Oracle® Application Server

Upgrade and Compatibility Guide

10*g* Release 3 (10.1.3.1.0)

B31015-01

October 2006

ORACLE®

Oracle Application Server Upgrade and Compatibility Guide, 10*g* Release 3 (10.1.3.1.0)

B31015-01

Primary Author: Peter LaQuerre

Contributing Author: Caroline Johnston

Contributors: Janga Aliminati, Bryan Atsatt, Virginia Beecher, Rao Bhethanabotla, Steve Button, Rachel Chan, Alfred Franci, Viresh Garg, Tugdual Grall, Helen Grembowicz, Dan Hynes, Viral Kamdar, Kishore Kumar, Michael Lehmann, Rajiv Maheshwari, Robert Nettleton, Debu Panda, Paul Parkinson, Thomas Pfaeffle, Mary Beth Roeser, Ekkehard Rohwedder, Dana Singleterry, J.J. Snyder, Jeff Steidl, Deborah Steiner, Jayaram Swaminathan, Sitaraman Swaminathan, Brian Wright, Frances Zhao

Contents

5 Taking Advantage of the 10*g* Release 3 (10.1.3.1.0) SOA Components

A Step-By-Step Upgrade Examples

B Differences Between 10*g* Release 3 (10.1.3.1.0) and Previous Releases

C Things to Consider When Redeploying 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) Applications

Index

Preface

This preface contains the following sections:

Audience

This document is intended for Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) administrators who want to:

- Understand the features and advantages of upgrading their Oracle Application Server J2EE environment to Oracle Application Server 10*g* Release 3 (10.1.3.1.0)
- Redeploy their 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) J2EE applications in 10*g* Release 3 (10.1.3.1.0)
- Integrate 10*g* Release 3 (10.1.3.1.0) middle tiers into their existing 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) application server environment
- Understand any compatibility issues between Oracle Application Server 10*g* (9.0.4), 10*g* Release 2 (10.1.2), and 10*g* Release 3 (10.1.3.1.0)

> **Note:** If you are currently using Oracle Application Server 10*g* Release 3 (10.1.3.0.0), refer to the *Oracle Application Server Patchset Notes 10g Release 3 (10.1.3.1.0) Patch Set 1 (10.1.3.1.0) for Linux and Microsoft Windows*.
>
> The 10*g* Release 3 (10.1.3.1.0) Patchset Notes are available as part of the patchset download from Oracle*MetaLink* and provide information about applying the 10*g* Release 3 (10.1.3.1.0) patchset to your existing 10*g* Release 3 (10.1.3.0.0) Oracle homes.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to

evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

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

Related Documents

For more information, refer the following documents in the Oracle Application Server 10*g* Release 3 (10.1.3.1.0) documentation library:

- *Oracle Application Server Installation Guide* for your platform
- *Oracle Application Server Administrator's Guide*
- *Oracle Containers for J2EE Configuration and Administration Guide*
- *Oracle Application Server Enterprise Deployment Guide*
- *Oracle SOA Suite Developer's Guide*

Conventions

The following text conventions are used in this document:

