle Application Server Web Services UDDI Registry schema in the metadata repository.

### 4.4.6.1 Upgrade Scenarios

There are two possible UDDI schema upgrade scenarios:

- Oracle Application Server Release 2 (9.0.2). Upgrade from this version is performed with the `wuru9020.sql` script.
- Oracle Application Server Release 2 (9.0.2), with the UDDI 9.0.2.3 patch number 2440287 (available on MetaLink) applied. Upgrade from this version is performed with the `wuru9023.sql` script.

**4.4.6.1.1 Upgrading From Release 2 (9.0.2) using wuru9020.sql** This upgrade script performs the following tasks:

- Destroys the contents of the UDDI schema (tables, PL/SQL packages, and so on)
- Invokes the 10*g* (9.0.4) installation script, which installs the 10*g* (9.0.4) schema

> **Note:** This upgrade destroys all existing data. To preserve existing data, use the UDDI `get_<xx>` API to get the existing entries, and then use `uddiadmin.jar` to import them after the upgrade.

**4.4.6.1.2 Upgrading From Release 2 (9.0.2) with UDDI 9.0.2.3 Patch using wuru9023.sql** This upgrade script performs the following tasks:

- Performs all initial structural changes required for the upgrade, such as creating and dropping tables, columns, and indexes.
- Performs all upgrades to the UDDI server properties and related configuration tables, such as inserting new rows and updating existing rows.

A detailed description of the processing performed by these scripts is included in Appendix A.2.8, "The Oracle Application Server Web Services UDDI Registry Schema Upgrade Process" on page A-24.

### 4.4.6.2 Upgrading the UDDI Schema

Follow the steps below to upgrade the UDDI schema:

1. Ensure that the UDDI database is backed up. There are several ways to do this; one example is provided below. Consult a database administrator or refer to Oracle documentation for more information.

***Example 4–3 Using the export Utility to Back Up UDDISYS Schema***

```
exp system FILE=uddisys.dmp OWNER=uddisys GRANTS=y ROWS=y
```

2. Ensure that the ORACLE_HOME environment variable is set to *<Infra_OH>* and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
3. Set the SRC_HOME environment variable to point to *<repCA_CD>*, which is the path to the Oracle Application Server 10*g* (9.0.4) Repository Creation Assistant CD-ROM.
4. Determine the UDDI schema version with these commands:

   **cd *<repCA_CD>*/repCA/uddi/admin**

   **sqlplus "sys as sysdba" @wurup.sql**

   - If the version is 9.0.2.1, use the `wuru9020.sql` script in Step 6.
   - If the version is 9.0.2.3, use the `wuru9023.sql` script in Step 6.
5. Ensure that the middle tier UDDI is stopped.

6. Start the upgrade script with the commands below:

   **cd <*repCA_CD*>/repCA/uddi/admin**

   **sqlplus "sys/<*password*> as sysdba"**

   **spool <*directory*>/<*log file name*>.log**

   > **Note:** You must specify a directory for the log file other than the current directory, that for the CD-ROM, since the CD-ROM is read only.

   ***(If schema version is 9.0.2.1)*** `@wuru9020.sql`

   ***(If schema version is 9.0.2.3)*** `@wuru9023.sql`

   **quit**

   The script writes all output to the <*directory*>/<*log file name*>.log file.

**4.4.6.2.1 Additional Steps for Version 9.0.2.1 Upgrades** If you upgraded from schema version 9.0.2.1, perform these steps:

1. Determine the character set used in the database with this command:

   **sqlplus "sys/<*password*> as sysdba"**

   ```
   select VALUE from NLS_DATABASE_PARAMETERS where PARAMETER =
   'NLS_CHARACTERSET';

   quit
   ```

2. Load taxonomy data with the commands below:

   **cd <*repCA_CD*>/repCA/uddi/admin**

   **sqlldr userid=sys control=naics-97.ctl log=<*directory*>/<*naics-logfilename*>.log bad=<*bad file name*>.log**

   **sqlldr userid=sys control=unspsc-73.ctl log=<*directory*>/<*unspsc-logfilename*>.log bad=<*bad file name*>.log**

> **Note:** You must specify a directory for the log file other than the current directory (that for the CD-ROM), because the CD-ROM is read only.

A password prompt appears.

3. Type the SYS password followed by **`' as sysdba'`**.
4. Issue one of the following commands:

   If the character set is UTF-8:

   ```
   sqlldr userid=sys control=iso3166-99.ctl
   log=<iso3166-logfilename>.log bad=<bad file name>.log
   ```

   If the character set is other than UTF-8:

   ```
   sqlldr userid=sys control=iso3166-99-ascii.ctl
   log=<iso3166-logfilename>.log bad=<bad file name>.log
   ```

#### 4.4.6.3 Verifying the Upgrade Results

All errors are captured in the log file you specified. The log file should contain no errors. The log file should contain the message:

```
SUCCESSFUL install/upgrade of schema: UDDISYS
```

#### 4.4.6.4 Resolving Errors

The following error message may appear in the log file:

**ORA-01435: user does not exist**

**Cause:** The database has not been used by Oracle UDDI.

**Action:** Perform a new installation of Oracle UDDI instead of an upgrade.

**ORA-00942: table or view does not exist**

**Cause:** The connected database user does not have privileges to access UDDI schema.

**Action:** Run the script again with SYS user privileges.

#### 4.4.6.5 UDDI Registry Schema Post-Upgrade Task

Optionally, you can gather statistics on the UDDISYS database schema, if you had been gathering them before the upgrade.

#### 4.4.6.6 Validating the UDDI Registry Schema Upgrade

To verify the success of the Oracle Application Server Web Services UDDI schema upgrade:

1. Start the middle tier by following Step 3 in Section 4.3, "Performing the Metadata Repository Upgrade" on page 4-10.
2. In a browser, access the UDDI main page at:

   `http://<host>:<port>/uddi/`
3. Click the links in the page to verify the inquiry and publishing endpoints.

### 4.4.7 Executing the Web Clipping Schema Upgrade Script

Follow the steps below to execute the Web Clipping schema. Because Web Clipping is a new component, the upgrade consists of adding tables and the necessary functions.

1. Ensure that the ORACLE_HOME environment variable is set to *<Infra_OH>* and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
2. Change directories to *<repCA_CD>*`/repCA/wcs/admin`
3. Connect to SQL*Plus with the command:

   **`sqlplus "sys as sysdba"`**
4. Change the current schema to WCRSYS with the command:

   ***`ALTER SESSION SET CURRENT_SCHEMA = WCRSYS;`***
5. Start the script with the command:

   **`@wcpinst.sql`**

#### 4.4.7.1 Verifying the Upgrade Results

Connect to SQL*Plus with SYS as SYSDBA and issue this command:

```
SELECT * from WCRSYS.WWWCP_OBFUSCATION$;
```

If the seeded key is present, then the script completed successfully.

A detailed description of the processing performed by this script is included in Section A.2.9, "The Web Clipping Upgrade Process" on page A-25.

### 4.4.8 Upgrading the Oracle Application Server Wireless Schema

The Oracle Application Server Wireless schema in the metadata repository is automatically upgraded during installation of the first Oracle Application Server Portal and Wireless or Business Intelligence and Forms middle tier, so there are no Infrastructure upgrade tasks for Oracle Application Server Wireless.

## 4.5 Upgrading the OracleAS Portal Repository

This section explains the steps involved in upgrading the OracleAS Portal Repository. It discusses the following topics:



### 4.5.1 Upgrading the Oracle9*i*AS Single Sign-On Server

Omit this step if the Oracle Application Server Single Sign-On server has already been upgraded to either 9.0.2.5 or 9.0.4. Execute this step only if you intend to run OracleAS Portal 10*g* (9.0.4) with Oracle9*i*AS 9.0.2.x and the update has not already been performed.

1. Log in to Oracle MetaLink at:

   `http://metalink.oracle.com`

2. Locate Patch 2995671 for your operating system.

3. Follow the instructions in the patch Readme file and in the Oracle9*i*AS Single Sign-On Migration Guide (`docs/sso_patch_902.html` file). These documents are included in the patch. The patch is applied to the Oracle9*i*AS Single Sign-On Server schema in the Oracle9*i*AS Release 2 (9.0.2) Infrastructure.

### 4.5.2 Creating a Backup

Back up the database and middle tiers before and after each major step of the upgrade process. Verify you have backed up the Portal you are upgrading. The OracleAS Portal specific schemas (depending on how it was installed) are:

- Portal Schema (e.g. PORTAL)
- Demo Schema (e.g. PORTAL_DEMO)
- Portal Public Schema (e.g. PORTAL_PUBLIC)
- Portal JSP Access (e.g. PORTAL_APP)
- Application Components (Database Providers) are generated into their own schemas. Verify you have backed up all the affected databases.

There are several ways to do a backup. Consult an experienced Database Administrator or refer to the Oracle documentation for further information on the back up and recovery procedures of the database.

---

**Note:** You can use the export utility provided with your Oracle9i database to transfer the contents to a file. Refer to the *Oracle9i Database Utilities Guide* for more information.

---

---

**Note:** During the upgrade, all existing content on the Portal Builder page is completely removed, and replaced by new content. If the Portal Builder page was customized, then those customizations are lost. Before the upgrade, identify and record all customizations to the Builder Page so that they can be re-applied after the upgrade.

---

### 4.5.3 Downloading the OracleAS Portal Repository Upgrade Patch

1. Log into Oracle MetaLink at:

   `http://metalink.oracle.com`

2. Locate patch 2778342.
3. Download the ZIP file onto a temporary directory in your computer and extract its contents. The file contains the following:

- Disk1 - a directory containing the scripts used to run the Portal middle tier updates and the repository upgrade
- `readme.html` - A readme file

4. Examine the `readme.html` file for any changes that could affect the remaining procedures.

### 4.5.4 Using Oracle Universal Installer to Copy OracleAS Portal Upgrade Scripts to the Middle Tier Oracle Home

1. Log in to the computer on which OracleAS 10*g* (9.0.4) middle tier is installed.
2. Set the ORACLE_HOME environment variable to <*destination_MT_OH*> , which is a 10*g* (9.0.4) middle-tier home, upgraded as described in Chapter 3, "Upgrading the Middle Tier".
3. Verify that your DISPLAY environment variable is set correctly. The installer brings up the user interface that requires this setting.
4. Run the Oracle Universal Installer to copy the Portal upgrade scripts to the middle tier Oracle Home using this command:

   ***<destination_MT_OH>/oui/bin/runInstaller***

   The following steps will guide you through the sequence of screens:
5. Click **Next** on the **Welcome** screen to proceed to the **File Locations** screen.
6. In the source path field on the **File Locations** screen, specify the `products.jar` file that was included in the download. For example, if you unzipped the downloaded file into your `/tmp/upg` directory, you would specify `/tmp/upg/Disk1/stage/products.jar` as the file.

***Figure 4–2 Specify File Locations screen***


7. Verify the destination Oracle Home name and path.
8. Click **Next**.

   The **Summary** screen appears.
9. Click **Install**.

   The process of copying the files to the Oracle Home can be from 15 to 30 minutes long. For the first few minutes, there is no activity on the screen, but the progress bar will soon become active.

   The **End of Installation** screen appears.
10. Click **Exit**.

    The directory `portal/upg/plsql` is created in the Oracle home. It contains the code required to upgrade your repository.

### 4.5.5 Assess Whether External Application ID Remapping is Necessary

If you are consolidating version 9.0.2 Oracle9*i*AS Single Sign-On servers that contain external applications, the external application identifiers must be unique. Every 9.0.2 Oracle9*i*AS Single Sign-On server is assigned a unique identifier that is used to derive the long form of the external application identifier. Oracle9*i*AS Portal also contains references to these external application identifiers.

The Oracle9*i*AS Portal repository references to these external application identifiers are updated automatically when you upgrade Oracle9*i*AS Portal 9.0.2 to OracleAS Portal10*g* (9.0.4). As part of the Oracle9*i*AS Single Sign-On server upgrade, you were instructed to apply an Oracle9*i*AS Portal version-specific patch to the Oracle9*i*AS Portal schema to convert the external application identifier references to the long format. If the script has been run before the upgrade, the upgrade code detects that it has been run and will not try to repeat the execution.

There is an exception that arises if, after upgrading the Oracle9*i*AS Single Sign-On server to version 9.0.2.5, you then associate an Oracle9*i*AS Portal 9.0.2 instance with the server that was not previously associcated with it, and the portal uses an External Applications portlet. In this case, you must omit the remapping of the identifiers when the portal is upgraded, since it is already storing the long external application identifiers in its customizations.

---

**Note:** To ensure a successful upgrade of External Application portlet customizations, carefully evaluate whether you need to run the `secfappno.sql` script in the Oracle9*i*AS Portal schema. If the incorrect External Application ID remapping script was run, or a required script was not run, the External Application customizations may not correspond to the External Application IDs in use by the associated Oracle9*i*AS Single Sign-On server.

If this happens, and errors occur when External Applications are accessed, users must recustomize the list of applications by clicking on the Customize link on the External Applications portlet. This will synchronize the application IDs with the IDs in use by Oracle9*i*AS Single Sign-On server.

---

To omit the remapping, run the script `<destination_MT_OH>/portal/upg/plsql/upg/9025-9026/wwc/upfappno.sql` in the Oracle9*i*AS Portal schema before you start the upgrade. The text below appears in the upgrade log file when an Oracle9*i*AS Portal schema is upgraded after this script has been run:

```
The External Application IDs have already been converted prior to the upgrade.
The SSO ID used for this purpose was NO_CONVERSION. No conversion will be
performed now.
```

> **See Also:** Oracle9*i*AS Single Sign-On Migration Guide, included with the Oracle9*i*AS Single Sign-On Server patch referred to in Section 4.5.1, "Upgrading the Oracle9iAS Single Sign-On Server" on page 4-31.

### 4.5.6 Executing the OracleAS Portal Repository Upgrade Script

The Portal repository upgrade is performed by running the `upgrade.pl` script. Follow these steps to perform the upgrade:

1. Ensure that you have downloaded and installed the most recent available version of the OracleAS Portal Repository Upgrade Patch. (New versions may be released periodically.)
   a. Log in to Oracle MetaLink at:

      `http://metalink.oracle.com`

   b. Locate Patch 2778342.
   c. Note the date the patch was last updated.
   d. If the patch was updated after you executed the steps in Section 4.5.3, "Downloading the OracleAS Portal Repository Upgrade Patch" on page 4-32, then download the patch and repeat the steps in that section.
2. Ensure that the environment is set to refer to *<destination_MT_OH>*.
3. Stop all middle tier services in *<destination_MT_OH>*.
4. Change directories to the root of the Portal source code. This is in the middle tier Oracle Home under `portal/upg/plsql`.

   This directory was created in the previous steps when the installer was run and the upgrade patch contents were extracted.

**Note:** If the file `upgrade.pl` is not located in the `portal/upg/plsql` directory under your middle tier Oracle Home, verify that the steps for running the Oracle Universal Installer in Section 4.5.4, "Using Oracle Universal Installer to Copy OracleAS Portal Upgrade Scripts to the Middle Tier Oracle Home" on page 4-33 have been followed accurately.

5. Run the upgrade script in precheck mode until there are no errors found.

   `<destination_MT_OH>/perl/bin/perl upgrade.pl -precheck [-l <log file>] [-t <tmp directory>] [-ssoid <sso id>]`

6. Run the upgrade script in normal mode and designate the parameters:

   `<destination_MT_OH>/perl/bin/perl upgrade.pl [-l <log file>] [-t <tmp directory>] [-ssoid <sso id>]`

***Table 4–1 upgrade.pl Script Parameters***

| Parameters | Description |
|---|---|
| `-precheck` | If precheck is specified, then only the prechecks are done and the upgrade stops. In this mode, the upgrade is not immediately stopped if a precheck fails. Instead, the errors for all prechecks are consolidated in the upgrade log.<br>Review the log to see a list of checks that failed. Refer to the *Oracle Application Server Portal Error Messages Guide* for solutions to errors. Run this mode until all of the prechecks succeed.<br>In this mode, the schema is not altered, so restoring from the backup is not necessary between executions of the script. |
| `-l <log file>` | The log file name.<br>Default: `upgrade.log`. |
| `-t <tmp directory>` | The temporary directory name. It must be empty and writeable. In this document, it is also referred to as *<upgrade tmp dir>*.<br>Default: `tmp` |

***Table 4–1 upgrade.pl Script Parameters***

| Parameters | Description |
|---|---|
| `-ssoid <sso id>` | The identifier of the Single Sign-on server with which the Portal is associated before the Portal schema is upgraded. This is a 15 character value that must match the identifier used when patching the Single Sign-on server.<br><br>This value is only required if either the Single Sign-on server identifier can not be generated automatically, or if the Single Sign-on server was patched earlier, but the Single Sign-on server identifier that was used needs to be changed. Typically, you do not have to pass this value, as the system generates it automatically.<br><br>Refer to Section 4.5.1, "Upgrading the Oracle9iAS Single Sign-On Server" on page 4-31 for patch details. |

The following examples assume that you have the appropriate middle tier Oracle Home perl executable accessible from your path:

```
perl upgrade.pl -precheck -l precheck.log -t prechecktmp
perl upgrade.pl -l myupgrade.log -t upgtmp
perl upgrade.pl -l myupgrade.log
perl upgrade.pl -t upgtmp
perl upgrade.pl
perl upgrade.pl -l myupgrade.log -t upgtmp -ssoid B63875271239654
```

The script prompts for the system setup information. Your answers are echoed back for verification at the end of the script. If you discover that you have entered incorrect information before the end of the script, you can exit before any changes are committed by answering no (n) to the last question.

The following are the questions from the script. Default answers to the questions are given in brackets.

Have you backed up your database (y | n)? [y]:

7. If you have not backed up the database, answer no (n), then back up the database and restart the script. If you have, answer yes (y).

   The following prompt appears:

   ```
   Enter the name of the schema you would like to upgrade
   [portal]:
   ```

8. Enter the schema name, if different from the a standard Oracle9*i*AS 9.0.2 infrastructure installation schema name of `portal`.

The following prompt appears:

```
Enter the password of the schema you would like to upgrade
[portal]:
```

9. Change the default value entered for the schema name, if appropriate.

   If needed, use the Oracle Directory Manager to obtain the randomized Oracle9*i*AS Portal schema password. Navigate to:

   a. Entry Management
   b. cn=OracleContext
   c. cn=Products
   d. cn=IAS
   e. cn=Infrastructure Databases
   f. OrclReferenceName=Infrastructure Database (for example: iasdb.server.domain.com)
   g. OrclResourceName=PORTAL
   h. Click on the above entry.
   i. Look for the orclpasswordattribute value on the right panel. This is the schema password.

      **See Also:** *Oracle Application Server 10g Administrator's Guide*

   The following prompt appears:

   ```
   Enter the password for the SYS user of your database
   [change_on_install]:
   ```

   ```
   Enter the TNS connect string to connect to the database
   [orcl]:
   ```

10. Provide the TNS connect string (found in the `tnsnames.ora` file). Test your access to the database by using the Portal schema, schema password, and connect string.

    The following message appears:

    ```
    At this point, no changes have been made to the database.
    Please review all the details displayed above. If you
    choose to stop the upgrade at this point, you will be able
    to start it again without restoring from a backup. If you
    ```

```
continue and a problem occurs, you will have to retrieve
your backup and start over. Is it OK to continue with the
upgrade? (y|n) [y]:
```

11. Verify that the answers are correct, and answer the question:
    - Enter `y` to proceed with the upgrade.
    - Enter `n` to stop the upgrade. If necessary, you can execute the upgrade script again, correcting answers as needed.

---

**Important:** Refrain from using the computer for other work during the upgrade. It will take several hours to upgrade the OracleAS Portal repository. The length of the upgrade varies, depending on the size of the repository, the computer work load, and other factors.

---

## 4.5.7 Examining the Log File

Any errors that occurred during the upgrade are captured in the `upgrade.log` file. You should examine this file to ensure that the upgrade was successful.

1. Locate the log file. The default name for the log is `upgrade.log`. The default location is *<destination_MT_OH>*`/portal/upg/plsql/upgrade.log`.

   If the upgrade runs to completion, the errors in the log are sent to standard output, and are also included in a separate section at the end of the log file. Use the line numbers in the section at the end of the log file to search for the errors when they occurred earlier in the file.

2. Open the log with a text editor.
3. Research all errors and warnings in the *Oracle Application Server Portal Error Messages Guide*.
4. Resolve any errors and warnings that have documented actions. Most errors require that you restore the repository from backup, resolve the problem, and perform the upgrade again.

---

**Note:** Oracle Corporation does not support portals used after an upgrade that was not clean. A clean upgrade has either zero errors or only benign errors. Benign errors are identified in the *Oracle Application Server Portal Error Messages Guide*.

---

5. Contact Oracle Support for any errors that are not documented.
6. Continue this process until all errors are resolved.

   The following is an example of the end of the log file after a successful upgrade (note the `Upgrade completed successfully` message and the lack of error messages):

```
### PHASE III STEP 18: Show Errors and Warnings
Upgrade step started at Thu Jun 26 09:05:36 PDT 2003
...
### Upgrade completed successfully
Upgrade Ended at Thu Jun 26 09:05:36 PDT 2003
```

   The following are examples of the bottom portions of log files in cases of unsuccessful upgrades:

   Example 1: Premature termination and no error / warning section:

```
### ERROR: Patch Failed with status code: 1.
###
### Upgrade aborted at Tue Jul 15 15:09:33 EDT 2003.
```

   Example 2: Normal termination but at least one error found (notice the line numbers preceding each error line):

```
###
### PHASE III STEP 18: Show errors and warnings
###
Upgrade step started at Fri June 6 20:32:02 2003
###
### WARNING: ### Upgrade completed with the following warnings
###               ...
###
### ERROR: ### Upgrade completed with the following errors
###      2803:ERROR at line 1:
###      2804:ORA-01418: specified index does not exist
```

The following are locations and contents of various generated files. Note that *<upgrade_tmp_dir>* refers to the temporary directory that was used when running `upgrade.pl`. By default, this directory is named `tmp` and is a subdirectory of *<destination_MT_OH>*`/portal/upg/plsql`.

*Table 4–2 Generated Files*

| Name of file | Location of file | Contents and Purpose |
|---|---|---|
| `upgrade.in` | `<upgrade_tmp_directory>/upgrade.in` | Contains the inputs to the upgrade script (answers to the questions contained in Section 4.5.6, "Executing the OracleAS Portal Repository Upgrade Script" on page 4-36). Passwords are not filled in for security reasons.<br>This file can be copied to a different location so that it is not overwritten in subsequent upgrades. The passwords can be included, and the new file used as standard input on all subsequent upgrades. |
| `<upgrade_log>` | `<upgrade_directory>/upgrade.log` | Contains all the logged information, including errors, warnings, and details. Errors are summarized at the end of the log file. |
| `upgrade.err` | `<upgrade_tmp_dir>/upgrade.err` | Contains the summary of the error messages from the end of the `<upgrade_log>`. |
| `upgrade.wrn` | `<upgrade_tmp_dir>/upgrade.wrn` | Contains the summary of the warning messages from the end of the `<upgrade_log>`. |

## 4.5.8 Completing the OracleAS Portal Repository Upgrade

This section provides instructions for completing the OracleAS Portal upgrade, and summarizes the changes introduced in the upgraded product. It contains these topics:

- Section 4.5.8.2, "Reconfiguring the OracleAS Portal for the Oracle Internet Directory" on page 4-43
- Section 4.5.8.3, "Starting all Middle Tiers That Use The Upgraded Portal Instance" on page 4-54
- Section 4.5.8.4, "Rebuilding the Oracle Text (formerly InterMedia Text) Indexes" on page 4-54
- Section 4.5.8.5, "Turning On Session Store Look-Up in Oracle9iAS Portal 9.0.2.6 Upgrades" on page 4-55
- Section 4.5.8.6, "Understanding OracleAS Portal Usage Changes" on page 4-55
- Section 4.5.8.7, "Understanding Changes to the Portlet Repository" on page 4-60

#### 4.5.8.1 Compile All Packages

Due to dependencies, the OracleAS Portal repository upgrade may have caused objects in the database to become invalid. You can execute the `utlrp.sql` script to recompile all invalid objects in the database. The instructions are provided in Section 5.6.1, "Executing the utlrp.sql Utility" on page 5-63.

#### 4.5.8.2 Reconfiguring the OracleAS Portal for the Oracle Internet Directory

You must reconfigure the upgraded OracleAS Portal 10*g* (9.0.4) instance for the Oracle Internet Directory, using the following steps:

1. Set the ORACLE_HOME environment variable to `<destination_MT_OH>`.
2. Change directories to the `assistants/opca` directory under `<destination_MT_OH>`.
3. Execute the `ptlasst` script by issuing this command:

   **`ptlasst.csh -mode MIDTIER [-i <install type>] -type DIPUNREG [-s <portal schema>] [–sp <schema password>] [-c <connect string>] [-ldap_h <OID server host>] [-ldap_p <OID server port>] [-ldap_d <installer DN>] -ldap_w <installer password> [-ldaps]`**

   > **Note:** The `-mode MIDTIER and –type DIPUNREG` parameters remove the OracleAS Portal's existing provisioning profile from the Oracle Internet Directory server of the associated Infrastructure. This enables a new profile to be created with all the provisioning events required in OracleAS Portal10*g* (9.0.4).

   where:

   - `-i <install type>` can be set to `typical` or `custom`.

     When this parameter is set to `typical`, the repository access APIs are used to get the details of Oracle Internet Directory server and the OracleAS Portal schema in the configured OracleAS Metadata Repository.

     When it is set to `custom`, the input provided on the command line is used for the configuration. The default value of this parameter is `typical`.

   - `-s <portal schema>` is the OracleAS Portal schema name. The schema name "portal" is used if none is specified.

- `-sp <schema password>` is the OracleAS Portal schema password.
- `-c <connect string>` is the connect string to the OracleAS Portal database. The format is `HostName:PortNumber:ServiceName`.
- `-ldap_h <OID server host>` is the host name of the Oracle Internet Directory server.
- `-ldap_p <OID server port>` is the port number of the Oracle Internet Directory server.
- `-ldap_d <installer DN>` is the Distinguished Name of the user who is executing the script. If this parameter is not specified, then the default value of `cn=orcladmin` is used. If the Oracle Internet Directory server version is 9.0.2, then only the `cn=orcladmin` user has the required privileges. For Oracle Internet Directory in Oracle Application Server 10*g* (9.0.4), this user must be the super user `cn=orcladmin`, or the user must be a member of all of the following groups:

  ```
  cn=Trusted Applications Admins,
  cn=Groups,cn=OracleContext

  cn=iASAdmins, cn=Groups,cn=OracleContext

  cn=IAS & User Mgmt Application Admins,
  cn=Groups,cn=OracleContext
  ```

- `-ldap_w <installer password>` is the password of the user who is executing the script.
- `-ldaps` is the flag that indicates that Oracle Internet Directory is SSL-enabled. If this parameter is specified, then the Oracle Internet Directory server port specified using the `–ldap_p` parameter must be SSL-enabled.

Below is an example of this command:

```
ptlasst.csh -mode MIDTIER -i custom -type DIPUNREG -s
portal –sp password1 -c dbserver.domain.com:1521:iasdb
-ldap_h oidserver.domain.com -ldap_p 3060 -ldap_d
“cn=installer,cn=users,dc=mycompany,dc=com” -ldap_w secret1
```

4. After the script completes processing, examine the `<portal schema>.log` file in the current directory. All errors are captured in the `<portal schema>.log` file. If there are no errors, the script was successful. Common error messages and their resolution are described in Section 4.5.8.2.1, "Resolving ptlasst Script Errors for -type DIPUNREG" on page 4-49.

5. Execute the `ptlasst` script to reassociate the OracleAS Portal instance with the Oracle Internet Directory server by issuing this command:

   **`ptlasst.csh -mode MIDTIER [-i <install type>] -type OID [-s <portal schema>] [-sp <schema password>] [-c <connect string>] [-ldap_h <OID server host>] [-ldap_p <OID server port>] [-ldap_d <installer DN>] -ldap_w <installer password> [-pwd <initial password>] [-ldaps]`**

   where:

   - `-i <install type>` can be set to `typical` or `custom`.

     When this parameter is set to `typical`, the repository access APIs are used to get the details of Oracle Internet Directory server and the OracleAS Portal schema in the configured OracleAS Metadata Repository.

     When it is set to `custom`, the input provided on the command line is used for the configuration. The default value of this parameter is `typical`.

   - `-s <portal schema>` is the OracleAS Portal schema name. The schema name "portal" is used if none is specified.

   - `-sp <schema password>` is the OracleAS Portal schema password.

   - `-c <connect string>` is the connect string to the OracleAS Portal database. The format is `HostName:PortNumber:ServiceName`.

   - `-ldap_h <OID server host>` is the host name of the Oracle Internet Directory server.

   - `-ldap_p <OID server port>` is the port number of the Oracle Internet Directory server.

   - `-ldap_d <installer DN>` is the Distinguished Name of the user who is executing the script. If this parameter is not specified, then the default value of `cn=orcladmin` is used. If the Oracle Internet Directory server version is 9.0.2, then only the `cn=orcladmin` user has the required privileges. For Oracle Internet Directory in Oracle Application Server 10*g* (9.0.4), this user must be the super user `cn=orcladmin`, or the user must be a member of all of the following groups:

     ```
     cn=Trusted Applications Admins,
     cn=Groups,cn=OracleContext
     ```

     ```
     cn=iASAdmins, cn=Groups,cn=OracleContext
     ```

```
cn=IAS & User Mgmt Application Admins,
cn=Groups,cn=OracleContext
```

- `-ldap_w <installer password>` is the password of the user who is executing the script.
- `-pwd <initial password>` is the initial password for the OracleAS Portal seeded users (PORTAL and PORTAL_ADMIN) in Oracle Internet Directory. This password is used if the seeded user entries do not already exist in Oracle Internet Directory. If the user entries are already present in the Oracle Internet Directory server, then the existing passwords are not modified.
- `-ldaps` is the flag that indicates that Oracle Internet Directory is SSL-enabled. If this parameter is specified, then the Oracle Internet Directory server port specified using the `–ldap_p` parameter must be SSL-enabled.

Below is an example of this command:

```
ptlasst.csh -mode MIDTIER -i custom -type OID -s portal –sp
password1 -c dbserver.domain.com:1521:iasdb -ldap_h
oidserver.domain.com -ldap_p 3060 -ldap_d
“cn=installer,cn=users,dc=mycompany,dc=com” -ldap_w secret1
-pwd welcome1
```

6. After the script completes processing, examine the `<portal schema>.log` file in the current directory. If an error occurs, and the Oracle Internet Directory association fails, then the following error message is written into the log file:

   ```
   ERROR: creating lightweight users and groups in OID ...
   exiting
   ```

   Common error messages and their resolution are described in Section 4.5.8.2.2, "Resolving ptlasst Script Errors for -type OID" on page 4-50.

   If the OracleAS Portal is associated with an Oracle Application Server 10*g* (9.0.4) Oracle Internet Directory, server, then the command in Step 5 adds the application entry as a member of the group `“cn=Common Group Attributes,cn=Groups,cn=OracleContext”`.

7. If the OracleAS Portal is not using an Oracle Application Server 10*g* (9.0.4) Oracle Internet Directory server, then skip Steps a, b and c below and continue with Step 8.

   a. Locate the application name in the log file created by executing `ptlasst`. The file is typically named as `<portal schema>.log` and is located in the `<destination_MT_OH>/assistants/opca` directory. The

application name is present in the the log file following the string `Portal Application Name:`

**b.** Check the Oracle Internet Directory privileges of the application name by issuing the following command:

**`ldapsearch –h <OID host> -p <OID port> -D <Privileged user DN> -w <Privileged user password> [-U <SSL auth mode>] -b "cn=Groups,cn=OracleContext" –s sub "uniquemember=orclapplicationcommonname=<application name>,cn=Portal,cn=Products,cn=OracleContext" dn`**

where:

- *<OID host>* is the host name of the OID server.
- *<OID port>* is the port number of the OID server. If this port is SSL-enabled then the `–U` parameter must be specified.
- *<Privileged user DN>* is the Distinguished Name of the user who is running the script. This user must be the super user `cn=orcladmin`, or it must be a member of all of the following groups:

  ```
  cn=Trusted Applications Admins,
  cn=Groups,cn=OracleContext
  ```

  ```
  cn=iASAdmins, cn=Groups,cn=OracleContext
  ```

  ```
  cn=IAS & User Mgmt Application Admins,
  cn=Groups,cn=OracleContext
  ```

- *<Privileged user password>* is the password of the user who is running the script.
- *<SSL auth mode>* is the SSL authentication mode. This parameter should be provided only if the port specified through the `–p parameter` is SSL-enabled. Valid values for this parameter are: 1 = no authentication required, 2 = one-way authentication required, 3 = two-way authentication required

  Below is an example of this command:

  ```
  ldapsearch –h oidserver.domain.com -p 3060 -D
  "cn=installer,cn=users,dc=mycompany,dc=com" -w secret1
  -b "cn=Groups,cn=OracleContext" –s sub
  "uniquemember=orclapplicationcommonname=portal.
  030908.1810,cn=Portal,cn=Products,cn=OracleContext" dn
  ```

c. Verify that the group "cn=Common Group Attributes,cn=Groups,cn=OracleContext" appears in the list returned.

8. Examine the log file created by executing ptlasst. The file is typically named as <*portal schema*>.log and is located in the <*destination_MT_OH*>/assistants/opca directory. Verify that the provisioning profile was created successfully; look for a message similar to the following:

```
orclODIPProfileName=C815C7285F9E241CE0340800208A8B00_
C817B7D771034368E0340800208A8B00, cn=Provisioning Profiles,
cn=Changelog Subscriber, cn=Oracle Internet Directory

.

.

.

The Provisioning Profile for the Application has been
created.

Profile Name: c815c7285f9e241ce0340800208a8b00_
c817b7d771034368e0340800208a8b00
```

If the profile creation was unsuccessful, a message similar to the following is generated:

```
ERROR: The Provisioning Profile for the Application could
not be created.
```

This error can occur if a provisioning profile already exists in the Oracle Internet Directory server.

9. Verify whether the provisioning profile already exists by performing the following steps:

a. Log in to the OracleAS Portal with a user account that is a member of the PORTAL_ADMINISTRATORS or the DBA group in the PORTAL.

b. Click the **Administer** tab.

c. Click the **Global Settings** link on the **Services** Portlet.

d. Click the **SSO/OID** tab.

e. Ensure that the **Directory Synchronization** section is displayed at the bottom of the page and the checkbox for **Enable directory synchronization** is checked. If the checkbox is checked, then no further action is necessary.

**f.** If the checkbox is not checked, diagnose the error on the Oracle Internet Directory server.

**4.5.8.2.1 Resolving ptlasst Script Errors for -type DIPUNREG** The following error messages may appear in the log file after you execute the script:

**Error: Could not connect to OID. Please check ldap_host and ldap_port parameters.**

**Cause:** The OID server host or port is incorrect or is unreachable.

**Action:** Ensure that you have specified the correct values for the `–ldap_h` and `–ldap_p` parameters. Also ensure that the OID server is running and is reachable from the middle tier environment.

**Error: Invalid directory credentials. Please check ldap_user_dn and ldap_user_password parameters.**

**Cause:** The user DN or password is incorrect.

**Action:** Ensure that the user exists in the OID server and the values of the `–ldap_d` and `–ldap_w` parameters are correct.

**Error: Could not delete Provisioning Profile.**

**Cause:** The specified user does not have the required privileges.

**Action:** Ensure that you have either specified cn=orcladmin for the –ldap_d parameter. If you have a Oracle Application Server 10*g* (9.0.4) Oracle Internet Directory server, then ensure that the user is a member of all of the following groups in the OID server:

```
cn=Trusted Applications Admins, cn=Groups,cn=OracleContext

cn=iASAdmins, cn=Groups,cn=OracleContext

cn=IAS & User Mgmt Application Admins,
cn=Groups,cn=OracleContext
```

**Error: Could not delete Provisioning Profile.**

**Cause:** The provisioning profile does not exist in the OID server.

**Action:** This error is benign. No action is needed.

**Error: Unable to get GUID of entry.**

**Cause:** The Distinguished Name of the application entry stored in the Portal schema is null. It is not possible to locate the application entry in the OID server without the Distinguished Name.

**Action:** Execute the `ptlasst` script to reassociate the OracleAS Portal instance with the Oracle Internet Directory, as specified in Step 5 of Section 4.5.8.2, "Reconfiguring the OracleAS Portal for the Oracle Internet Directory" on page 4-43. This procedure populates the Distinguished Name of the application entry in the Portal schema, and attempts to create the provisioning profile.

**4.5.8.2.2 Resolving ptlasst Script Errors for -type OID** The following error messages may appear in the log file after you execute the script:

**Error: Error in bind ... LDAP Error: 1024: Unknown Error Encountered**

**Cause:** The Oracle Internet Directory server host or port is incorrect or is unreachable.

**Action:** Ensure that you have specified the correct values for the `-ldap_h` and `-ldap_p` parameters. Also ensure that the Oracle Internet Directory server is running and is reachable from the middle tier environment.

**Error: Error in bind ... LDAP Error: 49: Invalid Credentials**

**Cause:** The user DN or password is incorrect.

**Action:** Ensure that the user exists in the Oracle Internet Directory server and the values of the `-ldap_d` and `-ldap_w` parameters are correct.

**Error: Error in bind... LDAP Error: 53: DSA is unwilling to perform**

**Cause:** The user is unable to bind to the Oracle Internet Directory server. This could happen for many reasons. One reason is that the user's account has been globally locked.

**Action:** Unlock the user account and rerun the script. If the problem persists, then diagnose the problem in the Oracle Internet Directory server.

**Error: Error in create_application_entry... LDAP Error: 50: Insufficient access: ERROR: creating application entry**

**Cause:** The specified user does not have the required privileges.

**Action:** Ensure that you have specified cn=orcladmin for the `-ldap_d` parameter, or, if you have an Oracle Application Server 10*g* (9.0.4) Oracle Internet Directory server, then ensure that the user is a member of all of the following groups in the OID server:

```
cn=Trusted Applications Admins, cn=Groups,cn=OracleContext
cn=iASAdmins, cn=Groups,cn=OracleContext
```

```
cn=IAS & User Mgmt Application Admins,
cn=Groups,cn=OracleContext
```

**Error: Error in create_user... LDAP Error: 53: DSA is unwilling to perform...ERROR: creating entry <username> user**

**Cause:** There may be a constraint violation in the Oracle Internet Directory server. This could happen if one of the attribute values does not conform to the validation rules defined in the Oracle Internet Directory.

**Action:** Ensure that all parameters have correct values. The password value passed by the `-pwd` parameter should conform to the password policy defined in the Oracle Internet Directory server.

**Error: The Provisioning Profile for the Application could not be created.**

**Cause:** Usually, this means that the provisioning profile is already present in the Oracle Internet Directory server.

**Action:** Remove the existing provisioning profile using `ptlasst -mode MIDTIER -type DIPUNREG` as described in Step 3 of Section 4.5.8.2, "Reconfiguring the OracleAS Portal for the Oracle Internet Directory" on page 4-43, and then execute `ptlasst -mode MIDTIER -type DIPREG` as described in Section 4.5.8.2.3, "Optional: Creating an OracleAS Portal Provisioning Profile in the Oracle Internet Directory" on page 4-51.

#### 4.5.8.2.3 Optional: Creating an OracleAS Portal Provisioning Profile in the Oracle Internet Directory

If the error described in Section 4.5.8.2.2, "Resolving ptlasst Script Errors for -type OID" on page 4-50, occurs, you should perform this procedure to create a provisioning profile for the upgraded OracleAS Portal 10*g* (9.0.4) instance.

1. Set the ORACLE_HOME environment variable to *<destination_MT_OH>*.
2. Change directories to the `assistants/opca` directory under *<destination_MT_OH>*.
3. Execute the `ptlasst` script by issuing this command:

```
ptlasst.csh -mode MIDTIER [-i <install type>] -type DIPREG
[-s <portal schema>] [–sp <schema password>]  [-c <connect
string>] [-ldap_h <OID server host>] [-ldap_p <OID server
port>] [-ldap_d <installer DN>] -ldap_w <installer
password> [-ldaps]
```

---

**Note:** The `-mode MIDTIER –type and DIPREG` parameters create a provisioning profile for the OracleAS Portal in the Oracle Internet Directory server of the associated Infrastructure.

---

where:

- `-i <install type>` can be set to `typical` or `custom`.

  When this parameter is set to `typical`, the repository access APIs are used to get the details of Oracle Internet Directory server and the OracleAS Portal schema in the configured OracleAS Metadata Repository.

  When it is set to `custom`, the input provided on the command line is used for the configuration. The default value of this parameter is `typical`.

- `-s <portal schema>` is the OracleAS Portal schema name. The schema name "portal" is used if none is specified.
- `-sp <schema password>` is the OracleAS Portal schema password.
- `-c <connect string>` is the connect string to the OracleAS Portal database. The format is `HostName:PortNumber:ServiceName`.
- `-ldap_h <OID server host>` is the host name of the Oracle Internet Directory server.
- `-ldap_p <OID server port>` is the port number of the Oracle Internet Directory server.
- `-ldap_d <installer DN>` is the Distinguished Name of the user who is executing the script. If this parameter is not specified, then the default value of `cn=orcladmin` is used. If the Oracle Internet Directory server version is 9.0.2, then only the `cn=orcladmin` user has the required privileges. For Oracle Internet Directory in Oracle Application Server 10*g* (9.0.4), this user must be the super user `cn=orcladmin`, or the user must be a member of all of the following groups:

  ```
  cn=Trusted Applications Admins,
  cn=Groups,cn=OracleContext
  ```

  ```
  cn=iASAdmins, cn=Groups,cn=OracleContext
  ```

  ```
  cn=IAS & User Mgmt Application Admins,
  cn=Groups,cn=OracleContext
  ```

- `-ldap_w <installer password>` is the password of the user who is executing the script.
- `-ldaps` is the flag that indicates that Oracle Internet Directory is SSL-enabled. If this parameter is specified, then the Oracle Internet Directory server port specified using the `–ldap_p` parameter must be SSL-enabled.

Below is an example of this command:

```
ptlasst.csh -mode MIDTIER -i custom -type DIPREG -s portal
–sp password1 -c dbserver.domain.com:1521:iasdb -ldap_h
oidserver.domain.com -ldap_p 3060 -ldap_d
“cn=installer,cn=users,dc=mycompany,dc=com” -ldap_w secret1
```

4. After the script completes processing, examine the `<portal schema>.log` file in the current directory. All errors are captured in the `<portal schema>.log` file. Verify that the provisioning profile was created successfully; look for a message similar to the following:

```
orclODIPProfileName=C815C7285F9E241CE0340800208A8B00_
C817B7D771034368E0340800208A8B00, cn=Provisioning Profiles,
cn=Changelog Subscriber, cn=Oracle Internet Directory

.

.

.

The Provisioning Profile for the Application has been
created.

Profile Name: c815c7285f9e241ce0340800208a8b00_
c817b7d771034368e0340800208a8b00
```

If the profile creation was unsuccessful, a message similar to the following is generated:

```
ERROR: The Provisioning Profile for the Application could
not be created.
```

This error can occur if a provisioning profile already exists in the Oracle Internet Directory server.

5. Verify whether the provisioning profile already exists by performing the following steps:

   a. Log in to the OracleAS Portal with a user account that is a member of the PORTAL_ADMINISTRATORS or the DBA group in the PORTAL.

**b.** Click the Administer tab.

**c.** Click the Global Settings link on the Services Portlet.

**d.** Click on the SSO / OID tab.

**e.** Ensure that the “Directory Synchronization" section is displayed at the bottom of the page and the checkbox for “Enable directory synchronization” is checked.If the checkbox is checked, then execute `ptlasst` using `-type MIDTIER` and `-mode DIPUNREG` as described in Section 4.5.8.2, "Reconfiguring the OracleAS Portal for the Oracle Internet Directory" on page 4-43, and then execute `ptlasst` again, using `-type MIDTIER` and `-mode DIPREG` as described in this section.

**f.** If the checkbox is not checked, diagnose the error on the Oracle Internet Directory server.

### 4.5.8.3 Starting all Middle Tiers That Use The Upgraded Portal Instance

After the script has executed successfully, start each middle tier that is using the upgraded Portal instance by performing these steps:

1. Start OPMN and the processes managed by it with this command:

   ***<destination_MT_OH>*/opmn/bin/opmnctl startall**

   Oracle Process Management and Notification and all of the processes it manages are started (i.e., Distributed Configuration Management, Oracle HTTP Server, OC4J instances, OracleAS Web Cache, Oracle Application Server Forms Services, and Oracle Application Server Reports Services).

2. Start the Oracle Enterprise Manager Application Server Control by issuing this command:

   ***<destination_MT_OH>*/bin/emctl start iasconsole**

### 4.5.8.4 Rebuilding the Oracle Text (formerly InterMedia Text) Indexes

This step is required since the upgrade script drops the Oracle Text indexes and does not re-create them for OracleAS Portal 10*g* (9.0.4). If you want to use the Oracle Text features in the upgraded instance, you must re-create the indexes with the provided script. Run the `<destination_MT_OH>/portal/upg/plsql/wws/ctxcrind.sql` script using SQL*Plus from the Portal schema. If necessary, this script is rerunnable.

This process is the same for creating the Text indexes in any Portal build. For more information, refer to the *Oracle Application Server Portal Configuration Guide*.

**Note:** This process can take several hours, depending on the size of the repository and the speed of the computer.

#### 4.5.8.5 Turning On Session Store Look-Up in Oracle9*i*AS Portal 9.0.2.6 Upgrades

If you are upgrading from an Oracle9*i*AS Portal of version 9.0.2.6 to which a fix for page portlet sub-page items not displaying was applied (Bug 2904052), you must re-run a script with the parameter set to TRUE to restore the fix after the upgrade.

1. Connect to the OracleAS Portal schema SQL*Plus as the schema owner.
2. Execute the script:

   ```
   <destination_MT_OH>/portal/upg/plsql/wws/ptlinvsw.sql TRUE
   ```

#### 4.5.8.6 Understanding OracleAS Portal Usage Changes

This section details the differences between OracleAS Portal 10*g* (9.0.4) and Oracle9iAS Portal 9.0.2, and discusses issues you may encounter after upgrade.

**4.5.8.6.1 Builder Page is Re-created** During the upgrade, all existing content on the Portal Builder page is completely removed, and replaced by new content. If the Portal Builder page was customized, then those customizations are lost. Before the upgrade, identify and record all customizations to the Builder Page so that they can be re-applied after the upgrade.

**4.5.8.6.2 Category and Perspective Templates** In Oracle9iAS Portal 9.0.2 each category and perspective had a page with two tabs. Each tab had an autoquery portlet on it, one tab displayed the items belonging to that category or perspective, and the other tab displayed the pages belonging to the category or perspective. Also, in Oracle9iAS Portal 9.0.2 users might have customized the category or perspective pages or the portlets on these pages.

In OracleAS Portal 10*g* (9.0.4), these pages are based on a category or perspective template which belongs to the page group. Therefore:

- Each page group that contains either a category or a perspective will on upgrade be populated with both a category template and a perspective template.
- All categories created in that page group after the upgrade will have category pages based on the category template. Similarly, perspectives created after the upgrade will have pages based on the perspective template.

- If a Oracle9iAS Portal 9.0.2 page group had neither categories nor perspectives, no templates will be created for this page group during the upgrade. Subsequently, if a new category is added to the page group, a category template will be created and the category page will be based on that template.
- Scripts are provided that can be run after the upgrade that will:
  - delete the Oracle9iAS Portal 9.0.2 category and perspective pages
  - create new category and perspective pages based on the template for the Oracle9iAS Portal 9.0.2 categories
- The scripts are located in `<destination_MT_OH>/portal/upg/plsql/upg/902-9025/wws` and are:

  `pstdefin.sql`

  `pstpgshw.sql`

  `pstundef.sql`

  `pstpgcre.sql`

  `pstprcpg.sql`

1. After upgrading from Oracle9iAS Portal 9.0.2 to OracleAS Portal 10*g* (9.0.4), you can delete existing category and perspective pages and re-create them based on the category and perspective templates that were created during the upgrade. You can apply this fix to a Oracle9iAS Portal 9.0.2 instance that has been upgraded to OracleAS Portal 10*g* (9.0.4). Copy all the above files into a temporary directory and make that directory the current working directory.
2. Connect to the Portal using SQL*Plus as the Portal schema user. For example:
   `sqlplus portal/portal@iasdb`
3. Configure the `pstdefin.sql` file with information about the page groups and whether you wish to re-create category pages only, perspective pages only or both. The descriptions of all the settings are in the `pstdefin.sql` file.
4. If necessary, use the `pstpgshw.sql` script to retrieve information from the Portal to configure the `pstdefin`.sql file.
5. Run the pstpgcre.sql script to apply the changes. For example:

   ```
   SQL> @pstpgcre.sql
   ```

**4.5.8.6.3 New Date Formats Used When Searching Portal Content** In OracleAS Portal 10*g* (9.0.4), the date format used when searching Portal content is DD-MON-YYYY, or the equivalent date format for other languages. The appropriate date format for

your language is displayed as hint text on the search form in which you enter dates. If you upgrade a Saved Searches portlet that used a different format, such as DD-MON-YY, no search results are returned in the ugpraded portlet. You must re-create the saved search with the correct date format.

**4.5.8.6.4 WebDAV Client Display Items** The following display items and characteristics are changed in OracleAS Portal 10*g* (9.0.4):

- Only the primary file associated with a Portal item is displayed (i.e., image files are not displayed).
- Only the current version of a file is shown.
- File and Folder names are case-sensitive, and the real name is displayed (not the internal name, as was the case in Oracle9iAS Portal 9.0.2, or the display name).
- Only the SHARED, CORPORATE and user-created page groups are shown.
- Versioning is exposed in OracleAS Portal 10*g* (9.0.4), but any upgraded checked out item will not appear to be locked (i.e. checked out from a WebDAV client).

**4.5.8.6.5 Sub-Page Display Items** In Oracle9iAS Portal 9.0.2, links to sub-pages could be displayed on a page by adding the Sub-Page Display item to the page. In OracleAS Portal 10*g* (9.0.4), the Sub-Page Display item has been replaced with the Sub-Page Links region.

If a 9.0.2 page included a Sub-Page Display item in its own region, after upgrade, the item will be replaced by a Sub-Page Links region. If the Sub-Page Display item appeared in a region that contained other items, the other items will be upgraded within that region, but the Sub-Page Display item cannot be upgraded to a Sub-Page Links region and will be deleted. A warning message will be recorded in the upgrade log stating that the Sub-Page Display item was not upgraded. In this case, the page designer will need to manually add the Sub-Page Links region to the upgraded page.

With 9.0.2 Sub-Page Display items, you could specify the sort order in which the sub-page links were listed. If there is a single Sub-Page Display item on a 9.0.2 page, after upgrade, the corresponding Sub-Page Links region will use the same order. However, if the 9.0.2 page included more than one Sub-Page Display item, and those items used different sort orders, there is no way to determine which sort order to use. Therefore, the links in the upgraded Sub-Page Links region will be sorted alphabetically.

**4.5.8.6.6 Links to Personal Pages** The location of the main list of personal pages has been moved so that it is now the root page of the Shared Objects page group. In Oracle9iAS Portal 9.0.2, this list was a sub-page of the Shared Objects page group.

Consequently, after upgrade, you will find that any Page Link items that previously pointed to this sub-page will no longer work. To fix this, you must edit the item to point to the new location.

Similarly, you will also need to edit any lists of objects that include the Personal Pages page to point to the new location.

**4.5.8.6.7 Granting the View Privilege to the PUBLIC User** In Oracle9*i*AS Portal 9.0.2, although it was not recommended, you could grant the View privilege to the PUBLIC user to make the page visible to users who were not logged on. In OracleAS Portal 10*g* (9.0.4), the user interface no longer allows you to do this. Instead, you must select the Display Page to Public Users check box. After upgrade, any pages with access control lists that assigned the View privilege to the PUBLIC user will have this privilege removed and the Display Page to Public Users check box selected instead.

**4.5.8.6.8 SAMPLEPAGEGROUP Page Name** The internal name of the Corporate Pages page group has been changed from SAMPLEPAGEGROUP to CORPORATEPAGES to more accurately reflect its content. If you have any links to this page group that use the internal name in the URL, then you will need to change them to reflect the name change.

**4.5.8.6.9 Transport Sets** During upgrade, all export/import objects (tables, packages, etc.) are dropped and re-created. Any data stored in the export/import tables is deleted. Consequently, all transport sets are removed during upgrade.

This ensures that users do not inadvertently try to import a transport set from a previous version after upgrade (export/import does not work across versions of the product).

In OracleAS Portal 10*g* (9.0.4), a new manifest screen has been added to the export and import screens. A manifest screen displays the list of objects in the transport set, provides granular control over the modes of import for the various objects found in the transport set, and indicates how the objects are related.

**See Also:** *Oracle Application Server Portal Configuration Guide*

**4.5.8.6.10 Providers Navigator** In Oracle9iAS Portal 9.0.2, all providers were in a single list in the Providers Navigator. If you had a lot of providers, it could be quite difficult to locate it within this list. In OracleAS Portal 10*g* (9.0.4), the Providers Navigator is divided into three areas: Locally Built Providers, Registered Providers, and Provider Groups. To find a provider, you must first drill down into the appropriate area.

**4.5.8.6.11 Preview Links in Portlet Repository** Some portlets in the Portlet Repository will no longer have a preview link, because:

- In Oracle9iAS Portal 9.0.2, if a portlet did not support the preview mode, the Portlet Repository still included a preview link next to that portlet. When the preview link was clicked, a screen displayed a message that no preview was available. In OracleAS Portal 10*g* (9.0.4), if a portlet does not support the preview mode, the Portlet Repository does not display a preview link next to that portlet.
- Some portlets that did support the preview mode in Oracle9iAS Portal 9.0.2 no longer support it in OracleAS Portal 10*g* (9.0.4). In this case, the preview link will no longer appear next to the portlet in the Portlet Repository.

Instead of preview links, you may see underscore characters in the portlet titles. These look very similar to hypertext links. To prevent these underscore characters from displaying you need to edit the style applied to the page or region and change the Font Decoration for the Display Name and Image Link style element to <None>.

**4.5.8.6.12 Help System URL** The help system has a new URL:

`http://host:port/portalHelp/en/`

If you have URL items pointing to the previous help URL, you must update those URLs to point to this new URL.

**4.5.8.6.13 Parameters and Events Enabled by Default** In OracleAS Portal 10*g* (9.0.4), this feature is enabled by default. Therefore, after upgrade, parameters and events are enabled for all page groups. To disable this feature, click the Properties link for the page group, then click the Configure tab. In the Parameters and Events section, click the Edit link and de-select the Enable Parameters and Events.

**4.5.8.6.14 Errors Viewing Upgraded Mobile Pages** Portlet Builder portlets, e.g., forms, reports, and charts (previously known as components) that are mobile capable may show the following error when added to a mobile page:

```
Error: The XML parser encountered an Error, and could not
complete the conversion for portlet id=9,41924, it returned
the following message: (1)
```

As a workaround, navigate to the portlet from the Portal Navigator, and edit the portlet. Go to the Display Options tab. At the bottom, under Mobile Display Options, make sure that at least one column is selected in the Columns to Display list. Click OK to update the portlet's attributes. The portlet will now operate correctly on the mobile page.

### 4.5.8.7 Understanding Changes to the Portlet Repository

In OracleAS Portal 10*g* (9.0.4), the titles of several seeded portlet repository pages are different from those in the previous release:

- **Administration** was changed to **Administration Portlets**
- **Portal Tools** was changed to **Portlet Builders**
- **Shared Instances** was changed to **Shared Portlets**
- **Content** was changed to **Portal Content Tools**
- **Oracle9iAS Portal Community** was changed to **Portal Community News**
- **User Content** was changed to **Published Portal Content**
- **New** was changed to **Portlet Staging Area**

***Figure 4–3 Oracle9iAS Release 2 (9.0.2) Portlet Repository***

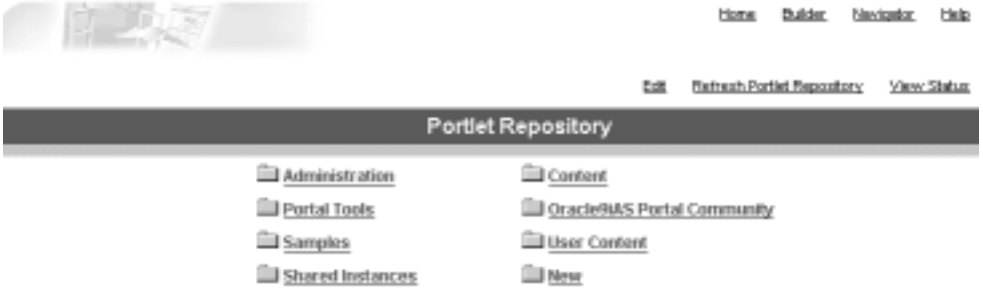


***Figure 4–4 Oracle Application Server 10g (9.0.4) Portlet Repository***


However, the pages are not automatically renamed when you upgrade from Oracle9iAS Portal 9.0.2. For example, if you had a page called Shared Instances in the portlet repository, it will not be renamed to Shared Portlets during the upgrade. To use the new titles, you must change the display names of the portlet repository pages.

**See Also:** *Oracle Application Server Portal User's Guide*

The upgrade creates pages in certain circumstances. The Portlet Staging Area page is created if the Oracle9iAS Portal does not have the New page. The Published Portal Content page is created if he Oracle9iAS Portal does not have the User Content page.

After upgrading, if you create a new page group, the provider page that represents the new page group is created under the Published Portal Content page, and will be visible to all users.

If you create or register a provider directly, the provider page will be created under the Portlet Staging Area page, and will be visible to the Portal Administrator and the user who created or registered the provider.

#### 4.5.8.7.1 Moving the Portlet Repository to the OracleAS Portal 10*g* (9.0.4) Format

By default, the portlet repository is upgraded in-place from Oracle9iAS Portal 9.0.2 to OracleAS Portal 10*g* (9.0.4). The existing pages, templates, items, etc. in the portlet repository are upgraded, and the new portlets are added into the repository.

Since the Release 2 (9.0.2) settings are preserved, the 10*g* (9.0.4) pages look very similar to the Release 2 (9.0.2) pages.

---

**Note:** If you had rendered the Portlet Repository in Release 2 (9.0.2) as grouped by Provider names, then after the upgrade the folders in the repository will be grouped by category, because the Group by Provider Name option has been deprecated in 10*g* (9.0.4). To create a similar organization, assign the portlet names to categories representing the Provider names.

---

If you want the repository to have the appearance of the OracleAS Portal 10*g* (9.0.4) installation, a script is available to re-create the upgraded portlet repository. The script removes the existing upgraded portlet repository and re-creates it. Use the script only if you do not wish to preserve customizations, settings, styles, banners, and so on in the portlet repository.

To re-create the repository, follow these steps:

1. Perform a backup of the database, since the script overwrites the repository and is not reversible.
2. Navigate to `<destination_MT_OH>/portal/upg/plsql/upg/309-903/wws/`.
3. Execute the `prrplc.pl` script by issuing this command:

   **`<destination_MT_OH>/perl/bin/perl prrplc.pl -t <portal tns name> [-s <portal schema>] [-w <portal schema password>]`**

   where:

   *<portal tns name>* is the TNS connect string for the database in which the OracleAS Portal repository resides.

   *<portal schema>* is the name of the Portal schema being modified.

   *<portal schema password>* is the password for the schema.

   For example:

   ```
   <destination_MT_OH>/perl/bin/perl prrplc.pl -t iasdb1 -s
   myportal  -w wh7HeT23
   ```

#### 4.5.8.8 Refreshing the OmniPortlet and Web Clipping Providers

The OmniPortlet and WebClipping providers are new in OracleAS Portal 10*g* (9.0.4). Consequently, these providers must be refreshed in the repository after the middle tier has been started. Repeat these steps for each OracleAS Portal instance that references these providers:

1. Log in to OracleAS Portal as an administrator.
2. Click the **Navigator** link.

   The **Portal Navigator** page appears.
3. Click the **Providers** tab.
4. Click **Locally Built Providers**.

   A sorted list of locally built providers appears.
5. Locate the OmniPortlet provider, using the **Next** and **Previous** links, if necessary.
6. Click the **Refresh** link for the OmniPortlet provider.
7. Locate the Web Clipping provider, using the **Next** and **Previous** links, if necessary.
8. Click the **Refresh** link for the Web Clipping provider.

### 4.5.9 Accessing the Upgraded OracleAS Portal

If there were no errors, you can access your upgraded Portal. Open a browser and access:

`http://<host>:<port>/pls/<dad>`

***Table 4–3 Access URL Parameters***

| Parameters | Description |
|---|---|
| *<host>* | The computer on which the OracleAS 10*g* (9.0.4) HTTP Server is running. |
| *<port>* | The port on which the Oracle9AS *10g* (9.0.4) HTTP Server is listening. |
| *<dad>* | The Portal DAD name of the upgraded Portal. This is the same DAD name you used before the upgrade. |

## 4.6 Upgrading Schemas in Customer Databases

Typically, schemas are contained in the Metadata Repository, but the schema for some components (for example, Oracle Application Server Discoverer and OracleAS Portal) might reside outside of the Infrastructure database. This section provides information on upgrading these components' schemas in customer databases.

**See Also:** *Oracle Application Server 10g Administrator's Guide*

### 4.6.1 Upgrading the Oracle Application Server Discoverer End User Layer Schema

If you have been using Oracle Application Server Discoverer version 9.0.2.52 or earlier, you must upgrade the End User Layer before you can use Oracle Application Server Discoverer 9.0.4 in 10*g* (9.0.4).

To upgrade the Oracle Application Server Discoverer End User Layer schema, you use Oracle Discoverer Administrator, shipped with the Oracle Developer Suite 10*g* (9.0.4).

**See Also:** *Oracle Discoverer Administrator Administration Guide in the Oracle Developer Suite documentation library.*

### 4.6.2 Upgrading the OracleAS Portal Schema

The procedure for upgrading the Portal schemas in the customer database is the same as that for upgrading the schemas in the Infrastructure, as long as the starting version is 9.0.2 or later. The steps are as follows:

1. Ensure that the OracleAS Portal repository database is a version that is supported by the upgrade process:

   9.0.1.5 is the minimum 9.0 version that is supported

   9.2.0.4 is the minimum 9.2 version that is supported

2. Follow the instructions provided in Section 4.5, "Upgrading the OracleAS Portal Repository" on page 4-31.

> **Note:** In any instructions that refer to the Infrastructure Oracle home or the Infrastructure database, use the Oracle home and database of the instance in which the OracleAS Portal schema resides.

General information on upgrading to OracleAS Portal 10*g* (9.0.4), as well as instructions for upgrading from other versions, is provided at:

`http://portalcenter.oracle.com/upgrades`

Select 9.0.4 from the list of upgrade targets on the page.

## 4.7 Activating 10*g* (9.0.4) Functionality for UDDI Applications

Because the Oracle Application Server upgrade was designed to support component operation with a Release 2 (9.0.2) Infrastructure, the UDDI sub-component of Oracle Application Server Web Services has a mode of operation for each version of the Infrastructure.

If the database schema in the repository was not upgraded, the UDDI sub-component will run in 9.0.2.3 compatibility mode after the upgrade.

To run in 10*g* (9.0.4) mode:

1. Upgrade the database schema in the metadata repository.
2. Restart the UDDI applications.

# 5

# Upgrading the Identity Management Services

This chapter explains how to upgrade Identity Management services. Before you perform the tasks in this chapter, you must perform the steps in Section 4.2, "Preparing to Upgrade the Metadata Repository" on page 4-3.

The chapter consists of the following sections:

# 5.1 Upgrading Identity Management

Identity Management comprises Oracle Application Server Single Sign-On and Oracle Internet Directory. This section describes possible configurations for Identity Management, and explains how to upgrade it using the Oracle Universal Installer. The following topics are included:

Section 5.1.1, "Identity Management Configuration Overview" on page 5-2

Section 5.1.2, "Understanding the Identity Management Upgrade Processes" on page 5-5

Section 5.1.3, "Using Oracle Universal Installer to Upgrade Identity Management" on page 5-7

## 5.1.1 Identity Management Configuration Overview

In Oracle9iAS Release 2 (9.0.2), a database tier is required to operate Oracle Application Server Single Sign-On and Oracle Internet Directory. The Metadata Repository contains the necessary schemas for these components.

An Oracle9iAS Release 2 (9.0.2) Identity Management configuration can be non-distributed, in which Oracle Application Server Single Sign-On and Oracle Internet Directory share a metadata repository. This is depicted in Figure 5–1. Alternatively, the Identity Management configuration can be distributed, in which Oracle Application Server Single Sign-On and Oracle Internet Directory each use a separate metadata repository. This is depicted inFigure 5–2.

In Oracle Application Server 10*g* (9.0.4), the distributed configuration is different from that in Release 2 (9.0.2), in that a single Metadata Repository is shared between Oracle Application Server Single Sign-On and Oracle Internet Directory, and Oracle Application Server Single Sign-On accesses it from a different computer. This is shown in Figure 5–3.

**Notes:** As shown in Figure 5–1, the non-distributed configuration in the 10*g* (9.0.4) release is similar to that in Oracle9*i*AS Release 2 (9.0.2)

If, in Oracle9iAS Release 2 (9.0.2), you had a Delegated Administration Services (DAS) or Directory Integration and Provisioning (DIP) operating in a middle tier, and you want to set up a DAS or DIP in 10*g* (9.0.4), you must perform a DAS-only or DIP-only installation in a separate Oracle home. See the section titled "Installing Identity Management Components Only" in the chapter "Installing OracleAS Infrastructure 10g" in the *Oracle Application Server 10g Installation Guide*.

***Figure 5–1 Non-Distributed Identity Management in Release 2 (9.0.2) and 10g (9.0.4)***

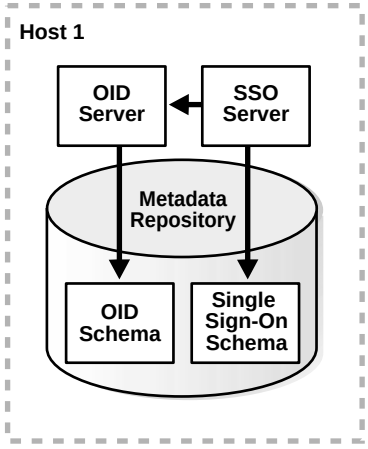


*Figure 5–2 Distributed Identity Management in Release 2 (9.0.2)*

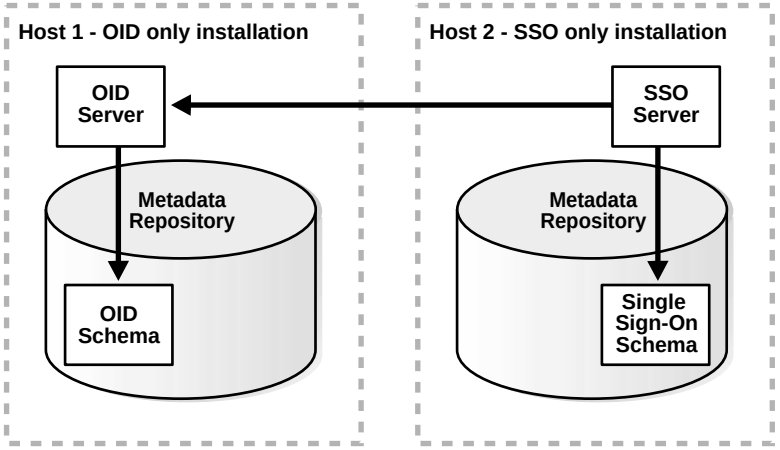
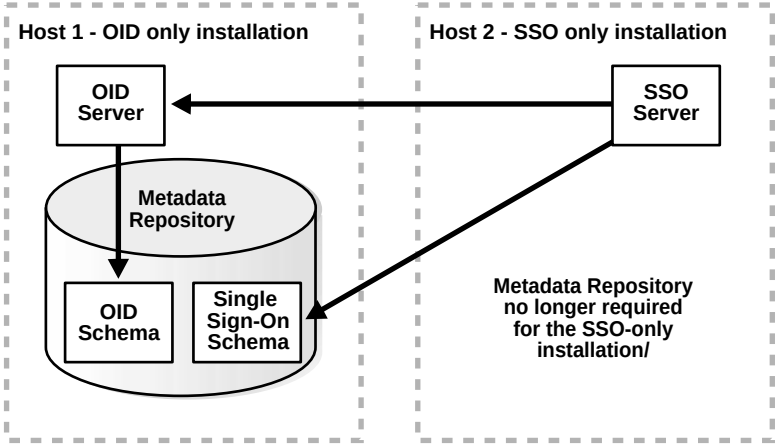



*Figure 5–3 Distributed Identity Management in 10g (9.0.4)*

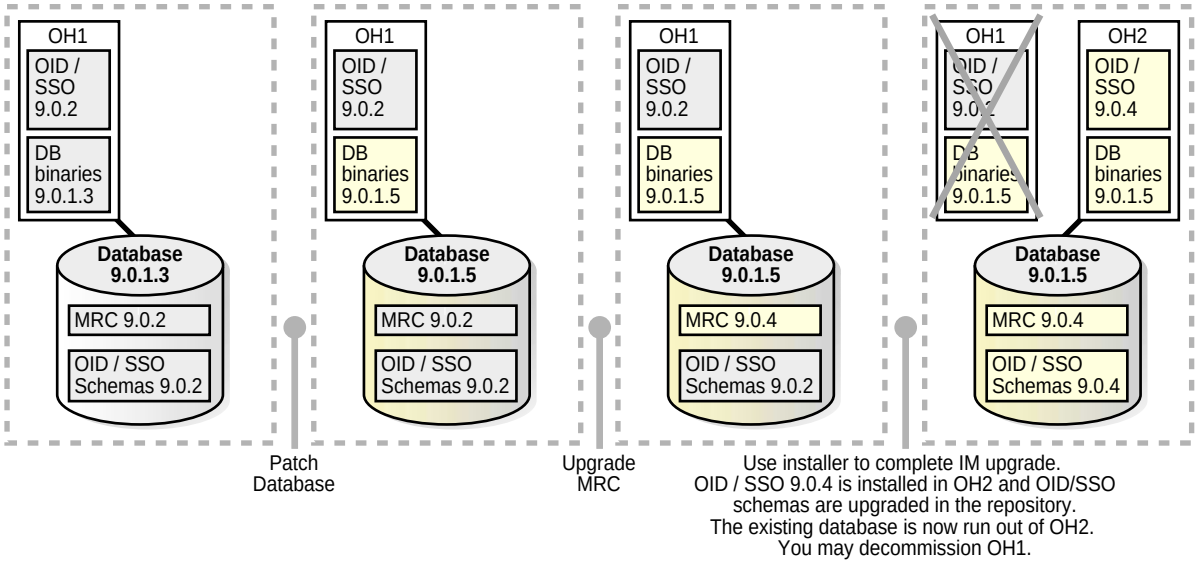


## 5.1.2 Understanding the Identity Management Upgrade Processes

The Identity Management schemas are contained in the Metadata Repository, along with other component schemas (such as those for OracleAS Portal and Oracle Ultra Search). However, the upgrade process for the Identity Management schemas (labeled OID / SSO in Figure 5–4) is different from the upgrade process for the component schemas (labeled MRC in Figure 5–4). The Identity Management schemas are upgraded by the Oracle Universal Installer, as shown in Figure 5–4, "Identity Management Upgrade". The component schemas are upgraded by individual scripts.

> **See Also:** Section 4.1, "Understanding the Metadata Repository Upgrade Process" on page 4-2

***Figure 5–4 Identity Management Upgrade***



#### 5.1.2.1 The Identity Management Upgrade Process

> **Note:** Before the Identity Management services can be upgraded, the 9.0.1.5 patch must be applied to the database.

The Identity Management upgrade consists of these steps:

1. The Metadata Repository Container Upgrade script is run.

   > **Note:** The Metadata Repository Container Upgrade script upgrades the Metadata Repository that is in use by the Identity Management services being upgraded. After this script is run, no new Oracle9*i*AS Release 2 (9.0.2) middle tier installations may use this Metdata Repository. However, existing Oracle9*i*AS Release 2 (9.0.2) middle tier installations will continue to function.

2. The Oracle Universal Installer is started; Oracle Internet Directory and Oracle Application Server Single Sign-On are installed in the new Oracle home and Oracle Internet Directory and Oracle Application Server Single Sign-On schemas are upgraded in the Metadata Repository.
3. All post-upgrade steps that are applicable to the upgraded configuration are performed, as described in Section 5.4, "Performing Infrastructure Post-Upgrade Tasks" on page 5-48.

   > **Note:** Do not manually delete any database (`*.dbf`) files that remain in the Oracle9*i*AS Release 2 (9.0.2) Infrastructure Oracle home (labeled OH1 in Figure 5–4) after Identity Management is upgraded to Oracle Application Server 10*g* (9.0.4). The Identity Management upgrade process does not copy or relocate any (`*.dbf`) files or redo log files to the destination Oracle home. If the (`*.dbf`) files were located in the source Oracle home before the Identity Management upgrade, they will remain there after the upgrade, unless you relocate them. For information on relocating the database files to the destination Oracle home, see Section 5.5, "Decommissioning the Release 2 (9.0.2) Oracle Home" on page 5-61.

## 5.1.3 Using Oracle Universal Installer to Upgrade Identity Management

The Identity Management upgrade is performed by Oracle Universal Installer. Oracle Universal Installer launches configuration assistants that upgrade the Oracle Internet Directory and Oracle Application Server Single Sign-On database schema. This upgrade can only be performed by a user with SYS credentials.

Before you start the Identity Management upgrade, ensure that:

- The steps in Section 4.2, "Preparing to Upgrade the Metadata Repository" on page 4-3 have been performed.
- The database server is running.
- The database listener is running.
- The Oracle Internet Directory server is running. To verify this, issue the following commands (each should return `"bind successful"`):

  ***<source_Infra_OH>*/bin/ldapbind -p <*Non-SSL port*>**

  ***<source_Infra_OH>*/bin/ldapbind -p <*SSL port*> -U 1**

### 5.1.3.1 Upgrading a Non-Distributed Identity Management Configuration

Follow these steps to upgrade a non-distributed Identity Management configuration (depicted in Figure 5–1, "Non-Distributed Identity Management in Release 2 (9.0.2) and 10g (9.0.4)"). Oracle Universal Installer will prompt you to stop and start certain components during the upgrade.

1. Log in to the computer on which Oracle9*i*AS Release 2 (9.0.2) is installed, as the same operating system user that performed the Oracle9*i*AS Release 2 (9.0.2) installation.
2. Mount the CD-ROM.

   **See Also:** *Oracle Application Server 10g Installation Guide*

3. Start the installer.

   **See Also:** *Oracle Application Server 10g Installation Guide*

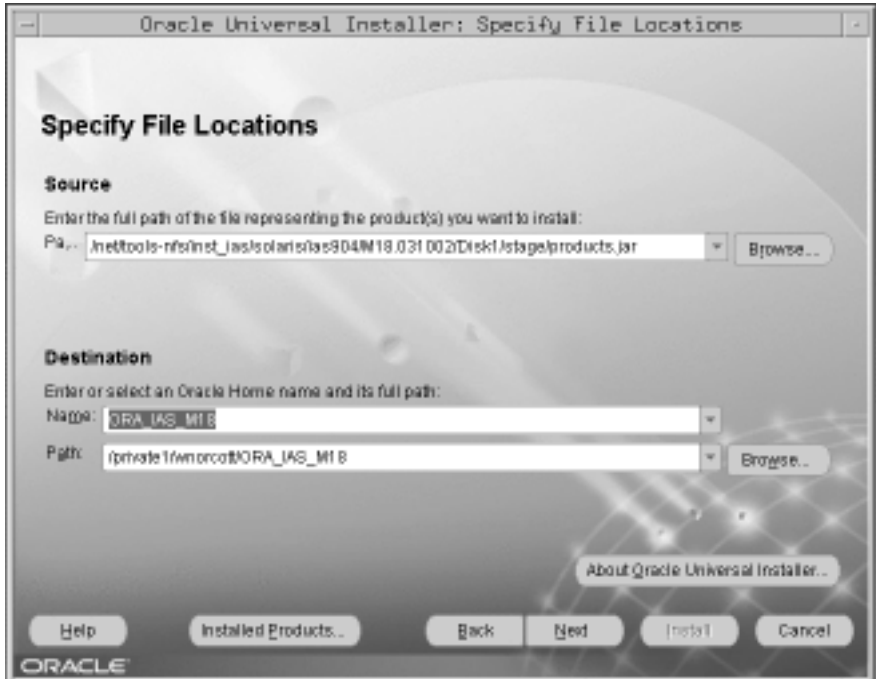
   The Welcome screen appears as shown in Figure 5–5.

***Figure 5–5 Welcome Screen***


**4.** Click Next.

The Specify File Locations screen appears as shown in Figure 5–6.

***Figure 5–6 Specify File Locations Screen***




**5.** Enter a new Oracle home name and a path for the 10*g* (9.0.4) upgrade and click Next.

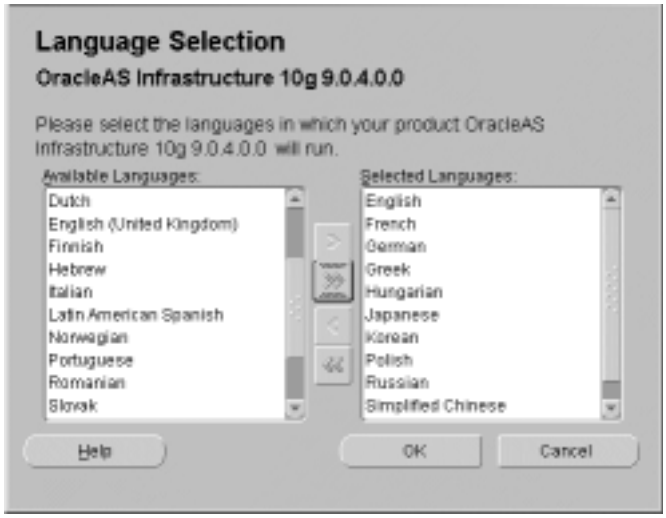
The Select a Product To Install screen appears as shown in Figure 5–7.

***Figure 5–7 Select a Product to Install Screen***


**6.** Select OracleAS Infrastructure 10*g*. If multiple languages are used in the Oracle9iAS Release 2 (9.0.2) Infrastructure, then click Product Languages. If you want only English to be installed in Oracle Application Server 10*g* (9.0.4), then click Next and continue with Step 8.

The Language Selection screen appears as shown in Figure 5–8.

***Figure 5–8 Language Selection Screen***




**7.** Select the languages you want to install and click OK.

> **Note:** If multiple languages were installed in Oracle9*i*AS Release 2 (9.0.2), select those languages. If you are not sure which languages were installed, but want languages other than English, click the double arrow button (>>) to select all languages.

The Select a Product to Install screen appears again.

**8.** Click Next.

The Select Installation Type screen appears as shown in Figure 5–9.

***Figure 5–9 Select Installation Type Screen***


**9.** Select Identity Management and OracleAS Metadata Repository and click Next.

The Upgrade Existing Infrastructure screen appears as shown in Figure 5–10.

***Figure 5–10 Upgrade Existing OracleAS Infrastructure Screen***


**10.** Select Upgrade Selected Oracle9*i*AS 9.0.2 Infrastructure.

**11.** Select the Infrastructure you want to upgrade from the drop-down list, then click Next. (If there is only one Infrastructure on the computer, then the drop-down list is inactive.)

The Specify Login for Oracle Internet Directory screen appears as shown in Figure Figure 5–11.

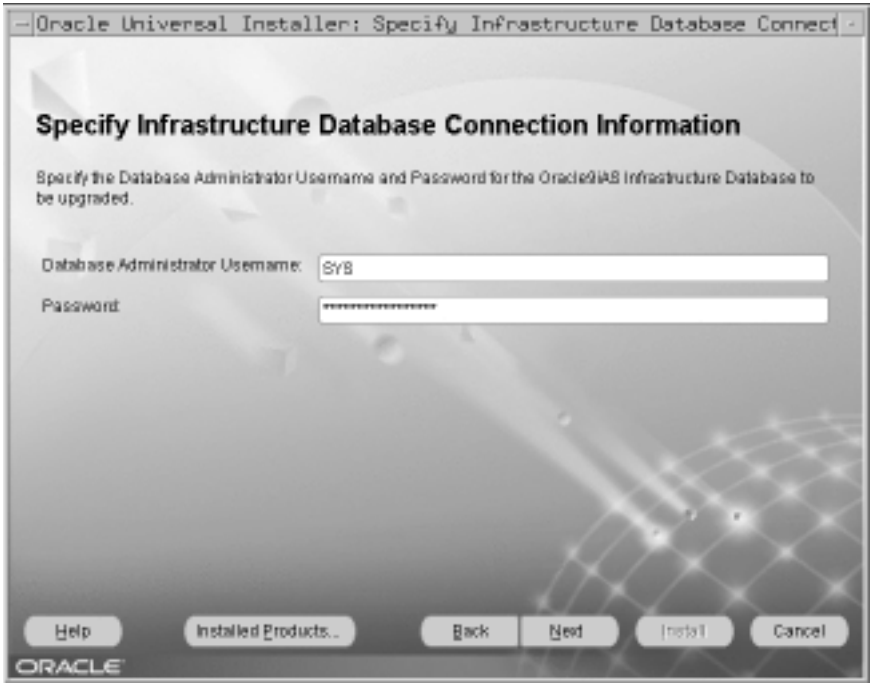
***Figure 5–11 Specify Login for Oracle Internet Directory Screen***


**12.** Enter the OID superuser DN in the Username field. The superuser DN cn=orcladmin is the default for this field; change this value if the OID superuser DN is not cn=orcladmin.

**13.** Enter the password in the Password field and click Next.

The Specify Infrastructure Database Connection screen appears as shown in Figure 5–27.

***Figure 5–12 Specify Infrastructure Database Connection Information Screen***




**14.** Enter SYS in the Username field and the SYS user's password in the Password field and click Next.

A warning dialog appears as shown in Figure 5–13, instructing you to stop processes in the Oracle home.

***Figure 5–13 Warning Dialog***

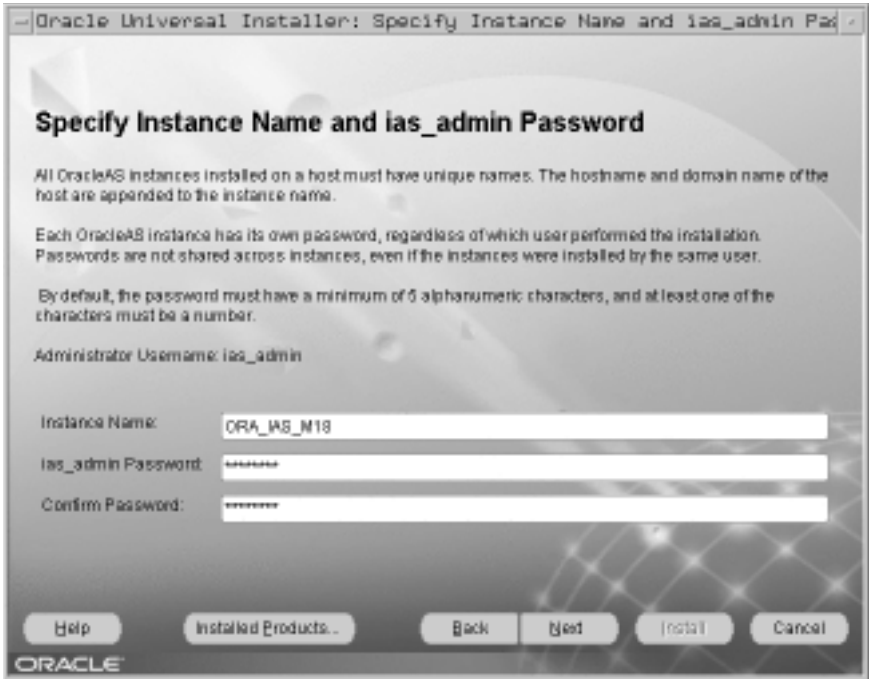

**15.** Stop Oracle Internet Directory and the Metadata Repository database listener.

**16.** Stop all processes in the Oracle home.

**17.** Ensure that the Metadata Repository database is running, then click OK.

The Specify Instance Name and ias_admin Password screen appears as shown in Figure 5–14.

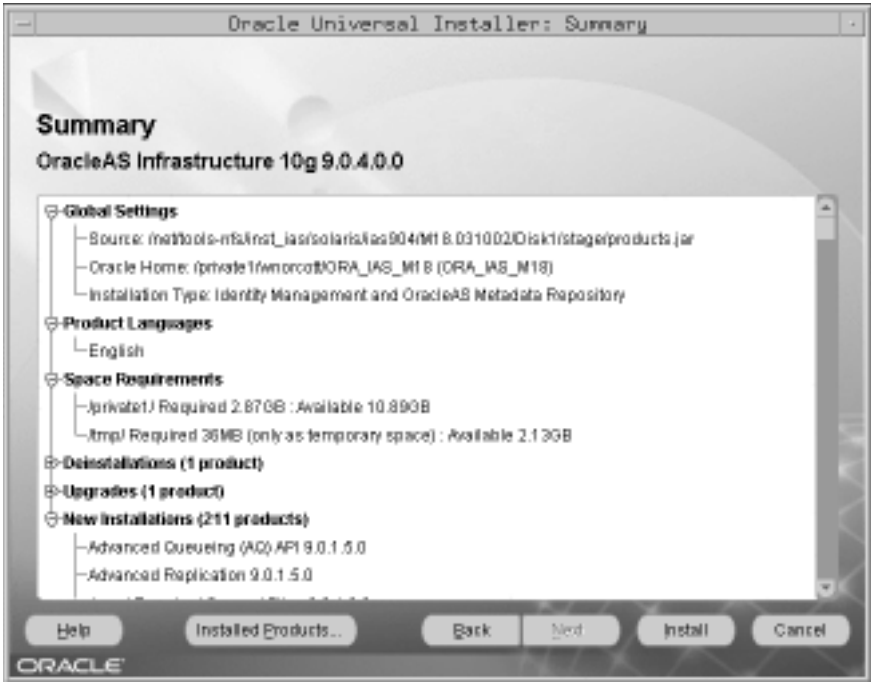
***Figure 5–14 Specify Instance Name and ias_admin Password Screen***




**18.** Complete the Instance Name, ias_admin Password, and Confirm Password fields and click Next.

The Summary screen appears as shown in Figure 5–15.

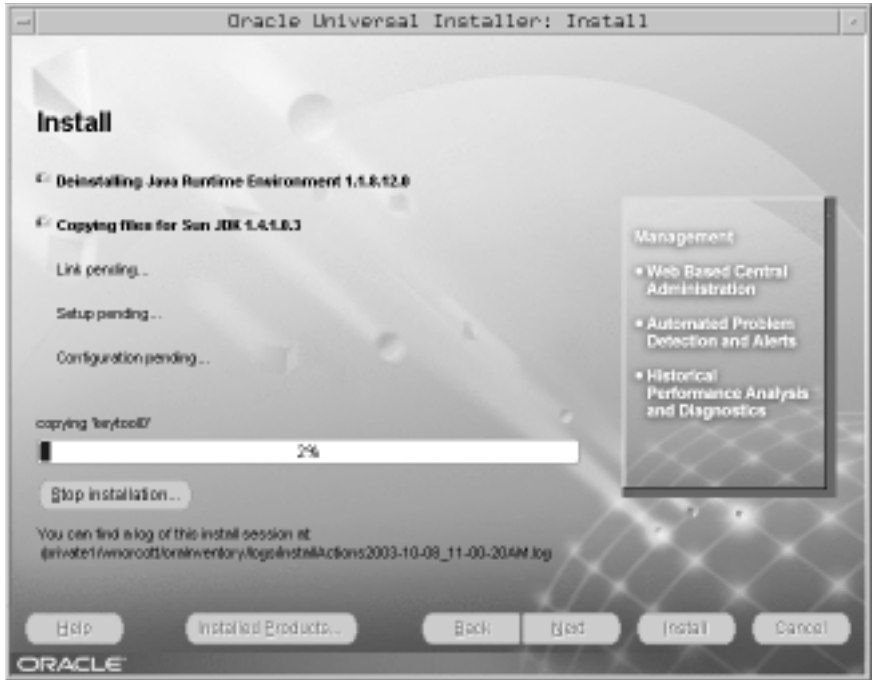
***Figure 5–15 Summary Screen***




**19.** Click Install.

The Install screen appears as shown in Figure 5–17, and the upgrade starts. The processing time varies, but it will be several minutes before you are prompted to take any action.

***Figure 5–16 Install Screen***



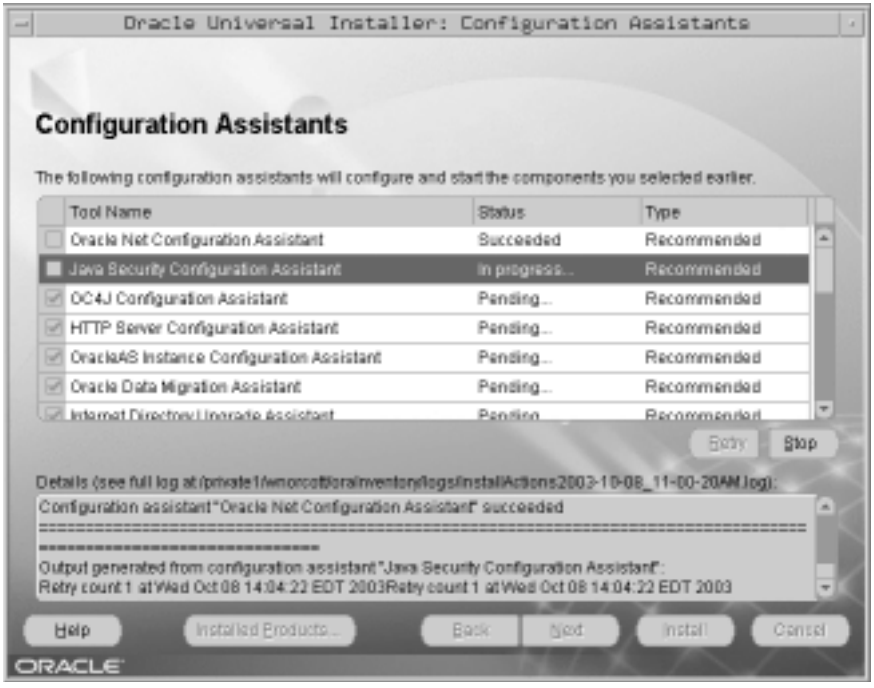

The Setup Privileges dialog appears as shown in Figure 5–17.

***Figure 5–17 Install Screen and Setup Privileges Dialog***


**20.** Open a window and run the script, then click OK in the dialog.

The script may take a few minutes to complete, depending on the speed and workload of the computer on which it is running. After the script completes, the Configuration Assistants screen appears as shown in Figure 5–18. The configuration process is lengthy.

***Figure 5–18 Configuration Assistants Screen***




**21.** Click Next.

After several minutes, the End of Installation screen appears as shown in Figure 5–18.

***Figure 5–19 End of Installation Screen***


**22.** Verify that Oracle Internet Directory and Oracle Application Server Single Sign-On are functioning and accessible.

> **See Also:** *Oracle Application Server 10g Administrator's Guide*, Chapter 1, "Accessing the Single Sign-On Server".

### 5.1.3.2 Upgrading a Distributed Identity Management Configuration

Follow the steps below to upgrade a distributed Identity Management configuration (depicted in Figure 5–2, "Distributed Identity Management in Release 2 (9.0.2)").This upgrade includes separate processes for Oracle Internet Directory and OracleAS Single Sign-On.

**5.1.3.2.1 Performing the Oracle Internet Directory Upgrade** Perform the steps in Section 5.1.3, "Using Oracle Universal Installer to Upgrade Identity Management" on page 5-7, and Section 5.1.3.1, "Upgrading a Non-Distributed Identity Management Configuration" on page 5-7.

After the upgrade, the Oracle Internet Directory server is running in the new Oracle home.

> **Note:** The Release 2 (9.0.2) installation of Oracle Application Server Single Sign-On is still functional after the Oracle Internet Directory upgrade. In general, however, the operation of middle tiers that are installed, upgraded, or re-configured to run with partially upgraded Identity Management Services is not supported.

**5.1.3.2.2 Performing the Oracle Application Server Single Sign-On Upgrade** Perform the steps below to upgrade the Oracle Application Server Single Sign-On server. Before you begin, ensure that:

- The Oracle Internet Directory upgrade is complete.
- You have credentials for the Oracle Application Server Single Sign-On database.
- You have credentials for the Oracle Internet Directory database.
- The Oracle Internet Directory database is running.

1. Log in to the computer on which Oracle9*i*AS Release 2 (9.0.2) Oracle Application Server Single Sign-On is installed.
2. Mount the CD-ROM.

   **See Also:** *Oracle Application Server 10g Installation Guide*

3. Start the installer.

   **See Also:** *Oracle Application Server 10g Installation Guide*

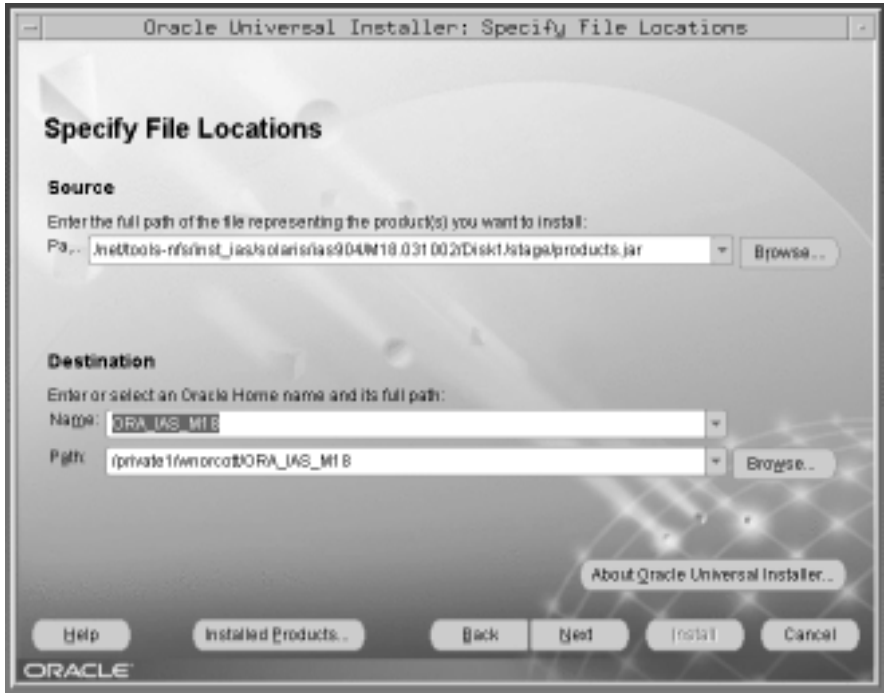
The Welcome screen appears as shown in Figure 5–20.

***Figure 5–20 Welcome Screen***


**4.** Click Next.

The Specify File Locations screen appears as shown in Figure 5–21.

***Figure 5–21 Specify File Locations Screen***




**5.** Enter a new Oracle home name and path for the 10*g* (9.0.4) upgrade and click Next.

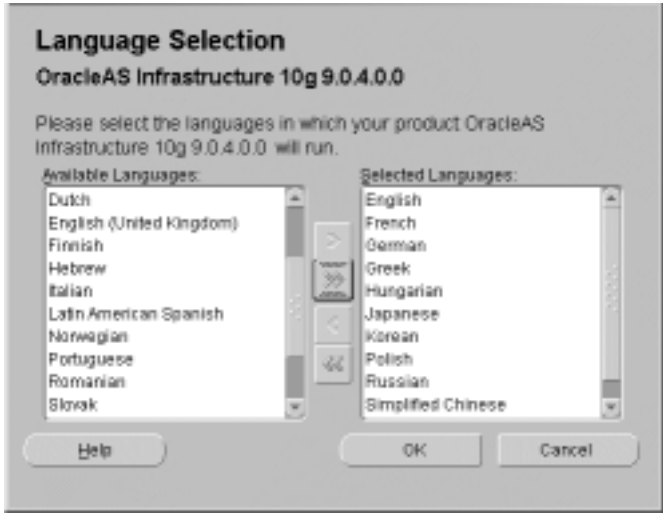
The Select a Product To Install screen appears as shown in Figure 5–22.

***Figure 5–22 Select a Product to Install Screen***


**6.** Select OracleAS Infrastructure 10*g*. If multiple languages are used in the Oracle9iAS Release 2 (9.0.2) Infrastructure, then click Product Languages. If you want only English to be installed in Oracle Application Server 10*g* (9.0.4), then click Next and continue with Step 8.

The Language Selection screen appears as shown in Figure 5–23.

***Figure 5–23 Language Selection Screen***




**7.** Select the languages you want and click OK.

> **Note:** If multiple languages were installed in Oracle9*i*AS Release 2 (9.0.2), select those languages. If you are not sure which languages were installed, but want languages other than English, click the double arrow button (>>) to select all languages.

The Select a Product To Install screen appears again.

**8.** Click Next.

The Select Installation Type screen appears as shown in Figure 5–24.

***Figure 5–24 Select Installation Type Screen***


**9.** Select Identity Management and OracleAS Metadata Repository and click Next.

The Upgrade Existing Infrastructure screen appears as shown in Figure 5–25.

***Figure 5–25 Upgrade Existing OracleAS Infrastructure Screen***


**10.** Ensure that the database listener in the Oracle9*i*AS Release 2 (9.0.2) Oracle Application Server Single Sign-On Oracle home is running.

**11.** Select Upgrade Selected Oracle9*i*AS 9.0.2 Infrastructure.

**12.** Select the Infrastructure you want to upgrade from the drop-down list, then click Next. (If there is only one Infrastructure, the drop-down list is inactive.)

The Specify Login for Oracle Internet Directory screen appears as shown in Figure Figure 5–26.

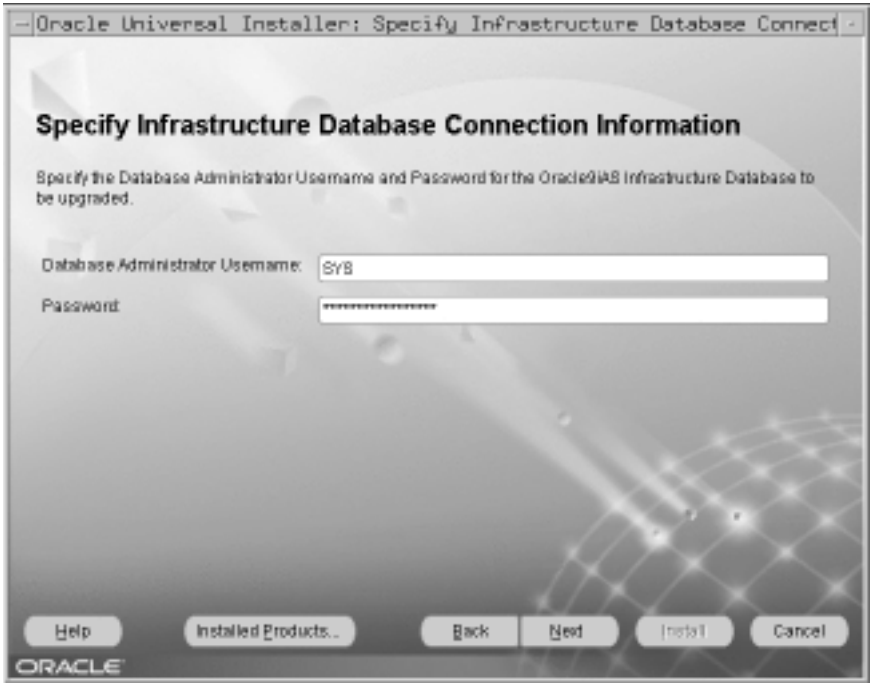
***Figure 5–26 Specify Login for Oracle Internet Directory Screen***


**13.** Enter the Oracle Internet Directory superuser DN in the Username field. The superuser DN cn=orcladmin is the default for this field; change this value if the DN is not cn=orcladmin.

**14.** Enter the password in the Password field and click Next.

The Specify Infrastructure Database Connection screen appears as shown in Figure 5–27.

***Figure 5–27 Specify Infrastructure Database Connection Information Screen***




**15.** Enter the Oracle Application Server Single Sign-On SYS user name in the Username field and the SYS user's password in the Password field and click Next. You are connecting to the Oracle Application Server Single Sign-On database.

The Specify OID Database Login screen appears as shown in Figure 5–28.

***Figure 5–28 Specify OID Database Login Screen***

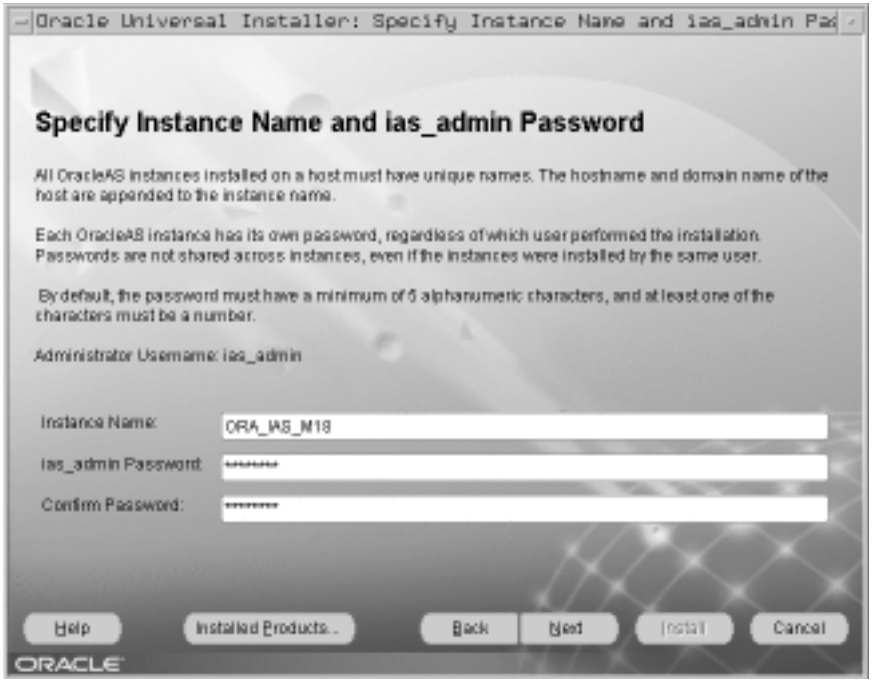

**16.** Enter the Oracle Internet Directory Database SYS user name in the Database Administrator Username field and the password in the password field, then click Next.

A warning dialog appears, instructing you to stop processes in the Oracle home.

**17.** Stop the Oracle HTTP Server and click OK.

The Specify Instance Name and ias_admin Password screen appears as shown in Figure 5–29.

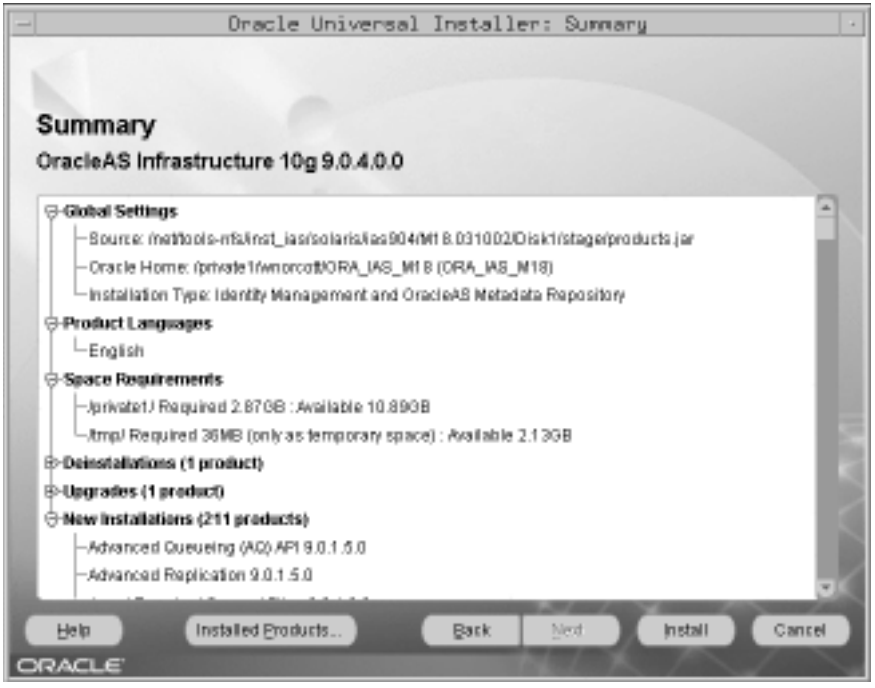
***Figure 5–29 Specify Instance Name and ias_admin Password Screen***




**18.** Complete the Instance Name, ias_admin Password, and Confirm Password fields and click Next.

The Summary screen appears as shown in Figure 5–30.

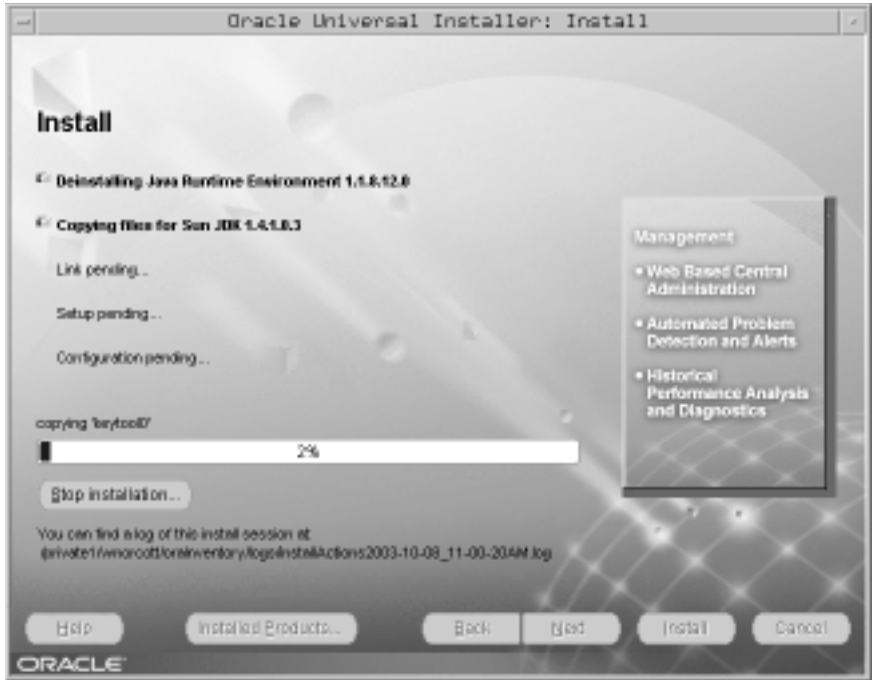
***Figure 5–30 Summary Screen***




**19.** Click Install.

The Install screen appears as shown in Figure 5–31, and the upgrade starts. The processing time varies, but it will be several minutes before you are prompted to take any action.

***Figure 5–31 Install Screen***



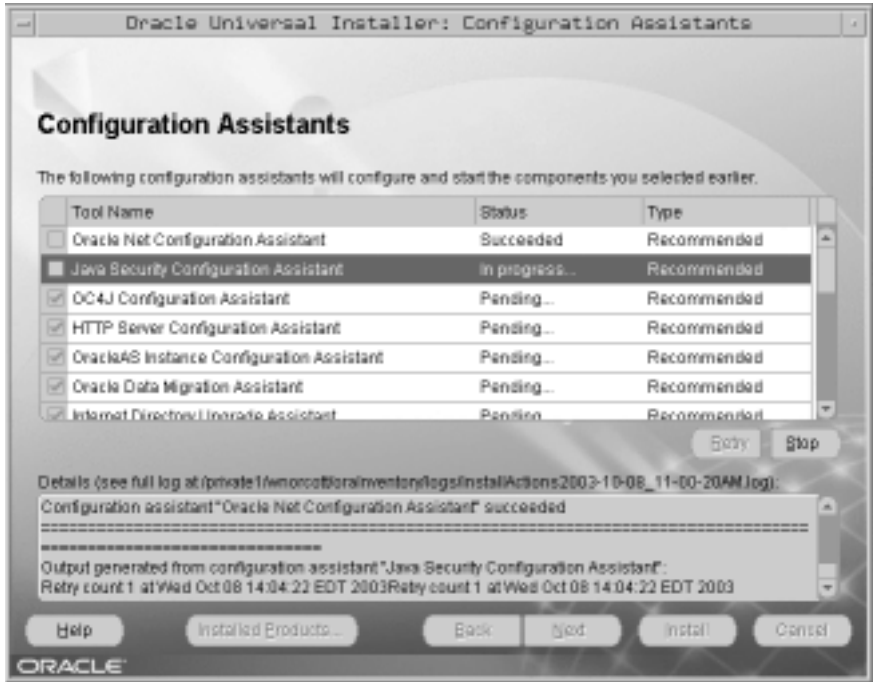

The Setup Privileges dialog appears as shown in Figure 5–32.

***Figure 5–32 Install Screen and Setup Privileges Dialog***


**20.** Open a window and run the script, then click OK in the dialog.

The script may take up to an hour to complete, depending on the speed and workload of the computer on which it is running. After the script completes, the Configuration Assistants screen appears as shown in Figure 5–33. The configuration process is lengthy.

***Figure 5–33 Configuration Assistants Screen***




**21.** Click Next.

The End of Installation screen appears as shown in Figure 5–34.

***Figure 5–34 End of Installation Screen***


---

**Note:** If the Delegated Administration Services was running in the Oracle9*i*AS Release 2 (9.0.2) Oracle Internet Directory or OracleAS Single Sign-On Oracle home, and you wish to configure it in the Oracle AS 10*g* (9.0.4) Oracle Internet Directory or OracleAS Single Sign-On Oracle home, you can do this using the Oracle Enterprise Manager Application Server Control. For instructions, see "Configuring Oracle Delegated Administration Services by Using Oracle Enterprise Manager Application Server Control" in the *Oracle Internet Directory Administrator's Guide*.

---

**22.** Verify that Oracle Application Server Single Sign-On is functioning and accessible.

**See Also:** *Oracle Application Server 10g Administrator's Guide*, Chapter 1, "Accessing the Single Sign-On Server".

## 5.2 Performing an Oracle Internet Directory Multi-Master Replication Upgrade

This section describes how to upgrade Oracle Internet Directory in a replicated environment. You can upgrade one computer at a time, or all of the computers at one time. Instructions are provided for each method in the following sub-sections:

Section 5.2.1, "Upgrading Oracle Internet Directory on One Replica" on page 5-39

Section 5.2.2, "Upgrading Oracle Internet Directory on Multiple Replicas Simultaneously" on page 5-42

Oracle Corporation recommends that during upgrade, in order to prevent conflicts, the replication environment be a Single Master (that is, only one replica is read/write and all others are read only).

### 5.2.1 Upgrading Oracle Internet Directory on One Replica

Upgrading one computer at a time makes Oracle Internet Directory available during the upgrade for additions, modifications, and searching.

Follow these steps to upgrade one replica at a time:

1. Identify and upgrade the Master Definition Site (MDS).

   **See Also:** *Oracle Internet Directory Administrator's Guide*, Chapter 25, Managing Directory Replication

2. Stop the replication server, the LDAP server, and `oidmon` on the replica to be upgraded.
3. Delete all Advanced Symmetric Replication (ASR) jobs on other replicas in the replicated environment by issuing the command:

   **`<source_Infra_OH>/ldap/admin/delasrjobs.sql`**

   All ASR jobs on other master sites that transfer changes to the MDS are deleted. This has the effect of taking the MDS out of the replication environment, so that no changes come to it, while other replicas continue to operate and replicate changes.
4. Stop the database and listener on the replica to be upgraded.
5. Start the Oracle Universal Installer.

   The database and Oracle Internet Directory are upgraded.

6. Start the database and the listener.

7. Test the connectivity to other replicas. The Net8 migration assistant might have modified `listener.ora` and `tnsnames.ora`, breaking connectivity. If connectivity is broken, identify the entries that were modified in the files, and restore the entries from the files in `<source_Infra_OH>/network/admin/` to the corresponding files:

   `<destination_Infra_OH>/network/admin/listener.ora`

   `<destination_Infra_OH>/network/admin/sqlnet.ora`

   See Section 3.8.5.3, "Upgrading the tnsnames.ora File" on page 3-60 for instructions and cautions on modifying the `tnsnames.ora` file.

8. Create jobs on each replica, after it is upgraded, by issuing the command:

   **`<destination_Infra_OH>/ldap/admin/remtool -asrrectify`**

   The jobs that were deleted in Step 3 are re-created. They will begin transferring the existing changes and new changes from other replicas to the upgraded replicas.

9. Perform the post-upgrade procedures.

   > **See Also:** Section 5.4.1, "Completing the Oracle Internet Directory Upgrade" on page 5-48.

10. After upgrading the Infrastructure to Oracle Application Server 10*g* (9.0.4), include the ORACLE_SID environment variable in the `<destination_Infra_OH>/opmn/conf/opmn.xml` file, as shown:

```
<?xml version = '1.0' encoding = 'UTF-8'?>
  <opmn xmlns="http://www.acme.com/ias-instance">
...
        </ias-component>
        <ias-component id="OID" status="enabled">
            <process-type id="OID" module-id="OID">
                <environment>
                                          <variable id="ORACLE_SID"
value="value_of_oracle_sid"/>
                                      </environment>
                <stop timeout="1800"/>
                <process-set id="OID" numprocs="1">
                                          <dependencies>
...
</opmn>
```

11. Ensure that the ORACLE_SID environment variable is set to the Oracle Internet Directory replica database.
12. Start the LDAP server and `oidmon` on the replica to be upgraded.
13. Change the password of the replication DN of the upgraded replica by issuing the following command:

    ***<destination_Infra_OH>/ldap/admin/remtool -presetpwd -v -bind <host>:<port>***

14. Start the replication server.
15. Upgrade each of the other master site replicas by performing Steps 2 through 11.
16. Upgrade the database replication table by performing the steps below:
    a. Stop the replication server on all replicas.
    b. Quiesce the replication environment by issuing this command on the MDS replica:

       ***<destination_Infra_OH>*/ldap/admin/remtool -suspendasr**

    c. Connect as `REPADMIN` (the database replication administrator) on the MDS replica and issue the following command:

       ```
       execute DBMS_REPCAT.ALTER_MASTER_REPOBJECT (sname=>
       'ODS', oname=> 'ASR_CHG_LOG', type=> 'TABLE', ddl_text=>
       'alter table ods.asr_chg_log modify target_dn varchar2
       (1024)')
       ```

    d. Execute the following SQL command repeatedly until the "no rows selected" message appears:

       ```
       SELECT * from dba_repcatlog WHERE request = 'ALTER_
       MASTER_REPOBJECT';
       ```

    e. Generate replication support for the ASR_CHG_LOG table by issuing the command:

       ```
       execute DBMS_REPCAT.GENERATE_REPLICATION_SUPPORT
       (sname=> 'ODS', oname=> 'ASR_CHG_LOG', type=> 'TABLE');
       ```

    f. Execute the following SQL command repeatedly until the "no rows selected" message appears:

```
SELECT * from dba_repcatlog WHERE request = 'ALTER_
MASTER_REPOBJECT';
```

g. Resume the database replication by issuing the following command:

```
<destination_Infra_OH>/ldap/admin/remtool -resumeasr
```

h. Start the replication server on all replicas.

---

**Note:** Changes made on the 10*g* (9.0.4) replica may not replicate on the prior version consumer replica. The changes that did not replicate are kept in the Human Intervention Queue for change replication, and will be applied successfully when the consumer replica is upgraded.

---

## 5.2.2 Upgrading Oracle Internet Directory on Multiple Replicas Simultaneously

Upgrading multiple replicas simultaneously ensures that the entire network is upgraded without a transient stage. The procedure is simpler than that for upgrading one replica at a time, but involves directory service downtime.

Follow these steps to upgrade multiple replicas simultaneously:

1. Stop the replication server, the LDAP server, and `oidmon` on all replicas in the Directory Replication Group.
2. Stop the database and listener on on all replicas in the DRG.
3. Start the Oracle Universal Installer.

   The database and Oracle Internet Directory are upgraded.
4. Start the database and the listener on all replicas.
5. Test the connectivity to other replicas. The Net8 migration assistant might have modified `listener.ora` and `tnsnames.ora`, breaking connectivity. If connectivity is broken, identify the entries that were modified in the files, and restore the entries from the files in `<source_Infra_OH>/network/admin/` to the corresponding files:

   `<destination_Infra_OH>/network/admin/listener.ora`

   `<destination_Infra_OH>/network/admin/sqlnet.ora`

   See Section 3.8.5.3, "Upgrading the tnsnames.ora File" on page 3-60 for instructions and cautions on modifying the `tnsnames.ora` file.

6. Perform the post-upgrade procedures.

   **See Also:** Section 5.4.1, "Completing the Oracle Internet Directory Upgrade" on page 5-48.

7. Upgrade the database replication table by performing the steps below:

   a. Stop the replication server on all replicas.

   b. Quiesce the replication environment by issuing this command on the MDS replica:

      ***<destination_Infra_OH>*/ldap/admin/remtool -suspendasr**

   c. Connect as REPADMIN (database replication administrator) on the MDS replica and issue the following command:

      ```
      execute DBMS_REPCAT.ALTER_MASTER_REPOBJECT (sname=>
      'ODS', oname=> 'ASR_CHG_LOG', type=> 'TABLE', ddl_text=>
      'alter table ods.asr_chg_log modify target_dn varchar2
      (1024)')
      ```

   d. Execute the following SQL command repeatedly until the "no rows selected" message appears:

      ```
      SELECT * from dba_repcatlog WHERE request = 'ALTER_
      MASTER_REPOBJECT';
      ```

   e. Generate replication support for the ASR_CHG_LOG table by issuing the command:

      ```
      execute DBMS_REPCAT.GENERATE_REPLICATION_SUPPORT
      (sname=> 'ODS', oname=> 'ASR_CHG_LOG', type=> 'TABLE');
      ```

   f. Execute the following SQL command repeatedly until the "no rows selected" message appears:

      ```
      SELECT * from dba_repcatlog WHERE request = 'ALTER_
      MASTER_REPOBJECT';
      ```

   g. Resume the database replication by issuing the following command:

      ***<destination_Infra_OH>*/ldap/admin/remtool -resumeasr**

   h. Verify that the replication environment is set up correctly by issuing the following command:

**<*destination_Infra_OH*>/ldap/admin/remtool -asrverify [-v -conn @<*repadmin*>/<*password*>@<*connect string for the mds replica*>]**

i. After upgrading the Infrastructure to Oracle Application Server 10*g* (9.0.4), include the ORACLE_SID environment variable in the <*destination_Infra_OH*>/opmn/conf/opmn.xml file, as shown:

```
<?xml version = '1.0' encoding = 'UTF-8'?>
  <opmn xmlns="http://www.acme.com/ias-instance">
...
        </ias-component>
        <ias-component id="OID" status="enabled">
            <process-type id="OID" module-id="OID">
                <environment>
                 <variable id="ORACLE_SID" value="value_of_oracle_sid"/>
                </environment>
                <stop timeout="1800"/>
                <process-set id="OID" numprocs="1">
                 <dependencies>
...
</opmn>
```

j. Change the password of the replication DN by issuing this command on each replica:

**<*destination_Infra_OH*>/ldap/admin/remtool -presetpwd -v -bind <*host*>:<*port*>**

k. Ensure that the ORACLE_SID environment variable is set to the Oracle Internet Directory replica database.

l. Start the replication server, the LDAP server, and oidmon on all the replicas.

## 5.3 Upgrading Oracle Internet Directory v. 9.2.0.x to 10*g* (9.0.4)

Oracle Internet Directory version 9.2.0.x, shipped with the Oracle9i Release 2 database, was a standalone release of Oracle Internet Directory. The Oracle Internet Directory database repository contained only Oracle Internet Directory schema.

The 10*g* (9.0.4) release supports upgrade of a v. 9.2.0.x Oracle Internet Directory deployed with the Oracle 9.2 database repository. Follow the steps below to perform this upgrade.

1. Stop all processes in the Oracle home.
2. Back up the database.
3. If the Oracle Internet Directory database was created with the Oracle9*i* Management and Integration installation type, you must install the Oracle9*i* Database 9.2.0.1.0 Software Only installation type into the same Oracle home, over the database created with the Management and Integration installation type. The Software Only installation type has the options required to use a 9.2 database as a metadata repository.
4. Use the Repository Creation Assistant to convert the 9.2 database to a metadata repository. See Chapter 10, "Installing the OracleAS Metadata Repository in an Existing Database" in the *Oracle Application Server 10g Installation Guide*.

   **Note:** On the **Register with Oracle Internet Directory** screen of the Repository Creation Assistant, select **Register Later**.

   The metadata repository now has the 10*g* (9.0.4) version of the schema for all OracleAS components except Oracle Internet Directory. The Oracle Internet Directory schema is still at version 9.2.
5. Create the Oracle Internet Directory tablespaces `olts_svrmgstore` and `olts_battrstore` in the 9.2.0.4 Oracle Internet Directory database repository by executing the following SQL statements as SYS:

   a. `create tablespace olts_svrmgstore datafile 'svrmg1_oid.dbf' size 1M reuse autoextend on MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL;`

   b. `create tablespace olts_battrstore datafile 'battrs1_oid.dbf' size 500K reuse autoextend on EXTENT MANAGEMENT LOCAL AUTOALLOCATE;`

6. Perform a 10*g* (9.0.4) Identity Management-only installation in a separate Oracle home, or on a different computer. (Select Oracle Internet Directory only), specifying the 9.2 database as the metadata repository database.

   **See Also:** *Oracle Application Server 10g Installation Guide*, Chapter 6, "Installing Oracle Internet Directory Only".

   During the installation, the Oracle Internet Directory Configuration Assistant is invoked. It performs a version check on the Oracle Internet Directory schema; if the version is 9.2.0.x, then it upgrades Oracle Internet Directory to 10*g* (9.0.4). The other configuration tools function as they would when a new installation is performed.

   After the installation, the following conditions are in effect:

   - The Oracle Internet Directory server is running on the non-SSL and SSL ports, as determined by the 10*g* (9.0.4) installation process. The Oracle Internet Directory ports in use are identified in the `<destination_Infra_OH>/config/ias.properties`, in the `OIDport` and `OIDsslport` properties.
   - The Oracle Internet Directory superuser and Oracle Internet Directory database schema (ODS) password are set to the same value as the `ias_admin` password specified during the Identity Management installation.

7. Set up appropriate access control policies required for the 10*g* (9.0.4) DAS and middle tier installation to operate with the upgraded Oracle Internet Directory by following the steps below:

**a.** Create an ldif (`upgrade92.ldif`) file with the entry shown below. Each value of the `orclaci` attribute (shown in bold below) must be a single line, without any line breaks, or an error will occur.

```
#--- BEGIN LDIF file contents---
dn: cn=Attribute Configuration, cn=DAS,cn=Products,cn=OracleContext
changetype: modify
add: orclaci
orclaci: access to entry by group="cn=OracleDASConfiguration,
cn=Groups,cn=OracleContext" (add,delete,browse) by * (noadd,nodelete)
orclaci: access to attr=(*) by group="cn=OracleDASConfiguration,
cn=Groups, cn=OracleContext" (read,write,search,compare) by *
(nowrite,nocompare)

dn: cn=Attribute Configuration,
cn=DAS,cn=Products,cn=OracleContext,%rlmDN%
changetype: modify
add: orclaci
orclaci: access to entry by group="cn=OracleDASConfiguration,
cn=Groups,cn=OracleContext,%rlmDN%" (add,delete,browse) by *
(noadd,nodelete)
orclaci: access to attr=(*) by group="cn=OracleDASConfiguration,
cn=Groups, cn=OracleContext,%rlmDN%" (read,write,search,compare) by *
(nowrite,nocompare)

#---END LDIF file contents------
```

**b.** Replace all occurrences of `%rlmDN%` in the `upgrade92.ldif` with the default realm DN. You can determine the default realm DN with the ldapsearch command shown below:

```
ldapsearch -h <oid host> -p <oid port> -D <OID superuser
DN> -w <OID superuser password> -b
"cn=common,cn=products,cn=oraclecontext" -s base
"objectclass=*" orcldefaultsubscriber
```

**c.** Issue the ldapmodify command below:

```
<destination_Infra_OH>/bin/ldapmodify -p <oid port> -h
<oid host> -D <OID superuser name> -w <OID superuser
password> -v -f  upgrade92.ldif
```

**8.** Perform the tasks in Section 5.4.1, "Completing the Oracle Internet Directory Upgrade" on page 5-48.

# 5.4 Performing Infrastructure Post-Upgrade Tasks

This section details the post-upgrade procedures which will complete the Infrastructure upgrade to 10*g* (9.0.4). It is organized into these sections:

- Section 5.4.1, "Completing the Oracle Internet Directory Upgrade" on page 5-48
- Section 5.4.2, "Completing the Oracle Application Server Single Sign-On Upgrade" on page 5-52
- Section 5.4.3, "Completing the Oracle Application Server Wireless Upgrade" on page 5-57

## 5.4.1 Completing the Oracle Internet Directory Upgrade

To complete the Oracle Internet Directory Upgrade, you should reconfigure all associated OracleAS Portal 10*g* (9.0.4) instances, if applicable, and refresh the Delegated Administration Services (DAS) URL cache. You may also want to execute performance enhancement scripts, and, if applicable, install a new DAS or Directory Integration and Provisioning (DIP) service.

### 5.4.1.1 Applying Patches for Portal 9.0.2.2.14 and 9.0.2.3

Some Portal versions require that you apply a patch to the Metadata Repository, as explained below:

- **You are operating Portal version 9.0.2.2.14 (9.0.2 Production in Oracle9iAS 9.0.2.0.1):** You must apply Patch 3238095, which corrects problems with registering users and groups in Oracle9*i*AS Release 2 (9.0.2) Identity Management configuration, and resolves interoperability issues.
- **You are operating Portal 9.0.2.3 (Oracle9iAS 9.0.2.3)**: You must apply Patch 2802414 to resolve interoperability issues.

To apply the patches:

1. Log in to Oracle MetaLink at:

   `http://metalink.oracle.com`

2. Locate the patch specified for the Portal version you are operating.
3. Follow the instructions in the patch Readme file.

#### 5.4.1.2 Reconfigure the OracleAS Portal Instances for the Oracle Internet Directory Server

If there are any OracleAS Portal 10*g* (9.0.4) instances using the upgraded Oracle Internet Directory server, they should be reconfigured for the Oracle Internet Directory server, as described in Section 4.5.8.2, "Reconfiguring the OracleAS Portal for the Oracle Internet Directory" on page 4-43. This step is required to ensure that the OracleAS Portal entries in Oracle Internet Directory are properly updated, and that t0he correct provisioning events required by Oracle Application Server 10*g* (9.0.4) are sent to the Portal.

---

**Note:** This step is required only for the OracleAS Portal 10*g* (9.0.4) instances. If there are multiple instances using the upgraded Oracle Internet Directory server, you must repeat this step for each instance.

---

#### 5.4.1.3 Refreshing the Delegated Administration Services (DAS) URL Cache

The URLs for the Delegated Administration Services are different in Oracle9*i*AS Release 2 (9.0.2) Oracle Internet Directory server and the Oracle Application Server 10*g* (9.0.4) Oracle Internet Directory server. When the Oracle Internet Directory server is upgraded, these URLs are updated to the correct values. However, OracleAS Portal maintains a cache of these URLs, which does not get upgraded, and is thus inconsistent with the set of URLs in 10*g* (9.0.4).

The procedure for refreshing the cache is dependent on the version you have. To refresh the cache, follow the steps in one of the sections below.

**5.4.1.3.1 Refreshing the Cache in Version 9.0.2.6 or Later** Follow these steps to refresh the URL cache:

1. Log in to the Portal as a Portal administrator.
2. Click the **Administer** tab.
3. Click the **Global Settings** link in the **Services** portlet.
4. Click the **SSO/OID** tab.
5. Note the values that appear under the section **Cache for OID Parameters**.
6. Click the checkbox next to **Refresh Cache for OID Parameters**.
7. Click **Apply**.

8. Verify that the values displayed under **Cache for OID Parameters** have changed.
9. Click **OK**.

**5.4.1.3.2 Refreshing the Cache in Versions Prior to 9.0.2.6** Follow these steps to refresh the URL cache:

1. Apply the one-off patch 3225970. This patch is available at:

   `http://metalink.oracle.com`.
2. Clear the Web Cache by performing these steps:
   a. Log in to the Portal as a Portal Administrator.
   b. Click the **Administer** tab.
   c. Click the **Global Settings** link in the **Services** portlet.
   d. Click the **Cache** tab.
   e. Click the checkbox next to **Clear the Entire Web Cache**.
   f. Click **OK**.
3. Clear the middle tier cache by performing these steps:
   a. Navigate to *<destination_MT_OH>*`/Apache/modplsql/cache`.
   b. Perform a recursive delete of all files under this directory.

### 5.4.1.4 Recommended Performance Enhancement Tasks

In 10*g* (9.0.4), Oracle Internet Directory provides some performance enhancements that Oracle Corporation recommends that you implement after upgrading. The implementation involves running two scripts: `oidpu904.sql` and `catalog.sh`, as described below. In the 10*g* (9.0.4) Oracle home:

1. Ensure that the ORACLE_HOME environment variable is set to *<destination_Infra_OH>* and the ORACLE_SID environment variable is set to the infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
2. Issue this command:

   **`sqlplus ods/`*`<ods password>`*`@`*`<net service name for OID database>`*`@`*`<destination_Infra_OH>`*`/ldap/admin/oidpu904.sql`**

for example:

```
sqlplus ods/welcome1@iasdb@<destination_Infra_
OH>/ldap/admin/oidpu904.sql
```

3. Re-create the index for the `orclnormdn` attribute by executing the `catalog.sh` script, which drops and re-creates the catalog for the `orclnormdn` attribute.

   a. Ensure that the OID server is operating in read-only mode. You can do this with the Oracle Directory Manager.

      **See Also:** *Oracle Internet Directory Administrator's Guide*, Table C-34, System Operation Attributes (Server Mode field), for instructions on how to make the server operate in read-only mode.

   b. Issue these commands to re-create the index for the `orclnormdn` attribute:

      ```
      <destination_Infra_OH>/ldap/bin/catalog.sh -connect <net
      service name for OID database> -delete -attr orclnormdn
      ```

      ```
      <destination_Infra_OH>/ldap/bin/catalog.sh -connect <net
      service name for OID database> -add -attr orclnormdn
      ```

4. Reset the OID server to operate in read-write mode. You can do this with the Oracle Directory Manager.

   **See Also:** *Oracle Internet Directory Administrator's Guide*, Table C-34, System Operation Attributes (Server Mode field), for instructions on how to make the server operate in read-write mode.

---

**Note:** If you had an older version (9.0.2 or 9.2) of DIP operating in a different Oracle home (on a different computer) and using the Oracle Internet Directory you are upgrading now, and you want to continue using the DIP, you must re-register the DIP server. See *Oracle Internet Directory Administrator's Guide* for instructions on registering the DIP server.

---

## 5.4.2 Completing the Oracle Application Server Single Sign-On Upgrade

To complete the Oracle Application Server Single Sign-On upgrade, depending on the configuration ugpraded, you may need to perform the tasks below.

### 5.4.2.1 Re-configuring the Oracle Application Server Single Sign-On Middle Tier

If the Release 2 (9.0.2) middle tier for the Single Sign-On server had custom configurations (e.g., Oracle HTTP Server configured for SSL, or the Oracle Application Server Single Sign-On server Database Access Descriptor had any custom configuration), then you must re-configure the upgraded 10*g* (9.0.4) middle tier in a like manner.

> **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide*, Chapter 9

### 5.4.2.2 Configuring Third-party Authentication

If the Release 2 (9.0.2) middle tier was configured to authenticate with a user certificate or third party authentication mechanism, then you must re-configure the 10*g* (9.0.4) OracleAS Single Sign-On server in a like manner.

> **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide*, Chapter 13

### 5.4.2.3 Installing Customized Pages in the Upgraded Server

If you have customized the login, password and the sign-off pages in the Release 2 (9.0.2) Single Sign-On server, then you must update those pages with 10*g* (9.0.4) specifications.

> **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide*, Chapter 12

### 5.4.2.4 Converting External Application IDs

> **Note:** You do not need to perform this task if you upgraded from an OracleAS Single Sign-On version of 9.0.2.5 or later.

To avoid ID conflicts while exporting and importing external application data among multiple OracleAS Single Sign-On server instances, external application IDs must be unique. In the Release 2 (9.0.2) release, external application IDs were sequential, and not unique across instances. If you are upgrading from Release 2 (9.0.2) directly to 10*g* (9.0.4), then you must convert existing short external application IDs to the longer format in the OracleAS Single Sign-On schema. Follow the steps below to convert the IDs:

1. Execute the `orasso` script from the OracleAS Single Sign-On schema directory using these commands:

   ```
   sqlplus orasso/<password>

   spool extappid.log

   @?/sso/admin/plsql/sso/ssoupeid.sql

   spool off
   ```

   If you have OracleAS Portal versions that are lower than 9.0.2.6 and that use the upgraded OracleAS Single Sign-On server, then you must apply patches to each instance according to the table below. Patches are available at:

   `http://metalink.oracle.com`

   > **Note:** You might need the `SSO_IDENTIFIER` value to apply the patches, if the value cannot be generated in the OracleAS Portal schema automatically, or if the OracleAS Single Sign-On server used a randomly selected value for the SSO_IDENTIFIER.

***Table 5–1 OracleAS Portal Patches for Converting to Long Format Application IDs***

| OracleAS Portal Version | Patch Number |
|---|---|
| 3.0.9.8.4 | 2769007 |
| 3.0.9.8.5 | 2665597 |
| 9.0.2, 9.0.2.3 | 2665607 |

#### 5.4.2.5 Setting Up OracleAS Single Sign-On Replication

If you are using Oracle Internet Directory replication and want to also use OracleAS Single Sign-On replication, add the upgraded 10*g* (9.0.4) tables in the replication group along with 9.0.4 OID. Follow the steps below to add OracleAS Single Sign-On tables for replication:

1. Stop the Oracle Internet Directory replication server on all replicas of the Directory Replication Group.
2. On the Master Directory replica, in *$ORACLE_HOME*/ldap/admin, issue the following command:

   **sqlplus repadmin/<*password*>@<*mds connect id*> @oidrssou.sql**

3. Start the Oracle Internet Directory replication server on all replicas of the Directory Replication Group.

> **See Also:** *Oracle Internet Directory Administrator's Guide*, Chapter 25, Managing Directory Replication

#### 5.4.2.6 Upgrading the OracleAS Single Sign-On Server with a Customized Middle Tier

If the Release 2 (9.0.2) OracleAS Single Sign-On server was using a middle tier other than the default mid-tier installation along with the OracleAS Single Sign-On server, then you must configure that middle tier to point to the upgraded OracleAS Single Sign-On server. For example, if there was a reverse proxy configured in the Release 2 (9.0.2) OracleAS Single Sign-On server middle tier, then you must configure it on the 10*g* (9.0.4) OracleAS Single Sign-On server middle tier.

#### 5.4.2.7 Troubleshooting Wireless Voice Authentication

If you want to use wireless voice authentication with the 10*g* (9.0.4) OracleAS Single Sign-On server, and it doesn't work, verify that the OracleAS Single Sign-On server entry is a member of the Verifier Services Group in Oracle Internet Directory (cn=verifierServices,cn=Groups,cn=OracleContext). This is a requirement for the wireless voice authentication feature. Follow the steps below to verify membership:

1. Issue the following command:

```
ldapsearch -h <host> -p <port> -D cn=orcladmin -w
<password> -b
"cn=verifierServices,cn=Groups,cn=OracleContext"
"objectclass=*"
```

The OracleAS Single Sign-On server is a member of the Verifier Services Group if it is listed as a `uniquemember` in the entry, as shown in Example 5–1.

***Example 5–1 OracleAS Single Sign-On Server uniquemember Listing***

```
cn=verifierServices, cn=Groups,cn=OracleContext
.
.
.
uniquemember=orclApplication
CommonName=ORASSO_SSOSERVER,cn=SSO,cn=Products,cn=OracleContext
.
.
.
```

### 5.4.2.8 Installing Languages in the OracleAS Single Sign-On Server

If you did not select any languages during the OracleAS Single Sign-On upgrade, or you want to install additional languages after the upgrade, you can install the necessary languages by following the steps below.

1. Copy the necessary language files from the Repository Creation Assistant CD-ROM Oracle home to the OracleAS Single Sign-On server Oracle home:

   ***cp <repCA_CD>/portal/admin/plsql/nlsres/ctl/<lang>/*.* <destination_Infra_OH>/sso/nlsres/ctl/<lang>/***

   where <*lang*> is the language code. For example, the language code for Japanese is `ja`.

2. Load the languages into the server.

   **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide*, Chapter 2, "Configuring Globalization Support" section.

#### 5.4.2.9 Re-Registering OracleAS Portal with the Upgraded OracleAS Single Sign-On Server

After performing a distributed Identity Management upgrade (depicted in Figure 5–2 and Figure 5–3) from Oracle9*i*AS Release 2 (9.0.2) to Oracle AS 10*g* (9.0.4), the OracleAS Single Sign-On schemas are relocated in the Oracle Internet Directory database. OracleAS Portal keeps a database link reference to the OracleAS Single Sign-On server password store schema `ORASSO_PS`. This link reference must be updated. To do this, re-register the corresponding OracleAS Portal with the upgraded OracleAS Single Sign-On server.

> **See Also:** *Oracle Application Server Portal Configuration Guide*, Appendix B.

#### 5.4.2.10 Re-Registering mod_osso with the Upgraded OracleAS Single Sign-On Server

After performing a distributed Identity Management upgrade (depicted in Figure 5–2 and Figure 5–3) from Oracle9*i*AS Release 2 (9.0.2) to Oracle AS 10*g* (9.0.4), you may need to re-register `mod_osso` in order for an Oracle9*i*AS Release 2 (9.0.2) middle tier to operate with the upgraded OracleAS Single Sign-On server. You will need to do this if the Oracle HTTP Server host and port information for mod_osso was changed. Before re-registering `mod_osso`, you must first set the value of the `ColocatedDBCommonName` attribute in the `<source_MT_OH>/config/ias.properties` file to the global database name of the new OracleAS Single Sign-On server database shared with Oracle Internet Directory (for example, `iasdb.host.mydomain`).

#### 5.4.2.11 Using an Upgraded Identity Management Configuration with Oracle9*i*AS Discoverer Release 2 (9.0.2)

If you upgraded an Identity Management configuration that was in use by Oracle9*i*AS Discoverer Release 2 (9.0.2), and you want to continue operating Oracle9*i*AS Discoverer Release 2 (9.0.2) with the upgraded Identity Management, then you must change the value of the `ColocatedDBCommonName` attribute in the `<source_MT_OH>/config/ias.properties` file. The value must be changed to the global database name of the database used by the upgraded Oracle Internet Directory (e.g., *`iasdb.oid_host_name.domain`*).

## 5.4.3 Completing the Oracle Application Server Wireless Upgrade

This section describes the tasks you must perform in order to complete the Oracle Application Server Wireless upgrade.

### 5.4.3.1 Upgrading Wireless User Accounts in Oracle Internet Directory

In Oracle Application Server Wireless Release 2 (9.0.2), user account numbers and PINs for wireless voice authentication were stored in the Wireless repository.

In Oracle Application Server Wireless 10*g* (9.0.4), new attributes are added in the object definition of the `orcluserV2` object class of Oracle Internet Directory to store the account number and PIN. As part of the Oracle Application Server Wireless upgrade from Release 2 (9.0.2) to 10*g* (9.0.4), user account numbers and PINs must be transferred from the Wireless repository to Oracle Internet Directory.

This upgrade step can be performed only after the Oracle AS Infrastructure and all middle tiers are upgraded to 10*g* (9.0.4). If they are not performed, the Oracle Application Server Wireless server will continue to authenticate voice devices locally (without Oracle Application Server Single Sign-On).

To upgrade the account numbers and PINs:

1. Issue the command:

   ***<destination_MT_OH>*/wireless/bin/migrate902VoiceAttrsToOID.sh *<destination_MT_OH>* *<ldapmodify location>* *<userdn>* *<password>* *<ldif file location>* *<log file>***

   where:

   - *<ldapmodify location>* is the location of the `ldapmodify` utility (usually *<destination_MT_OH>*`/bin`)
   - *<user dn>* is the DN of the Oracle Internet Directory administrator user
   - *<password>* is the password of the Oracle Internet Directory administrator user
   - *<ldif file location>* is the absolute path to the ldif (Lightweight Directory Interchange Format) file. This file contains user account numbers and PINs and is uploaded to Oracle Internet Directory by the `ldapmodify` utility. This temporary file may be removed after the user upgrade procedure has been completed successfully.
   - *<log file>* is the absolute path to the log file

Example:

```
migrate902VoiceAttrsToOID.sh /private/ias904/
/private/ias904/bin/ldapmodify cn=orcladmin welcome1
/private/ias904/users.ldif /private/ias904/users.log
```

#### 5.4.3.2 Adding Unique Constraint on the orclWirelessAccountNumber Attribute in Oracle Internet Directory

In 10*g* (9.0.4), Oracle Internet Directory does not automatically set unique constraints on any user attributes. Wireless voice authentication will not function properly unless a unique constraint is set on the `orclWirelessAccountNumber` attribute of the `orclUserV2` object class.

Set the unique constraint by performing the steps below after the middle tier and infrastructure upgrades are complete.

1. Execute `<destination_MT_OH>/wireless/bin/addAccountNumberUniqueConstraint.sh`. The script takes one argument, the full path to the Oracle home. For example:

   ```
   addAccountNumberUniqueConstraint.sh <destination_MT_OH>
   ```

2. Restart the Oracle Internet Directory server.

#### 5.4.3.3 Disabling Oracle Application Server Wireless Upgrade Triggers in the Infrastructure Repository

When Oracle Application Server Wireless 10*g* (9.0.4) is installed against an Oracle9iAS Release 2 (9.0.2) infrastructure, a number of triggers are automatically installed, that ensure that both Oracle9iAS Wireless Release 2 (9.0.2) and Oracle Application Server Wireless 10*g* (9.0.4) middle tiers can function correctly. Once all Oracle9iAS Wireless Release 2 (9.0.2) middle tiers and the infrastructure tier have been upgraded to Oracle Application Server Wireless 10*g* (9.0.4), you must execute the following script to disable any upgrade-related triggers.

```
disable902-904_trg.sh
```

This script is located in the `<destination_MT_OH>/wireless/bin` directory. You must set the ORACLE_HOME environment variable before you execute the script.

#### 5.4.3.4 Activating All Oracle Application Server Wireless 10*g* (9.0.4) Features

When Oracle Application Server Wireless 10*g* (9.0.4) is installed against an Oracle9iAS Release 2 (9.0.2) Infrastructure, a number of features are disabled by

default, as they are not compatible with existing Oracle9iAS Wireless Release 2 (9.0.2) middle tiers that are installed against the same Infrastructure. After all Oracle9iAS Wireless Release 2 (9.0.2) middle tiers have been upgraded to Oracle Application Server Wireless 10g (9.0.4), you can manually enable these features. Once you have enabled these features, the Oracle9iAS Wireless Release 2 (9.0.2) middle tiers will no longer function correctly.

Enable the Oracle Application Server Wireless 10*g* (9.0.4) features by executing the following script from any of the Oracle Application Server Wireless 10*g* (9.0.4) middle tiers, using the command below. This script is in the *<destination_MT_OH>*/wireless/bin directory.

**upload.sh ../repository/xml/activate-9040.xml -l *<wireless user name>*/*<password>***

where:

- *<wireless user name>* is the name of the Oracle Application Server Wireless user
- *<password>* is the password of the Oracle Internet Administrator

For example:

```
upload.sh ../repository/xml/activate-9040.xml -l
orcladmin/welcome1
```

### 5.4.3.5 Assigning Change Password Privilege to OracleAS Wireless

In Oracle Application Server 10*g* (9.0.4), by default, the OracleAS Wireless application entity does not have the privileges to change the user password. Consequently, upon installation, users cannot change the password to the OracleAS Wireless server. However, you can enable functionality to change passwords by assigning the UserSecurityAdmins privilege to the OracleAS Wireless application entity.

To do this, execute the script *<destination_MT_OH>*wireless/bin/assignUserSecurityAdminsPrivilege.sh

The syntax is:

assignUserSecurityAdminsPrivilege.sh *<oid super user dn>* *<user password>*

where:

- *<oid super user dn>* is the Distinguished Name of the Oracle Internet Directory super user. This user should have privileges to grant UserSecurityAdmins privileges to application entities.
- *<user password>* is the password of the Oracle Internet Directory super user.

For example:

```
assignUserSecurityAdminsPrivilege.sh cn=orcladmin welcome1
```

#### 5.4.3.6 Specifying URL Query Parameters for Wireless Services That Use the HTTP Adapter

When you use the HTTP adapter to build Wireless services, one of the service parameters that you must specify is the URL to a back-end application. In some cases, you may send some query parameters to the back-end application. There are two ways to do this from OracleAS Wireless, shown in Example 5–2 and Example 5–3. In Example 5–2, the parameter name is `fn` and the value is `Joe`.

***Example 5–2 URL Using a Query Parameter***

```
http://localhost:7777/myapp/home.jsp?fn=Joe
```

The query parameter is sent only in the request for the first page of that service. If there is a link from the first page to some other pages, then the parameter is not added to the request for those pages.

***Example 5–3 URL Using an Extra Service Parameter***

```
http://localhost:7777/myapp/home.jsp
```

Instead of modifying the URL, you add an extra service parameter with name `fn` and value `Joe`. The the parameter is sent to all pages, not just the first one. The parameter is also sent with all HTTP redirect requests. However, this method also sends extra URL parameters to the OracleAS Single Sign-On server, which causes the server to return an error.

The error occurs when the back-end application is protected by mod_osso. In that case, the request to that application is intercepted and redirected to the Oracle SSO server for user authentication. The OracleAS Single Sign-On server has restrictive rules concerning query parameters that can be sent to it. Consequently, for back-end applications protected by mod_osso, you must change the Wireless service and add the query parameter to the URL as shown in Example 5–2.

## 5.5 Decommissioning the Release 2 (9.0.2) Oracle Home

After you complete the Identity Management upgrade, you will probably want to consider relocating the database files to a location outside of the source Oracle home. Even after the Identify Managment upgrade is complete, the database files still remain in the source Oracle home. If you decide to deinstall the source Oracle home, these database files will still remain there unless you take steps to relocate them. It is a good idea to relocate the files as a safeguard against inadvertently deleting them (for example, by deleting the entire source Oracle home directory tree). In addition, there may be performance benefits to moving the database files outside of the source Oracle home.

After the database files have been relocated and the software in the source Oracle home has been deinstalled, then you may safely delete the entire source Oracle home directory tree.

This procedure is intended to be performed by a database administrator, and is described in greater detail in the *Oracle9i Database Administrator's Guide*.

### 5.5.1 Deinstalling Oracle9*i*AS Release 2 (9.0.2) and Deleting the Source Oracle Home

If you have relocated the Release 2 (9.0.2) files, you may wish to delete the old Oracle home. To do this, deinstall the Release 2 (9.0.2) infrastructure instance in the source Oracle home using the same version of Oracle Universal Installer that was used to install it, or a later version, and then delete all files from *<source_Infra_OH>*.

Deinstalling an Oracle9*i*AS Release 2 (9.0.2) or (9.0.3) instance when there is also an OracleAS 10*g* (9.0.4) instance on the computer requires a patch. Before you deinstall such an instance, be aware of the issues associated with this deinstallation that may apply to your configuration.

> **See Also:** *Oracle Application Server 10g Installation Guide* for information on deinstalling a Release 2 (9.0.2) or (9.0.3) instance when a 10g (9.0.4) instance exists on the same computer.

### 5.5.2 Relocating Data, Control, and Log Files

Follow these steps to relocate data, control, and log files.

1. Create a directory for the relocated files in a location that is separate from the source Infrastructure Oracle home.
2. Copy all data files to the directory created in Step 1.

   **See Also:** *Oracle9i Database Administrator's Guide*, section titled "Renaming and Relocating Datafiles"

3. Copy all log files to the directory created in Step 1.

   **See Also:** *Oracle9i Database Administrator's Guide*, section titled "Renaming and Relocating Datafiles"

4. Relocate all control files to the directory created in Step 1.

   **See Also:** *Oracle9i Database Administrator's Guide*, section titled "Creating Additional Copies, Renaming, and Relocating Control Files"

## 5.6 Validating the Identity Management Upgrade

This section describes the steps you must perform after the Identity Management Upgrade to ensure that the upgrade was successful.

### 5.6.1 Executing the utlrp.sql Utility

You must run the `utlrp.sql` utility as a post-installation step. This PL/SQL procedure recompiles all PL/SQL packages that may have been invalidated during the upgrade to 10*g* (9.0.4). To run this utility, do the following:

1. Ensure that the upgraded Metadata Repository database is running.
2. Ensure that the ORACLE_HOME environment variable is set to *<Infra_OH>* and the ORACLE_SID environment variable is set to the Infrastructure database SID. If they are not, follow the instructions in Section 4.2.1, "Setting the Environment for Upgrading the Metadata Repository" on page 4-4.
3. Connect to the database in the destination Infrastructure Oracle home as SYS as SYSDBA in single user mode.
4. Issue the following command at the SQL*Plus prompt:

   ```
   @?/rdbms/admin/utlrp.sql
   ```

### 5.6.2 Checking for Invalid Database Objects

Follow these steps to ensure that none of the database objects that are required by Oracle Application Server are invalid:

1. Connect to the database in the destination Infrastructure Oracle home as SYSDBA.
2. Issue the following command:

   ```
   SELECT owner, object_type, object_name
     FROM all_objects
     WHERE status='INVALID';
   ```

The query should not return any database objects that have an Oracle AS component schema (such as PORTAL, WIRELESS, etc.) in the 'owner' column.

### 5.6.3 Testing Oracle Application Server Single Sign-On Connectivity

After the Identity Management upgrade is complete, log in to Oracle Application Server Single Sign-On as user ORCLADMIN. A successful login indicates that Oracle Application Server Single Sign-On and Oracle Internet Directory are functioning after the Identity Management upgrade.

1. In a browser, access the Oracle Enterprise Manager in the destination Infrastructure Oracle home by entering its URL. Ensure that you provide the correct host name and port number. For example:

   ```
   http://infrahost.mycompany.com:1812
   ```

   The Oracle Enterprise Manager page displays, withthe Oracle Application Server 10*g* (9.0.4) Identity Management instance in the **Standalone Instances** section.

2. Click the link for the Identity Management instance.

   The **System Components** page appears.

3. Verify that the status of the Oracle HTTP Server, Oracle Internet Directory, and Oracle Application Server Single Sign-On components is **Up**.

4. In the browser, access the ORASSO page by entering its URL. Ensure that you enter the correct host name and port number for the upgraded Oracle HTTP Server. For example:

   ```
   http://infrahost.mycompany.com:7777/pls/orasso/ORASSO.home
   ```

   The ORASSO page appears.

5. Click the **Login** link (in the upper right corner of the page).

   A page appears with **User Name** and **Password** fields.

6. Enter ORCLADMIN in the User Name field, and the password you have selected for ORCLADMIN in the Password field.

7. Click **Login**.

   The Oracle Application Server Single Sign-On Server **Administration** page appears, thus validating the basic operation of the upgraded Identity Management components (Oracle Application Server Single Sign-On and Oracle Internet Directory).

# A

# Component Upgrade Process Reference

This appendix provides details on the upgrade process for each component of the middle tier and the Infrastructure installations. The OracleAS Upgrade Assistant does the middle tier upgrade processing only; the Infrastructure upgrade processing is done by individual scripts for the components that require upgrades. This appendix has two major sections:

Section A.1, "Middle Tier Upgrade Processes" on page A-1

Section A.2, "Infrastructure Upgrade Processes" on page A-19

## A.1 Middle Tier Upgrade Processes

This section describes the processing actions of the OracleAS Upgrade Assistant for each component upgrade. The order in which the actions are presented is not necessarily the order in which they are actually performed. When the order is known, or significant, the processing sequence is presented as numbered steps. Depending on installation type (that is, the components included), not all of these processes are performed.

---

**Notes:** If the Release 2 (9.0.2) installation type is Unified Messaging (an installation type that is no longer available) the OracleAS Upgrade Assistant processes it as a Business Intelligence and Forms upgrade, performing all component upgrades.

Oracle9*i*AS Clickstream Intelligence is not included in 10*g* (9.0.4), so there is no upgrade process for it.

---

1. The Oracle Process Manager and Notification Server (OPMN) Upgrade Process on page A-2
2. The Instance Configuration Data Upgrade Process on page A-3
3. The Oracle Application Server Containers for J2EE (OC4J) Upgrade Process on page A-4
4. The Oracle HTTP Server Upgrade Process on page A-5
5. The Oracle Application Server Web Cache Upgrade Process on page A-8
6. The mod_plsql Upgrade Process on page A-10
7. The Oracle Enterprise Manager Upgrade Process on page A-10
8. The Oracle Application Server Web Services UDDI Registry Upgrade Process on page A-11
9. The Oracle Ultra Search Upgrade Process on page A-12
10. The OracleAS Portal Middle Tier Upgrade Process on page A-12
11. The Oracle Application Server Wireless Upgrade Process on page A-13
12. The Oracle Application Server Forms Services Upgrade Process on page A-16
13. The Oracle Application Server Discoverer Upgrade Process on page A-16
14. The Oracle Application Server Reports Services Upgrade Process on page A-17

### A.1.1 The Oracle Process Manager and Notification Server (OPMN) Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading the Oracle Process Manager and Notification Server:

1. Converts the `<source_MT_OH>/opmn/conf/opmn.xml` file to the 10*g* (9.0.4) format.
2. Merges the converted `<source_MT_OH>/opmn/conf/opmn.xml` file with the `<destination_MT_OH>/opmn/conf/opmn.xml` file. Inserts all custom nodes into `<destination_MT_OH>/opmn/conf/opmn.xml` except the node containing `gid="dcm-daemon"`.

#### A.1.1.1 Oracle Process Manager and Notification Server (OPMN) Upgrade Items

The following files are changed or copied in the OPMN upgrade process:

- <*destination_MT_OH*>/opmn/conf/opmn.xml
- Custom process binaries defined in opmn.xml
- Oracle wallets

---

**Notes:** Port values of particular interest for OPMN are the notification server ports (local, remote, request) and OC4J ports (ajp, rmi, jms).

The OPMN upgrade process does not upgrade any changes that were made to Oracle Application Server Containers for J2EE instances in opmn.xml. This includes the instances created by the installer (home, OC4J_WIRELESS, OC4J_DEMOS, OC4J_PORTAL OC4J_BI_FORMS) and instances created by users. The OC4J upgrade process upgrades the home instance and any user-defined instances with applications deployed in the source Oracle home. The other installer-created OC4J instances adopt the Oracle Application Server 10*g* (9.0.4) settings in opmn.xml. If you want to preserve settings from Oracle9*i*AS Release 2 (9.0.2), you must create them manually.

See Section 3.8.2.3, "Upgrading OC4J Instances Created by the Installer" on page 3-34.

---

### A.1.2 The Instance Configuration Data Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading instance configuration data:

1. Loads the <*source_MT_OH*>/config/iasschema.xml file from the source Oracle home.
2. Compares the <*source_MT_OH*>/config/iasschema.xml file with the <*destination_MT_OH*>/config/iasschema.xml file.
3. Writes any differences found in the source Oracle home file to the destination Oracle home file.

#### A.1.2.1 Instance Configuration Upgrade Items

The following file is changed in the instance configuration data upgrade process:

- *<destination_MT_OH>*/config/iasschema.xml

### A.1.3 The Oracle Application Server Containers for J2EE (OC4J) Upgrade Process

The Oracle Application Server Containers for J2EE (OC4J) upgrade process consists of these steps:

1. The OracleAS Upgrade Assistant creates a separate process, which connects to the source Oracle home.
2. The process uses its Distributed Configuration Management version to examine the instances selected for upgrade.
3. Distributed Configuration Management creates a list of the instances in the source Oracle home in which the applications are deployed. It ignores Oracle-specific OC4J instances, such as OC4J_Portal, defined in the *<source_MT_OH>*/j2ee/deploy.ini* file. The instances are the OC4J upgrade candidates.
4. Distributed Configuration Management builds a list of EAR files for the applications listed.
5. The OracleAS Upgrade Assistant creates a backup of upgraded files, appending a preUpgrade suffix. If necessary, to create a unique file name, it appends an integer, for example, *<file name>*.preUpgrade.1.
6. The OracleAS Upgrade Assistant copies principals.xml, data-sources.xml, jazn-data.xml, and jazn.xml to the destination Oracle home.
7. Adds properties defined in the oc4j.properties file to the opmn.xml file, using the SMI API.
8. The OracleAS Upgrade Assistant rebuilds, and then redeploys the EAR files to the destination Oracle home. In this step, the OracleAS Upgrade Assistant searches for all orion-specific files in the application-deployments directory of the applications. It also searches for application-specific configuration files, such as principals.xml and jazn-data.xml.
9. Distributed Configuration Management updates mod_oc4j.conf with the mount points associated with each deployed application.

### A.1.3.1 OC4J Upgrade Items

The following files are changed in the OC4J upgrade process:

- *<destination_MT_OH>*/j2ee/*<name of OC4J instance>*/config/principals.xml
- *<destination_MT_OH>*/j2ee/*<name of OC4J instance>*/config/data-sous.xml
- *<destination_MT_OH>*/Apache/Apache/conf/mod_oc4j.conf
- EAR files for applications discovered in *<source_MT_OH>* OC4J instances.
- *<destination_MT_OH>*/j2ee/*<name of OC4J instance>*/config/jazn-data.xml
- *<destination_MT_OH>*/j2ee/*<name of OC4J instance>*/config/jazn.xml
- *<destination_MT_OH>*/opmn/conf/opmn.xml (properties from the oc4j.properties file in *<source_MT_OH>* are added to it)

---

**Note:** Manual procedures may be required to complete the Oracle Application Server Containers for J2EE upgrade. See Section 3.8.2, "Completing the Oracle Application Server Containers for J2EE (OC4J) Upgrade" on page 3-33.

---

## A.1.4 The Oracle HTTP Server Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle HTTP Server (OHS):

- Copies the httpd.conf file from the source Oracle home to the destination Oracle home, replacing the *<source_MT_OH>* path with *<destination_MT_OH>*, then applies 10*g* (9.0.4) file changes, and customizations made since Release 2 (9.0.2), to the corresponding file in the destination Oracle home.
- Searches the mod_oc4j.conf file in *<source_MT_OH>* for Oc4jMount directives and copies the Oc4jMount directives that contain the string ajp13://, cluster:// or instance:// to the mod_oc4j.conf file in *<destination_MT_OH>*.

- Copies the `mod_osso.conf` file from the source Oracle home to the destination Oracle home, replacing the source Oracle home path with the destination Oracle home path. The `osso.conf` file referenced by the OssoConfigFile directive will be copied and converted into 10*g* (9.0.4) obfuscated files.
- Copies the `moddav.conf` file from the source Oracle home to the destination Oracle home, replacing the source Oracle home path with the destination Oracle home path.
- Searches `Include` directives in the `httpd.conf` file recursively to locate user-defined configuration files; copies these files from the source Oracle home to `the destination Oracle home`. If the files were found in the source Oracle home, the OracleAS Upgrade Assistant replaces the source Oracle home path with the destination Oracle home path. If the files were found outside of the source `Oracle home`, the OracleAS Upgrade Assistant saves a copy of the original file with a `.preUpgrade` extension in the destination Oracle home, then replaces the source Oracle home file with the new file.
- Searches the `LoadModule` directives recursively to find related module dynamic libraries; copies the libraries from the source Oracle home to the destination Oracle home.
- Searches the `SSLWallet` directives recursively to find Oracle wallets; copies the wallets from the source Oracle home to the destination Oracle home.
- Locates CGI and fastcgi scripts by searching all configuration files for directories and files named in `ScriptAlias` or `ScriptAliasMatch` directives, and the `ExecCGI` option in `Options` directives (defined in `Directory` or `File` containers). Copies the directories and files from the source Oracle home to the destination Oracle home.
- Copies static document directories found in the (non-default) location specified by the `DocumentRoot` directive from the source Oracle home to the destination Oracle home. If the `DocumentRoot` directive is the default, static documents are not upgraded.

**Notes:** Web sites are often configured with Web Cache as the first listener; in these cases, the Oracle HTTP Server's Listen port may need to be synchronized with equivalent Web Cache port values after upgrade. The settings are shown in Table 3–5, "Oracle HTTP Server and Oracle Application Server Web Cache Port Settings" on page 3-32.

The OracleAS Upgrade Assistant does not upgrade the `oracle_apache.conf` and `mod_plsql.conf` files.

Static files referenced by the `Alias` or `mod_rewrite` directives are not upgraded. Any such files in an Oracle home must be upgraded manually.

### A.1.4.1 OHS Upgrade Items

The following files are changed or copied in the OHS upgrade process:

- *<destination_MT_OH>*`/Apache/Apache/conf/httpd.conf`
- *<destination_MT_OH>*`/Apache/Apache/conf/mod_oc4j.conf`
- *<destination_MT_OH>*`/Apache/Apache/conf/mod_osso.conf`
- *<destination_MT_OH>*`/Apache/Apache/oradav/conf/moddav.conf`
- *<destination_MT_OH>*`/Apache/Apache/conf/osso/osso.conf`
- User-defined configuration files named in `Include` directives (found in recursive search of all configuration files starting with `httpd.conf`)
- `.so` files (module dynamic libraries) named in `LoadModule` directives in all configuration files
- Oracle wallets named in `SSLWallet` directives in all configuration files
- CGI and fastcgi programs named in `ScriptAlias`, `ScriptAliasMatch` or `Options (ExecCGI)` directives in all configuration files
- Static documents and directories

**Note:** Manual procedures may be required to complete the Oracle HTTP Server upgrade. See Section 3.8.1, "Completing the Oracle HTTP Server Upgrade" on page 3-31.

## A.1.5 The Oracle Application Server Web Cache Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle Application Server Web Cache:

1. Locates `webcache.xml` and `internal.xml` in the source Oracle home.
2. Moves configuration data from `<source_MT_OH>/webcache/webcache.xml` to `<destination_MT_OH>/webcache/webcache.xml`.
3. Copies error pages and wallet files from the source Oracle home to the destination Oracle home.

---

**Note:** If you customized the directory location of the event log (specified by `ACCESSLOG LOGDIR` property in the `webcache.xml` file) in Oracle9*i*AS Release 2 (9.0.2), be aware that this customization will not be upgraded.

---

### A.1.5.1 Oracle Application Server Web Cache Upgrade Items

The following files are involved in the Oracle Application Server Web Cache upgrade process:

- `<destination_MT_OH>/webcache/webcache.xml`
- `<destination_MT_OH>/webcache/docs/*` (error page files copied from `the source Oracle home` to the destination Oracle home.)
- Wallet files (copied from `<source_MT_OH>/webcache/wallets` to `<destination_MT_OH>/webcache/wallets`; see explanation below)

**A.1.5.1.1 Wallet File Upgrades** The OracleAS Upgrade Assistant upgrades wallets by copying them from the source Oracle home to the destination Oracle home. Wallets outside of the source Oracle home need not be copied.

Oracle Application Server Web Cache can have multiple listening ports, and each port can have a different wallet. When connecting to the origin server, it can use another wallet (`OSWALLET` in the example below).

***Example A–1 Oracle Application Server Web Cache Wallet Configuration after Upgrade***

```
<LISTEN IPADDR="ANY" PORT="4445" PORTTYPE="NORM" SSLENABLED="SSLV3_V2H">
      <WALLET><destination_MT_OH>/webcache/wallets/subdir1</WALLET>
</LISTEN>
 <LISTEN IPADDR="ANY" PORT="4447" PORTTYPE="NORM" SSLENABLED="SSLV3_V2H">
       <WALLET>/some/other/path/wallets/default</WALLET>
 </LISTEN>
    ......
    ......
       <OSWALLET><destination_MT_OH>/webcache/wallets/default</OSWALLET>
```

In this example, Oracle Application Server Web Cache is using three wallets. The first and third are in the source Oracle home. The first wallet will be copied to *<destination_MT_OH>*/webcache/wallets/subdir1. The third wallet will be copied to *<destination_MT_OH>*/webcache/wallets/default. The second wallet will not be copied, since it does not reside in the Oracle home. After upgrade, the wallet in webcache.xml will be pointing to the original directory.

---

**Note:** Web sites are often configured with Web Cache as the first listener; in these cases, the Oracle HTTP Server's Listen port may need to be synchronized with equivalent Web Cache port values after upgrade. See Section 3.8.1, "Completing the Oracle HTTP Server Upgrade" on page 3-31.

---

## A.1.6 The mod_plsql Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading mod_plsql:

1. Locates `dads.conf` and `cache.conf` in the source Oracle home.
2. Parses each source item, keeping results in memory.
3. Applies the parsing results to the destination Oracle home.
4. If necessary, uses the default value (`<destination_MT_OH>/Apache/modplsql/cache/`) for the `PlsqlCacheDirectory` property in the `cache.conf` file.
5. Copies the `oradav.conf` file from the source Oracle home to the destination Oracle home.

### A.1.6.1 mod_plsql Upgrade Items

The following files are involved in the mod_plsql upgrade process:

- `<destination_MT_OH>/Apache/modplsql/conf/dads.conf`
- `<destination_MT_OH>/Apache/modplsql/conf/cache.conf`
- `<destination_MT_OH>/Apache/oradav/conf/oradav.conf`

## A.1.7 The Oracle Enterprise Manager Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle Enterprise Manager:

1. Examines the `<source_MT_OH>sysman/emd/targets.xml` file for port entries related to the targets in the file.
2. Replaces the corresponding port entries in the `<destination_MT_OH>sysman/emd/targets.xml` file.

### A.1.7.1 Oracle Enterprise Manager Upgrade Items

The following files are changed in the Oracle Enterprise Manager upgrade process:

- `<destination_MT_OH>/sysman/emd/targets.xml`

## A.1.8 The Oracle Application Server Web Services UDDI Registry Upgrade Process

The Oracle Universal Installer and the OracleAS Upgrade Assistant perform these steps during installation of the Portal and Wireless installation type:

1. The OracleAS Upgrade Assistant extracts the *<source_MT_OH>*/ds/uddi/config/uddiserver.config file for later use.
2. The OracleAS Upgrade Assistant extracts properties from the *<source_MT_OH>*/ds/uddi/config/uddiserver.config file and applies them to the *<destination_MT_OH>*/uddi/config/uddiserver.config file.

### A.1.8.1 Oracle Application Server Web Services UDDI Registry Upgrade Items

The following files are changed in the Oracle Application Server Web Services Registry upgrade process:

- *<destination_MT_OH>*/uddi/config/uddiserver.config

---

**Notes:** If the repository database schema was not upgraded before the Oracle Application Server Web Services upgrade, the UDDI registry for Oracle Application Server Web Services will run in 9.0.2.3 compatibility mode.

You must restart UDDI applications after the upgrade so that the new configuration takes effect.

The path for the uddiserver.config file in 10*g* (9.0.4) is different from the path in Oracle9*i*AS Release 2 (9.0.2).

In Oracle9*i*AS Release 2 (9.0.2), it was:

*<Oracle home>*/ds/uddi/config/uddiserver.config

In Oracle Application Server 10*g* (9.0.4), it is:

*<Oracle home>*/uddi/config/uddiserver.config

---

## A.1.9 The Oracle Ultra Search Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle Ultra Search:

1. Discover the `data-sources.xml` file in the source Oracle home.
2. Copy the `data-sources.xml` file from *<source_MT_OH>* to *<destination_MT_OH>*.

### A.1.9.1 Oracle Ultra Search Upgrade Items

The following files are changed in the Oracle Ultra Search upgrade process:

- *<destination_MT_OH>*`/j2ee/OC4J_Portal/config/data-sources.xml`

## A.1.10 The OracleAS Portal Middle Tier Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading the OracleAS Portal middle tier:

1. Retrieves the source Oracle home Web Cache listener, administration and invalidation ports.
2. Applies these ports to the destination Oracle home.
3. Retrieves Oracle Enterprise Manager target data from the source Oracle home.
4. Applies these details to the destination Oracle home.

### A.1.10.1 OracleAS Portal Upgrade Items

The following files are changed in the OracleAS Portal upgrade process:

- *<destination_MT_OH>*`/portal/conf/iasconfig.xml`
- *<destination_MT_OH>*`/sysman/emd/targets.xml`

---

**Note:** Manual procedures are required to complete the Portal upgrade. See Section 3.8.4, "Completing the OracleAS Portal Middle Tier Upgrade" on page 3-45.

---

## A.1.11 The Oracle Application Server Wireless Upgrade Process

The Oracle Application Server Wireless upgrade process consists of the steps below.

1. The Oracle Application Server Wireless Configuration Assistant upgrades the Oracle Application Server Wireless schema in the Oracle Application Server Metadata Repository during installation of the first Oracle Application Server Wireless middle tier. (See Section A.1.11.2, "Oracle Application Server Wireless Upgrade Items (List A)" on page A-15).
2. The OracleAS Upgrade Assistant copies runtime customization classes such as listener hooks and customized folder renderers from the source Oracle home to the destination Oracle home. (See Section A.1.11.2, "Oracle Application Server Wireless Upgrade Items (List A)" on page A-15).
3. The OracleAS Upgrade Assistant copies the process configuration information for the Oracle Application Server Wireless standalone processes from the Oracle Application Server Wireless Release 2 (9.0.2) middle tier to the Oracle Application Server Wireless 10*g* (9.0.4) middle tier, and configures the 10*g* (9.0.4) middle tier accordingly. (See Section A.1.11.3, "Oracle Application Server Wireless Upgrade Items (List B)" on page A-15).

   The configuration information for the Oracle Application Server Wireless Java processes is stored in the Oracle Application Server Wireless schema in the metadata repository. During upgrade from Release 2 (9.0.2) to 10*g* (9.0.4), the OracleAS Upgrade Assistant copies the process configuration information from the Release 2 (9.0.2) middle tier to the 10*g* (9.0.4) middle tier by creating additional entries in the Oracle Application Server Wireless schema. Because these processes are managed by Oracle Process Management and Notification in 10*g* (9.0.4), the Oracle Process Management and Notification configuration for the Oracle Application Server Wireless middle tier is also upgraded.

**Notes:** Regarding upgrade of Oracle Application Server Wireless Java processes:

In Release 2 (9.0.2), process information was node-specific. In 10*g* (9.0.4), it is Oracle home-specific. Thus, if you have multiple Release 2 (9.0.2) middle tiers on one node, you will notice that the upgrade process copies the information for all Release 2 (9.0.2) to the 10*g* (9.0.4) instance being upgraded,

Instance names within a web site must be unique. During upgrade, the names of copied processes are made unique by appending a number to the name.

In addition to the Release 2 (9.0.2) processes, the upgraded 10*g* (9.0.4) will contain new process types introduced in 10*g* (9.0.4).

The process type "Alert Engine" has been renamed to "Notification Engine". The "Industrial Device Portal (Release 2 (9.0.2)) and "PIM Notification Dispatcher" (Oracle Collaboration Suite v. 1) processes are obsolete and therefore not upgraded.

### A.1.11.1 Upgrade of Oracle Application Server Wireless Middle Tiers and Wireless Schema

The steps below describe the process for upgrading multiple Oracle Application Server Wireless middle tiers and Oracle Application Server Wireless schema:

1. All Oracle Application Server Wireless middle tiers in the farm are stopped, including any Oracle Collaboration Suite middle tiers that are configured to run Oracle Application Server Wireless.
2. The first Oracle Application Server Wireless middle tier is selected for upgrade.
3. The first 10*g* (9.0.4) Oracle Application Server Wireless middle tier is installed on the same computer as the Oracle Application Server Wireless middle tier identified in step 2. During installation, the Oracle Application Server Wireless schema in the Oracle Application Server Repository is upgraded.
4. The OracleAS Upgrade Assistant is invoked.

   At this point, all Release 2 (9.0.2) Oracle Application Server Wireless middle tiers can be restarted to reduce downtime.

### A.1.11.2 Oracle Application Server Wireless Upgrade Items (List A)

The following files are changed in the first phase of the Oracle Application Server Wireless upgrade process:

- <*destination_MT_OH*>/wireless/server/classes/*.class
- <*destination_MT_OH*>/wireless/server/classes/*.properties

### A.1.11.3 Oracle Application Server Wireless Upgrade Items (List B)

The following files are changed in the second phase of the Oracle Application Server Wireless upgrade process:

- <*destination_MT_OH*>/opmn/conf/opmn.xml

## A.1.12 The Oracle Application Server Forms Services Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle Application Server Forms Services:

1. Locates the files described in Oracle Application Server Forms Services Upgrade Items (List A, List B, and List C) below.
2. Makes a backup of the files described in Section A.1.12.2, "Oracle Application Server Forms Services Upgrade Items (List A)" on page A-16 with a `preUpgrade` suffix, and copies the files from the source Oracle home to the destination Oracle home without modification.
3. Copies the files described in Section A.1.12.3, "Oracle Application Server Forms Services Upgrade Items (List B)" on page A-16 with a `902` suffix, and copies the files from the source Oracle home to the destination Oracle home.
4. Makes a backup of the files described in Section A.1.12.4, "Oracle Application Server Forms Services Upgrade Items (List C)" on page A-16 with a `preUpgrade` suffix, and extracts customizations from the file in the source Oracle home and merges it into the file in the destination Oracle home.

**A.1.12.1** The following files are changed in the upgrade process:

#### A.1.12.2 Oracle Application Server Forms Services Upgrade Items (List A)

- <*destination_MT_OH*>/forms90/ftrace.cfg
- <*destination_MT_OH*>/forms90/java/oracle/forms/registry/Registry.dat
- User-defined Oracle Application Server Forms Services *.htm files used by formsweb.cfg

#### A.1.12.3 Oracle Application Server Forms Services Upgrade Items (List B)

- <*destination_MT_OH*>/forms90/search_replace.properties
- <*destination_MT_OH*>/forms90/converter.properties

#### A.1.12.4 Oracle Application Server Forms Services Upgrade Items (List C)

- <*destination_MT_OH*>/forms90/server/formsweb.cfg
- <*destination_MT_OH*>/forms90/server/default.env
- <*destination_MT_OH*>/forms90/server/forms90.conf
- <*destination_MT_OH*>/j2ee/properties/oc4j_bi_forms.properties (Oracle Application Server Forms Services deployment entries)
- User-defined Oracle Application Server Forms Services configuration files that are equivalents of formsweb.cfg and default.env.

### A.1.13 The Oracle Application Server Discoverer Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle Application Server Discoverer:

- Merges the changes made to the Pref.txt and .reg_key.dc files in the Release 2 (9.0.2) instance to the equivalent files in the 10*g* (9.0.4) instance.
- Merges the changes made to the *.xml files in the Release 2 (9.0.2) instance to the equivalent files in the 10*g* (9.0.4) instance.
- Creates a backup copy of each configuration file, named as follows:

  <*file name*>_upgrade_backup.<*file suffix*>

### A.1.13.1 Oracle Application Server Discoverer Upgrade Items

The following files are changed in the Oracle Application Server Discoverer upgrade process:

- *<destination_MT_OH>*/util/Pref.txt
- *<destination_MT_OH>*/discoverer/.reg_key.dc
- *<destination_MT_OH>*/j2ee/OC4J_BI_Forms/applications/discoverer/web/WEB-INF/configuration.xml
- *<destination_MT_OH>*/j2ee/OC4J_BI_Forms/applications/discoverer/web/common/xsl/ui_config.xml
- *<destination_MT_OH>*/j2ee/OC4J_BI_Forms/applications/discoverer/web/plus_files/xsl/plus_config.xml

> **Note:** The OracleAS Upgrade Assistant does not upgrade `*.xsl` (style sheet) files. If any of these are customized in the Release 2 (9.0.2) installation, and you want these customizations in the 10*g* (9.0.4) installation, you must re-apply the customizations manually. Style sheet files are listed in Section 3.8.5, "Completing the Oracle Application Server Discoverer Viewer Upgrade" on page 3-58.

## A.1.14 The Oracle Application Server Reports Services Upgrade Process

The OracleAS Upgrade Assistant performs these steps when upgrading Oracle Application Server Reports Services:

- Copies all configuration files except `jdbcpds.conf` from the source Oracle home to the destination Oracle home.
- Copies all resource files from the source Oracle home to the destination Oracle home.
- Copies *<source_MT_OH>*/bin/reports.sh to the destination Oracle home, and adds the `NLS_LANG` environment variable, if it is not in the script.
- Copies reports server persistent files from the source Oracle home to the destination Oracle home.

### A.1.14.1 Oracle Application Server Reports Services Upgrade Items

The following files are changed in the Oracle Application Server Reports Services upgrade process:

- *<destination_MT_OH>*/reports/conf/*.conf (except jdbcpds.conf)
- *<destination_MT_OH>*/reports/conf/*.xml
- *<destination_MT_OH>*/reports/plugins/resource/*.*
- *<destination_MT_OH>*/reports/conf/cgicmd.dat
- *<destination_MT_OH>*/reports/server/*.dat
- *<destination_MT_OH>*/bin/reports.sh
- *<destination_MT_OH>*/reports/conf/rwservlet.properties

---

**Note:** If you used non-default values for the default environment variables, you must manually reset these after the upgrade. See Section 3.8.6, "Completing the Oracle Application Server Reports Services Upgrade" on page 3-60.

---

# A.2 Infrastructure Upgrade Processes

This section describes the upgrade processing of the Infrastructure. Infrastructure components and functionality are described in Chapter 4, "Upgrading the Metadata Repository" and Chapter 5, "Upgrading the Identity Management Services".

It contains the following subsections:



## A.2.1 The Identity Management Upgrade Process

The Identity Management components Oracle Application Server Single Sign-On and Oracle Internet Directory are upgraded by the Oracle Universal Installer. There are two possible configurations upgraded: distributed, in which Oracle Application Server Single Sign-On and Oracle Internet Directory reside on different computers, each with its own metadata repository, and non-distributed, in which they share a metadata repository on one computer.

The upgrade processing is interactive, and performed by the Oracle Universal Installer in each scenario. The Oracle Universal Installer is aware of the configuration being upgraded, and prompts for the necessary information and launches the appropriate configuration tools.

The source and destination configurations are depicted in:

- Figure 5–1, "Non-Distributed Identity Management in Release 2 (9.0.2) and 10g (9.0.4)" on page 5-3
- Figure 5–2, "Distributed Identity Management in Release 2 (9.0.2)" on page 5-4
- Figure 5–3, "Distributed Identity Management in 10g (9.0.4)" on page 5-4

## A.2.2 The Metadata Repository Container Schema Upgrade Process

The Metadata Repository Container schema upgrade adds support for new schemas in the metadata repository, and updates the Oracle Internet Directory entry. The script performs one or both functions, depending on the credentials supplied when it is executed.

> **See Also:** Section 4.2.3.1, "Executing mrc.pl for New Schema Creation" on page 4-6 and Section 4.2.3.2, "Executing mrc.pl for Oracle Internet Directory Entry Update" on page 4-7.

The Metadata Repository Container schema creation upgrade process (the `mrc.pl` script) performs these steps:

1. Checks to determine whether the `ORACLE_HOME` environment variable is set; if not, the process reports an error and exits.
2. Checks to determine whether the Oracle home is a Release 2 (9.0.2) Infrastructure; if it is not, the process reports an error and exits.
3. Connects to the metadata repository database; if unable to connect, the process exits.
4. Checks to determine whether the database version is 9.0.1.5; if not, the process reports an error and exits.
5. Creates these schemas: `wcrsys`, `oca`, `oraoca_public`, `ip`, `wk_test` and `internet_appserver_registry` (with the same password as the user name and corresponding tablespace) and these tablespaces: `ias_meta`, `wcrsys_ts`, `ocats`, `ip_dt`, `ip_rt`, `ip_idx`, `ip_lob` `OLTS_SVRMGSTORE`, `oltsbattrstore`. If the creation of any tablespace or user fails, the process reports an error and continues.
6. Writes message to standard output "Creation of new schemas successful" if all schemas were created successfully.

The Metadata Repository Container Oracle Internet Directory entry upgrade process (the `mrc.pl` script) performs these steps:

1. Checks to determine whether the ORACLE_HOME environment variable is set; if not, the process reports an error and exits.
2. Checks to determine whether the Oracle home is a Release 2 (9.0.2) Infrastructure; if it is not, the process reports an error and exits.
3. Connects to the metadata repository database using the following user names and passwords: oca/oca, oraoca_public, and wcrsys/wcrsys. (These should have been created by the mrc.pl schema creation process that preceded the Oracle Internet Directory entry update.) If unable to connect to all of these users, the process reports an error and exits.
4. Randomizes the password for the users.
5. Creates all required security entries in Oracle Internet Directory and new schema entry, using randomized password.
6. Changes the password of users in metadata repository.
7. Writes message to standard output "Update of new OID entries successful" if creation of OID entries and update of randomized passwords was successful.

### A.2.3 The Process Connect Upgrade Process

The Process Connect upgrade process creates Process Connect schema.

### A.2.4 The Oracle Application Server Certificate Authority Upgrade Process

The Oracle Application Server Certificate Authority schema upgrade process performs these steps:

- Creates OCA and ORAOCA_PUBLIC database users
- Tables and views

### A.2.5 The Oracle Ultra Search Schema Upgrade Process

The Oracle Ultra Search schema upgrade process performs these steps:

- Copies the Oracle Ultra Search 10*g* (9.0.4) binaries to the Infrastructure database directories.
- Connects to repository database as SYS.
- Performs existence check for the WKSYS schema. Gets version number stored in the schema. If the version number is not 9.0.2 (or its patch release version), an error occurs and the process stops.

- Runs the SQL*Plus script `wkdbmig.sql` to upgrade the schema and data in `WKSYS`.
- Executes a `loadjava` command to load Java stored packages into the `WKSYS` schema.
- Runs the SQL*Plus script to create the demo schema and create an Ultra Search instance based on the demo schema.
- Updates the version number in the component registry.
- Configures the Oracle Ultra Search schema information stored in the Oracle Internet Directory.

#### A.2.5.1 Oracle Ultra Search Schema Upgrade Items

The following is changed in the Oracle Ultra Search schema upgrade process:

- `WKSYS` schema

### A.2.6 The OracleAS Portal Schema Upgrade Process

The OracleAS Portal schema upgrade process (the `upgrade.pl` script) performs these steps:

1. Exports a subset of the Oracle9*i*AS Portal 9.0.2 tables to a dump file.
2. Applies an Oracle9*i*AS patch to the existing schema, if necessary.
3. Displays all invalid objects before the upgrade starts. (All Oracle9*i*AS Portal packages must be valid at this stage.)
4. Disables the DBMS jobs in the Oracle9*i*AS Portal schema. These will be re-enabled at the end of the upgrade.
5. Drops statistics gathered on all tables in the Oracle9*i*AS Portal schema. The statistics supported by the Oracle9*i*AS Portal will be gathered near the end of the upgrade.
6. Installs OWA packages, if necessary.

> **Note:** This step is executed only the current version of the OWA packages is not the latest expected version. This step causes invalidation of all packages in the instance that depend on the OWA packages. In particular, most Oracle9*i*AS Portal packages in all Oracle9*i*AS Portal schemas on the instance will become invalid. For this reason, after upgrade, you may need to recompile packages on other affected schemas after the upgrade. The upgrade script only recompiles packages in the Portal schema being upgraded.

7. Drops all existing Java objects from the Portal schema.

> **Note:** If there are any Java objects in the Portal schema that do not belong to Oracle9*i*AS Portal, you must reinstall them manually after the upgrade.

8. Deletes all product messages in all languages.
9. Loads all of the latest product messages in all languages that were previously installed.

> **Note:** If any product messages were altered before the upgrade, you must manually re-apply the changes.

10. Changes the schema and compiles all OracleAS Portal packages.
11. Exports temporary tables created during the upgrade.
12. Recompiles any invalid non-Oracle9*i*AS Portal objects. Warnings are issued if any invalid non-Oracle9*i*AS Portal objects remain after this step.
13. Updates the OracleAS Portal version if there are no fatal compilation errors.
14. Searches for errors and warnings in the upgrade log file and displays those encountered; if there were no errors, the process displays a success message.

### A.2.6.1 OracleAS Portal Schema Upgrade Items

The following schemas are changed in the OracleAS Portal schema upgrade process:

- PORTAL (Portal schema)
- PORTAL_DEMO (Portal demonstration schema)
- PORTAL_PUBLIC (Portal public schema)
- PORTAL_APP (Portal JSP Access)

## A.2.7 The Oracle Application Server Syndication Server Schema Upgrade Process

The Oracle Application Server Syndication Server upgrade script `ossupg902.sql` performs these steps:

- Connects to the database as SYSDBA.
- Performs pre-upgrade validations (such as proper DSGATEWAY version) to determine whether the upgrade can proceed. If any of the checks fail, the upgrade stops.
- Destroys all schema contents, such as tables, PL/SQL packages, etc., in the DSGATEWAY schema.
- Invokes the 10*g* (9.0.4) installation script to install the 10*g* (9.0.4) database schema.

### A.2.7.1 Oracle Application Server Syndication Server Schema Upgrade Items

The following schema is changed in the Oracle Application Server Syndication Server schema upgrade process:

- DSGATEWAY

## A.2.8 The Oracle Application Server Web Services UDDI Registry Schema Upgrade Process

The Oracle Application Server Web Services schema upgrade script `wuru9023.sql` (used for Oracle Application Server Release 2 (9.0.2) with the UDDI 9.0.2.3 patch) performs these steps:

- Connects to repository database.
- Performs pre-upgrade validations (such as proper UDDI version) to determine whether the upgrade can proceed. If any of the checks fail, the upgrade stops.

- Performs initial structural changes, such as creating and dropping tables, columns, and indexes.
- Performs upgrades to the UDDI server properties and configuration, inserting new rows and updating existing rows in the configuration tables.

The Oracle Application Server Web Services schema upgrade helper script `wuru9020.sql` (used for Oracle Application Server Release 2 (9.0.2)) performs these steps:

- Performs pre-upgrade validations (such as proper UDDI version) to determine whether the upgrade can proceed. If any of the checks fail, the upgrade stops.
- Destroys all schema contents, such as tables, PL/SQL packages, etc., in the UDDI database schema.
- Invokes the 9.0.4 installation script to install the 10*g* (9.0.4) database schema.

### A.2.8.1 Oracle Application Server Web Services Schema Upgrade Items

The following is changed in the Oracle Application Server Web Services schema upgrade process:

- `UDDISYS` schema.

---

**Note:** The UDDI middle tier upgrade must precede the UDDI schema upgrade.

---

## A.2.9 The Web Clipping Upgrade Process

Web Clipping is a new component, so there are no dependencies on other upgrade processes in the middle tier or Infrastructure. The upgrade process creates new tables and constraints, defines a package with functions and procedures, and seeds the tables with randomized data.

### A.2.9.1 Web Clipping Schema Upgrade Items

The `WCRSYS` schema is created in the Web Clipping schema upgrade process.

## A.2.10 The Oracle Application Server Wireless Schema Upgrade Process

The Oracle Application Server Wireless schema in the Oracle9iAS Metadata Repository is automatically upgraded during installation of the first Oracle Application Server Wireless 10*g* (9.0.4) middle tier. You do not have to perform any additional upgrade steps on the Oracle9iAS Infrastructure. See Section A.1.11.1, "Upgrade of Oracle Application Server Wireless Middle Tiers and Wireless Schema" on page A-14 for details.

# B

# Files Reference

This appendix lists and describes files that contain settings and configuration data that is transferred from the source to the target Oracle home in an upgrade.

The Upgrade Type column indicates the method of upgrading the file. In some cases, the OracleAS Upgrade Assistant processes the file, copying the *majority* of, if not all of, the customizations made to it to the 10*g* (9.0.4) Oracle Application Server instance. (Exceptions such as port numbers, mount points, and other configuration settings that may require adjustment after the upgrade are identified in the guide; you may need to alter the file after upgrade.)

Files of the Manual upgrade type are not processed by the OracleAS Upgrade Assistant. Customizations made to these files in the source instance must be transferred manually to the equivalent files in the destination instance.

A text description of each file follows the tables:

Table B–1, "Files Containing Upgrade Data (Sorted by File)"

Table B–2, "Files Containing Upgrade Data (Sorted by Path)"

Upgraded and created schemas are listed in Table B–3, "Infrastructure Upgrade (Schema) Processes"

*Table B–1 Files Containing Upgrade Data (Sorted by File)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| *.* | reports/plugins/resource | OracleAS Upgrade Assistant |
| *.* | webcache/wallets | OracleAS Upgrade Assistant |
| *.class | wireless/server/classes | OracleAS Upgrade Assistant |
| *.conf (except jdbcpds.conf) | reports/conf | OracleAS Upgrade Assistant |
| *.dat | reports/server | OracleAS Upgrade Assistant |
| *.htm | forms90/server | OracleAS Upgrade Assistant |
| *.properties | wireless/server/classes | OracleAS Upgrade Assistant |
| *.xml | reports/conf | OracleAS Upgrade Assistant |
| .reg_key.dc | discoverer902/bin/.reg_key.dc (which upgrades to discoverer/.reg_key.dc) | OracleAS Upgrade Assistant |
| application-client.xml | j2ee/<name of OC4J instance>/config/application-client.xml | Manual |
| application.xml | j2ee/<name of OC4J instance>/config/application.xml<br>j2ee/home/applications/<name of OC4J application>/META-INF/application.xml | Manual |
| cache.conf | Apache/modplsql/conf/cache.conf | OracleAS Upgrade Assistant |

*Table B–1 Files Containing Upgrade Data (Sorted by File)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| cgimd.dat | `reports/conf/cgimd.dat` | OracleAS Upgrade Assistant |
| configuration.xml | `j2ee/OC4J_BI_Forms/applications/discoverer/web/WEB-INF/configuration.xml` | OracleAS Upgrade Assistant |
| converter.properties | `forms90/converter.properties` | OracleAS Upgrade Assistant |
| dads.conf | `Apache/modplsql/conf/dads.conf` (ORASSO DAD upgrade) | Manual |
| dads.conf | `Apache/modplsql/conf/dads.conf` | OracleAS Upgrade Assistant |
| data-sources.xml | `j2ee/OC4J_Portal/config/data-sources.xml` and `j2ee/<name of OC4J instance>/config/data-sources.xml` | OracleAS Upgrade Assistant |
| default-web-site.xml | `j2ee/<name of OC4J instance>/config/default-web-site.xml` | Manual |
| default.env (or user-defined environment file) | `forms90/server/default.env` | OracleAS Upgrade Assistant |
| dms.conf | `Apache/Apache/conf/dms.conf` | Manual |
| EAR files for applications defined in the server.xml file in the source instance | `j2ee/<name of OC4J instance>/applications/*.ear` | OracleAS Upgrade Assistant |
| forms90.conf | `forms90/server/forms90.conf` | OracleAS Upgrade Assistant |
| formsweb.cfg (or user-defined servlet configuration file) | `forms90/server/formsweb.cfg` | OracleAS Upgrade Assistant |
| ftrace.cfg | `forms90/trace/ftrace.cfg` | OracleAS Upgrade Assistant |
| global-web-application.xml | `j2ee/<name of OC4J instance>/config/global-web-application.xml` | Manual |

*Table B–1 Files Containing Upgrade Data (Sorted by File)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| http-web-site.xml | j2ee/<name of OC4J instance>/config/ http-web-site.xml | Manual |
| httpd.conf | Apache/Apache/conf/httpd.conf | OracleAS Upgrade Assistant |
| ias.properties | config/ias.properties | Manual |
| iasschema.xml | config/iaschema.xml | OracleAS Upgrade Assistant |
| index.html | Apache/Apache/htdocs/index.html | Manual |
| internal.xml | webcache/internal.xml | Manual |
| jazn-data.xml | j2ee/home/config/jazn-data.xml and j2ee/home/application-deployments/ <name of application>/jazn-data.xml | OracleAS Upgrade Assistant |
| jazn-data.xml | sysman/j2ee/config/jazn-data.xml | Manual |
| jazn.xml | j2ee/home/config/jazn.xml | OracleAS Upgrade Assistant |
| jdbcpds.conf | reports/conf/jdbcpds.conf | Manual |
| jms.xml | j2ee/home/config/jms.xml | Manual |
| mod_oc4j.conf | Apache/Apache/conf/mod_oc4j.conf | OracleAS Upgrade Assistant |
| mod_osso.conf | Apache/Apache/conf/mod_osso.conf | OracleAS Upgrade Assistant |
| moddav.conf | Apache/Apache/oradav/conf/moddav.conf | OracleAS Upgrade Assistant |
| oc4j.properties | j2ee/oc4j.properties | OracleAS Upgrade Assistant |
| oc4j_bi_forms.properties | j2ee/properties/oc4j_bi_forms.properties | OracleAS Upgrade Assistant |

*Table B–1 Files Containing Upgrade Data (Sorted by File)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| opmn.xml | opmn/conf/opmn.xml | OracleAS Upgrade Assistant |
| oracle_apache.conf | Apache/Apache/conf/oracle_apache.conf | Manual |
| oradav.conf | Apache/oradav/conf/oradav.conf | OracleAS Upgrade Assistant |
| orion-application.xml | j2ee/home/application-deployments/<name of application>/orion-application.xml | Manual |
| orion-web.xml | j2ee/home/application-deployments/<name of application>/orion-web.xml | Manual |
| plsql.conf | Apache/Apache/conf/osso/osso.conf | OracleAS Upgrade Assistant |
| plsql.conf | Apache/modplsql/conf/plsql.conf | Manual |
| plus_config.xml | j2ee/OC4J_BI_Forms/applications/discoverer/web/plus_files/xsl/plus_config.xml | OracleAS Upgrade Assistant |
| Pref.txt | discoverer902/util/Pref.txt (which upgrades to discoverer/util/Pref.txt) | OracleAS Upgrade Assistant |
| principals.xml | j2ee/<name of OC4J instance>/config/principals.xml | OracleAS Upgrade Assistant |
| provideruiacls.xml | j2ee/OC4J_Portal/applications/jpdk/jpdk/WEB-INF/deployment_providerui/provideruiacls.xml and j2ee/OC4J_Portal/applications/portalTools/providerBuilder/WEB-INF/deployment_providerui/provideruiacls.xml | Manual |
| proxyinfo.xml | reports/conf/proxyinfo.xml | OracleAS Upgrade Assistant |
| Registry.dat | forms90/java/oracle/forms/registry/Registry.dat | OracleAS Upgrade Assistant |

*Table B–1 Files Containing Upgrade Data (Sorted by File)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| reports.sh | bin/reports.sh | OracleAS Upgrade Assistant |
| rmi.xml | j2ee/home/config/rmi.xml | Manual |
| rwcgi.sh | bin/rwcgi.sh | Manual |
| rwclient.sh | bin/rwclient.sh | Manual |
| rwconverter.sh | bin/rwconverter.sh | Manual |
| rwengine.sh | bin/rwengine.sh | Manual |
| rwgenkey.sh | bin/rwgenkey.sh | Manual |
| rwproxy.sh | bin/rwproxy.sh | Manual |
| rwrqv.sh | bin/rwrgv.sh | Manual |
| rwrun.sh | bin/rwrun.sh | Manual |
| rwserver.sh | bin/rwserver.sh | Manual |
| rwservlet.properties | reports/conf/rwservlet.properties | OracleAS Upgrade Assistant |
| search_replace.properties | forms90/search_replace.properties | OracleAS Upgrade Assistant |
| server.xml | j2ee/home/config/server.xml | Manual |
| targets.xml | sysman/emd/targets.xml | OracleAS Upgrade Assistant |
| textpds.conf | reports/conf/textpds.conf | OracleAS Upgrade Assistant |
| tnsnames.ora | network/admin/tnsnames.ora | Manual |
| uddiserver.config | ds/uddi/config/uddiserver.config (which upgrades to uddi/config/uddiserver.config) | OracleAS Upgrade Assistant |
| ui_config.xml | j2ee/OC4J_BI_ Forms/applications/discoverer/web/common/xsl | OracleAS Upgrade Assistant |
| web.xml | j2ee/home/default-web-app/WEB-INF/web.xml | Manual |

*Table B–1 Files Containing Upgrade Data (Sorted by File)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| web.xml | j2ee/OC4J_Portal/ applications/portal/portal/WEB-INF/web.xml | Manual |
| webcache.xml | webcache/webcache.xml | OracleAS Upgrade Assistant |
| xmlpds.conf | reports/conf/xmlpds.conf | OracleAS Upgrade Assistant |

*Table B–2 Files Containing Upgrade Data (Sorted by Path)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| dms.conf | Apache/Apache/conf/dms.conf | Manual |
| httpd.conf | Apache/Apache/conf/httpd.conf | OracleAS Upgrade Assistant |
| mod_oc4j.conf | Apache/Apache/conf/mod_oc4j.conf | OracleAS Upgrade Assistant |
| mod_osso.conf | Apache/Apache/conf/mod_osso.conf | OracleAS Upgrade Assistant |
| oracle_apache.conf | Apache/Apache/conf/oracle_apache.conf | Manual |
| plsql.conf | Apache/Apache/conf/osso/osso.conf | OracleAS Upgrade Assistant |
| index.html | Apache/Apache/htdocs/index.html | Manual |
| moddav.conf | Apache/Apache/oradav/conf/moddav.conf | OracleAS Upgrade Assistant |
| cache.conf | Apache/modplsql/conf/cache.conf | OracleAS Upgrade Assistant |
| dads.conf | Apache/modplsql/conf/dads.conf | OracleAS Upgrade Assistant |

*Table B–2 Files Containing Upgrade Data (Sorted by Path)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| dads.conf | `Apache/modplsql/conf/dads.conf` (ORASSO DAD upgrade) | Manual |
| plsql.conf | `Apache/modplsql/conf/plsql.conf` | Manual |
| oradav.conf | `Apache/oradav/conf/oradav.conf` | OracleAS Upgrade Assistant |
| reports.sh | `bin/reports.sh` | OracleAS Upgrade Assistant |
| rwcgi.sh | `bin/rwcgi.sh` | Manual |
| rwclient.sh | `bin/rwclient.sh` | Manual |
| rwconverter.sh | `bin/rwconverter.sh` | Manual |
| rwengine.sh | `bin/rwengine.sh` | Manual |
| rwgenkey.sh | `bin/rwgenkey.sh` | Manual |
| rwproxy.sh | `bin/rwproxy.sh` | Manual |
| rwrqv.sh | `bin/rwrgv.sh` | Manual |
| rwrun.sh | `bin/rwrun.sh` | Manual |
| rwserver.sh | `bin/rwserver.sh` | Manual |
| ias.properties | `config/ias.properties` | Manual |
| iasschema.xml | `config/iaschema.xml` | OracleAS Upgrade Assistant |
| .reg_key.dc | `discoverer902/bin/.reg_key.dc` (which upgrades to `discoverer/.reg_key.dc`) | OracleAS Upgrade Assistant |
| Pref.txt | `discoverer902/util/Pref.txt` (which upgrades to `discoverer/util/Pref.txt`) | OracleAS Upgrade Assistant |
| uddiserver.config | `ds/uddi/config/uddiserver.config` (which upgrades to `uddi/config/uddiserver.config`) | OracleAS Upgrade Assistant |
| converter.properties | `forms90/converter.properties` | OracleAS Upgrade Assistant |

*Table B–2 Files Containing Upgrade Data (Sorted by Path)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| Registry.dat | forms90/java/oracle/forms/registry/Registry.dat | OracleAS Upgrade Assistant |
| search_replace.properties | forms90/search_replace.properties | OracleAS Upgrade Assistant |
| *.htm | forms90/server | OracleAS Upgrade Assistant |
| default.env (or user-defined environment file) | forms90/server/default.env | OracleAS Upgrade Assistant |
| forms90.conf | forms90/server/forms90.conf | OracleAS Upgrade Assistant |
| formsweb.cfg (or user-defined servlet configuration file) | forms90/server/formsweb.cfg | OracleAS Upgrade Assistant |
| ftrace.cfg | forms90/trace/ftrace.cfg | OracleAS Upgrade Assistant |
| EAR files for applications defined in the server.xml file in the source instance | j2ee/<name of OC4J instance>/applications/*.ear | OracleAS Upgrade Assistant |
| application-client.xml | j2ee/<name of OC4J instance>/config/application-client.xml | Manual |
| default-web-site.xml | j2ee/<name of OC4J instance>/config/default-web-site.xml | Manual |
| global-web-application.xml | j2ee/<name of OC4J instance>/config/global-web-application.xml | Manual |
| http-web-site.xml | j2ee/<name of OC4J instance>/config/http-web-site.xml | Manual |
| application.xml | j2ee/<name of OC4J instance>/config/application.xml<br>j2ee/home/applications/<name of OC4J application>/META-INF/application.xml | Manual |

*Table B–2 Files Containing Upgrade Data (Sorted by Path)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| principals.xml | j2ee/<name of OC4J instance>/ config/principals.xml | OracleAS Upgrade Assistant |
| orion-application.xml | j2ee/home/application-deployments/ <name of application>/orion-application.xml | Manual |
| orion-web.xml | j2ee/home/application-deployments/ <name of application>/orion-web.xml | Manual |
| jazn-data.xml | j2ee/home/config/jazn-data.xml and j2ee/home/application-deployments/ <name of application>/jazn-data.xml | OracleAS Upgrade Assistant |
| jazn.xml | j2ee/home/config/jazn.xml | OracleAS Upgrade Assistant |
| jms.xml | j2ee/home/config/jms.xml | Manual |
| rmi.xml | j2ee/home/config/rmi.xml | Manual |
| server.xml | j2ee/home/config/server.xml | Manual |
| web.xml | j2ee/home/default-web-app/WEB-INF/web.xml | Manual |
| oc4j.properties | j2ee/oc4j.properties | OracleAS Upgrade Assistant |
| ui_config.xml | j2ee/OC4J_BI_ Forms/applications/discoverer/web/common/xsl | OracleAS Upgrade Assistant |
| plus_config.xml | j2ee/OC4J_BI_Forms/applications/discoverer/web/ plus_files/xsl/plus_config.xml | OracleAS Upgrade Assistant |
| configuration.xml | j2ee/OC4J_BI_ Forms/applications/discoverer/web/WEB-INF/ configuration.xml | OracleAS Upgrade Assistant |
| provideruiacls.xml | j2ee/OC4J_Portal/applications/jpdk/jpdk/ WEB-INF/deployment_providerui/provideruiacls.xml and j2ee/OC4J_ Portal/applications/portalTools/providerBuilder/W EB-INF/deployment_providerui/provideruiacls.xml | Manual |
| web.xml | j2ee/OC4J_Portal/ applications/portal/portal/WEB-INF/web.xml | Manual |

*Table B–2 Files Containing Upgrade Data (Sorted by Path)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| data-sources.xml | j2ee/OC4J_Portal/config/data-sources.xml and j2ee/<name of OC4J instance>/config/data-sources.xml | OracleAS Upgrade Assistant |
| oc4j_bi_forms.properties | j2ee/properties/oc4j_bi_forms.properties | OracleAS Upgrade Assistant |
| tnsnames.ora | network/admin/tnsnames.ora | Manual |
| opmn.xml | opmn/conf/opmn.xml | OracleAS Upgrade Assistant |
| *.conf (except jdbcpds.conf) | reports/conf | OracleAS Upgrade Assistant |
| *.xml | reports/conf | OracleAS Upgrade Assistant |
| cgimd.dat | reports/conf/cgimd.dat | OracleAS Upgrade Assistant |
| jdbcpds.conf | reports/conf/jdbcpds.conf | Manual |
| proxyinfo.xml | reports/conf/proxyinfo.xml | OracleAS Upgrade Assistant |
| rwservlet.properties | reports/conf/rwservlet.properties | OracleAS Upgrade Assistant |
| textpds.conf | reports/conf/textpds.conf | OracleAS Upgrade Assistant |
| xmlpds.conf | reports/conf/xmlpds.conf | OracleAS Upgrade Assistant |
| *.* | reports/plugins/resource | OracleAS Upgrade Assistant |

*Table B–2 Files Containing Upgrade Data (Sorted by Path)*

| File | Path from Oracle Home | Upgrade Type |
|---|---|---|
| *.dat | reports/server | OracleAS Upgrade Assistant |
| targets.xml | sysman/emd/targets.xml | OracleAS Upgrade Assistant |
| jazn-data.xml | sysman/j2ee/config/jazn-data.xml | Manual |
| internal.xml | webcache/internal.xml | Manual |
| *.* | webcache/wallets | OracleAS Upgrade Assistant |
| webcache.xml | webcache/webcache.xml | OracleAS Upgrade Assistant |
| *.class | wireless/server/classes | OracleAS Upgrade Assistant |
| *.properties | wireless/server/classes | OracleAS Upgrade Assistant |

***.***

The contents of the reports / plugins / resource directory. These files are configuration files for Oracle Application Server Reports Services.

***.***

The contents of the webcache / wallets directory. These files are wallet files for Oracle Application Server Web Cache.

***.class**

These files are class files for Oracle Application Server Wireless applications.

***.conf**

These files are user-defined files for Oracle Application Server Reports Services.

***.dat**

These files are user-defined files for Oracle Application Server Reports Services.

***.htm**

These files are user-defined files for deploying Forms applet clients.

***.properties**

These files are properties files for Oracle Application Server Wireless applications.

***.xml**

These files are user-defined files for Oracle Application Server Reports Services.

**.reg_key.dc**

The .reg_key.dc file is the internal registry for storing default user preferences and system preferences for Oracle Application Server Discoverer. The preferences stored in Pref.txt are applied to this registry.

**application-client.xml**

The application-client.xml file contains configuration information used across all J2EE servers.

**application.xml**

The application.xml file is the deployment descriptor for enterprise application archives.

**cache.conf**

The cache.conf file specifies the characteristics of the mod_plsql caching system. There are two types of caches in use:

PLSQL Cache: Caches dynamically generated content that doesn't change often. Applications using the OWA_CACHE package, such as OracleAS Portal, use this feature to improve performance and reduce the overhead associated with database requests.

Session cookie cache: Caches the cookie value generated by the Oracle Application Server Single Sign-On server for a particular session, thereby increasing performance by avoiding a round trip to the database to obtain user credentials. Applications using Oracle Application Server Single Sign-On benefit from this feature.

### cgimd.dat

The cgimd.dat file is a configuration file for Oracle Application Server Forms Services.

### configuration.xml

The configuration.xml file contains configuration information for the Oracle Application Server Discoverer Viewer and Portlet Provider servlets.

### converter.properties

The converter.properties file is a Forms configuration file that defines such options as log file name, Reports servlet directory, and Reports server host.

### dads.conf

The dads.conf file is a mod_plsql configuration file that contains Database Access Descriptor (DAD) entries.

### data-sources.xml

The data-sources.xml file contains data source configuration information for databases used by OC4J applications.

### dms.conf

The dms.conf file is the configuration file for the Dynamic Monitoring Service.

### default-web-site.xml

The default-web-site.xml file contains the configuration for the default OC4J Web site for Oracle Application Server.

### default.env

The default.env file contains environment variables used by an Oracle Application Server Forms Services runtime process. A user-defined environment file may be used instead of default.env.

### forms90.conf

The forms90.conf file contains the Oracle Application Server Forms Services configuration settings for the Oracle HTTP Server.

### formsweb.cfg

The formsweb.cfg file is the Oracle Application Server Forms Services servlet default configuration file. It defines the parameter values used by the servlet. A user-defined servlet configuration file may be used instead of formsweb.cfg.

### ftrace.cfg

The ftrace.cfg file specifies event groups for use with the Forms Trace feature in Oracle Application Server Forms Services.

### global-web-application.xml

The global-web-application.xml file is the main Oracle Application Server Containers for J2EE (OC4J) deployment file. It is a global configuration file for web applications; it contains settings that are inherited by other deployments.

### http-web-site.xml

The http-web-site.xml file is the default web site configuration file for Oracle Application Server Containers for J2EE (OC4J).

### httpd.conf

The httpd.conf file is the main configuration file for the Oracle HTTP Server.

### ias.properties

The ias.properties file contains settings for the Oracle Application Server instance. The component configuration section lists the components and their launch success value. The installation data section lists the version, installation type, components included, instance name, and infrastructure host and port information. The infrastructure database section lists database names.

### iasschema.xml

The iasschema.xml file contains a schema configuration entry for each Oracle Application Server components. The entry contains component names, schema name and database connection information.

### index.html

The index.html file is the default home page for Oracle Application Server. It provides links to documentation, demonstrations, and summaries of new features.

**internal.xml**

The internal.xml file contains internal configuration settings for Oracle Application Server Web Cache.

**internal_admin.xml**

The internal_admin.xml file contains internal configuration settings for Oracle Application Server Web Cache.

**jazn-data.xml**

The jazn-data.xml file is a configuration file for the Java Authentication and Authorization Service (JAAS). It contains JAAS data on users, roles, policies, and LoginModules. It is used only when the XML provider type is specified.

**jazn.xml**

The jazn.xml file is a configuration file for the Java Authentication and Authorization Service (JAAS). It is used when the LDAP/OID or XML provider type is specified.

**jdbcpds.conf**

The jdbcpds.conf file is a Oracle Application Server Reports Services configuration file for JDBC drivers. It is pre-configured for the drivers provided with the installation, but you can add entries for any other drivers you want to use.

**jms.xml**

The jms.xml file contains configuration for the Java Messaging Service subsystem, such as: the hostname or IP address and port the JMS server binds to, queues and topics to be bound in the JNDI tree, and log settings.

**moddav.conf**

The moddav.conf file configures and loads the mod_oradav module, which enables distributed authoring and versioning of Web documents.

**mod_oc4j.conf**

The mod_oc4j.conf file contains mount points for routing requests from the Oracle HTTP Server to OC4J instances.

### mod_osso.conf

The mod_osso.conf file is the configuration file forOracle Application Server Single Sign-On

### oc4j_bi_forms.properties

The oc4j_bi_forms.properties file is the Oracle Application Server Containers for J2EE deployment file for the Business Intelligence and Forms installation type.

### oc4j.properties

The oc4j.properties file contains application-specific properties for an OC4J instance.

### opmn.xml

The opmn.xml file is the Oracle Process Management and Notification Service (OPMN) configuration file. It identifies the Oracle Application Server Containers for J2EE (OC4J) and Oracle HTTP Server (OHS) processes that OPMN will start and manage.

### oracle_apache.conf

The oracle_apache.conf file contains custom configuration settings for the Oracle HTTP Server.

### oradav.conf

The oradav.conf file contains the OraDAV configuration parameters for Portal access. The httpd.conf file contains an Include directive for the oradav.conf file.

### orion-application.xml

The orion-application.xml file is the Oracle Application Server Containers for J2EE (OC4J) application deployment file. It is generated during deployment, and specifies whether to automatically create and delete tables for Container-Managed Persistence (CMP) beans, the default data source for CMP beans, security role mappings, the user manager, and Java Naming and Directory Interface (JNDI) namespace access.

### orion-web.xml

The orion-web.xml file is the Oracle Application Server Containers for J2EE (OC4J) web application deployment file. It is generated during deployment, and contains settings such as buffering, character sets, document root, locales, session tracking, and Java Naming and Directory Interface (JNDI) mappings.

**osso.conf**

The osso.conf file is the configuration file for the Oracle Application Server Single Sign-On server.

**plsql.conf**

The plsql.conf file is the configuration file for Oracle HTTP Server module mod_ plsql.

**plus_config.xml**

The plus_config.xml file contains user interface customizations information for the Oracle Application Server Discoverer Plus client, including transports for communication between the applet and the RMI server.

**Pref.txt**

The pref.txt file contains default user preferences and system preferences for Oracle Application Server Discoverer.

**principals.xml**

The principals.xml file contains a set of users, groups permissions and certificates.

**provideruiacls.xml**

The provideruiacls.xml file contains security settings for Portal Development Kit Services for Java (JPDK) web providers.

**proxyinfo.xml**

The proxyinfo.xml file is anOracle Application Server Reports Servicesfile thatcontains information on a proxy server, if one is used. This file is specified in the `destination` element in the mainOracle Application Server Reports Servicesconfiguration file.

**Registry.dat**

The Registry.dat file contains customizations for font mapping and the icon images path in Oracle Application Server Forms Services.

**reports.sh**

The reports.sh file is an Oracle Application Server Reports Services script.

**rmi.xml**

The rmi.xml file contains configuration that allows Remote Method Invocation (RMI), allowing other servers to access the server for RMI requests.

**rwcgi.sh**

The rwcgi.sh file is an Oracle Application Server Reports Services script.

**rwclient.sh**

The rwclient.sh file is an Oracle Application Server Reports Services script.

**rwconverter.sh**

The rwconverter.sh file is an Oracle Application Server Reports Services script.

**rwengine.sh**

The rwengine.sh file is an Oracle Application Server Reports Services script.

**rwgenkey.sh**

The rwgenkey.sh file is an Oracle Application Server Reports Services script.

**rwproxy.sh**

The rwproxy.sh file is an Oracle Application Server Reports Services script.

**rwrqv.sh**

The rwrgv.sh file is an Oracle Application Server Reports Services script.

**rwrun.sh**

The rwrun.sh file is an Oracle Application Server Reports Services script.

**rwserver.sh**

The rwserver.sh file is an Oracle Application Server Reports Services script.

**rwservlet.properties**

The rwservlet.properties file contains properties for Oracle Application Server Reports Services.

**search_replace.properties**

The search_replace properties file contains search and replace strings and built-in classes for the Oracle Application Server Forms Services Migration Assistant.

**server.xml**

The server.xml file contains the configuration for the Oracle Application Server Containers for J2EE (OC4J) instance.

**targets.xml**

The targets.xml file contains representations of resources accessed from the Oracle Enterprise Manager Web site, such as Oracle Application Server Single Sign-On and OracleAS Portal.

**textpds.conf**

The textpds.conf file is anOracle Application Server Reports Servicesfile thatcontains configuration information for the text pluggable data source. It is used to configure the format (delimiter, columns, data type) of text files used as data sources for reports.

**tnsnames.ora**

The tnsnames.ora file contains entries that describe the locations of Oracle databases, Application server components and other Oracle products use the entries in this file to connect to databases.

**uddiserver.config**

The uddiserver.config file contains configuration for the Web Services registry.

**ui_config.xml**

The ui_config.xml file contains user interface customizations information for Oracle Application Server Discoverer Viewer.

**web.xml**

The web.xml file is the J2EE deployment descriptor for a web application.

**webcache.xml**

The webcache.xml file is the configuration file for Web Cache.

### xmlpds.conf

The xmlpds.conf file is anOracle Application Server Reports Servicesfile that contains configuration information for the XML pluggable data source.

***Table B–3 Infrastructure Upgrade (Schema) Processes***

| Upgrade Process | Upgrade Item |
|---|---|
| Section A.2.2, "The Metadata Repository Container Schema Upgrade Process" | ■ OCA (created)<br>■ ORAOCA_PUBLIC (created)<br>■ WCRSYS (created)<br>■ IP (created)<br>■ WK_TEST (created)<br>■ INTERNET_APPSERVER_REGISTRY |
| Section A.2.6, "The OracleAS Portal Schema Upgrade Process" | ■ PORTAL (OracleAS Portal schema)<br>■ PORTAL_DEMO (OracleAS Portal demonstration schema)<br>■ PORTAL_PUBLIC (OracleAS Portal public schema)<br>■ PORTAL_APP (Portal JSP Access) |
| Section A.2.5, "The Oracle Ultra Search Schema Upgrade Process" | ■ WKSYS schema |
| Section A.2.8, "The Oracle Application Server Web Services UDDI Registry Schema Upgrade Process" | ■ UDDISYS schema |
| Section A.2.7, "The Oracle Application Server Syndication Server Schema Upgrade Process" | ■ DSGATEWAY schema |

# Index

## Symbols



## Numerics



## A



## B



## C

## D



## E



## F

## H



## I



## J



## L

## M



## N



## O

## P

## Q



## R



## S



## T

## U



## V



## W



## X