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

1

Overview of the 10*g* Release 3 (10.1.3.1.0) Upgrade Process

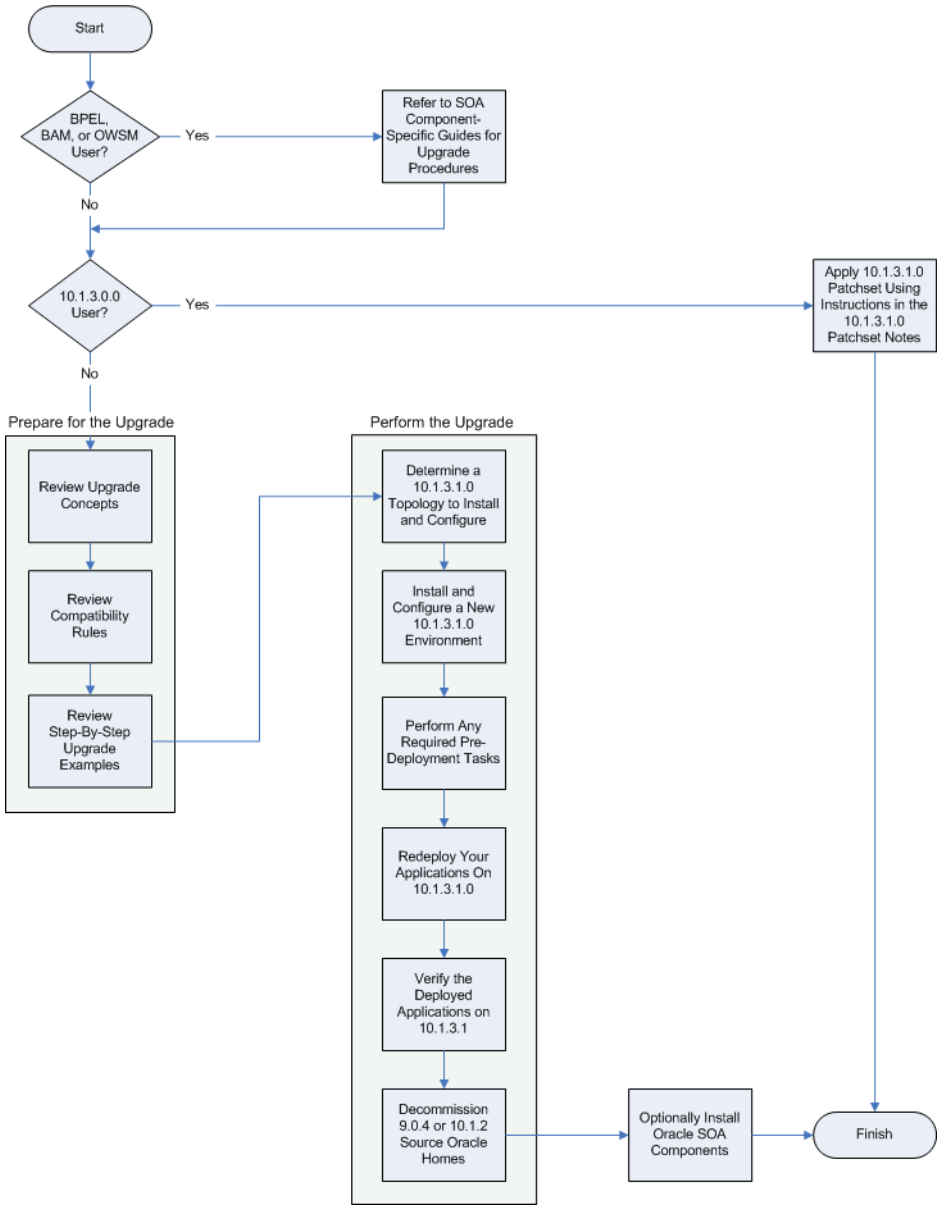
This chapter provides a high-level overview of the 10*g* Release 3 (10.1.3.1.0) upgrade process. Refer to the following sections for more information:

- Flow Chart of the 10g Release 3 (10.1.3.1.0) Upgrade Process
- Table Describing the Steps in the 10g Release 3 (10.1.3.1.0) Upgrade Process

1.1 Flow Chart of the 10*g* Release 3 (10.1.3.1.0) Upgrade Process

Figure 1–1 provides a flow chart of the 10*g* Release 3 (10.1.3.1.0) upgrade process. Review this chart to get familiar with the steps you will be required to take, based on your existing version of Oracle Application Server.

Figure 1–1 Flow Chart of the 10g Release 3 (10.1.3.1.0) Upgrade Process



1.2 Table Describing the Steps in the 10*g* Release 3 (10.1.3.1.0) Upgrade Process

Table 1–1 describes each of the steps in the upgrade process flow chart, which is shown in Figure 1–1. The table also provides information on where to get more information on each step in the process.

Table 1–1 Description of Steps in the 10g Release 3 (10.1.3.1.0) Upgrade Process

Step	Description	More Information
BPEL, BAM, or OWSM User?	Some components of the 10*g* Release 3 (10.1.3.1.0) Oracle SOA Suite, such as Oracle BPEL Process Manager, Oracle Business Activity Monitoring, and Oracle Web Services Manager, were previously available as standalone products. In some cases, special instructions apply if you are upgrading from a previous version of a Oracle SOA Suite component.	Section 5.4, "Where to Learn About Upgrading From Previous Versions of the SOA Components"
Refer to SOA Component-Specific Guides for Upgrade Procedures	If you are using a previous version of the Oracle SOA Suite components, you should refer to the component-specific documentation for information on upgrading to 10*g* Release 3 (10.1.3.1.0).	Section 5.4, "Where to Learn About Upgrading From Previous Versions of the SOA Components"
10.1.3.0.0 User?	The procedure you use to upgrade your environment varies significantly, based on whether or not you are upgrading from 10*g* Release 3 (10.1.3.0.0).	Chapter 2, "Upgrade Concepts" for information about the new features provided for 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0)
Apply 10.1.3.1 Patchset Using Instructions in the 10.1.3.1.0 Patchset Notes	If you are upgrading from 10*g* Release 3 (10.1.3.0.0), then you apply the 10*g* Release 3 (10.1.3.1.0) patchset to your existing 10*g* Release 3 (10.1.3.0.0) Oracle home directories.	The *Oracle Application Server Patchset Notes 10g Release 3 (10.1.3.1.0) Patch Set 1 (10.1.3.1.0) for Linux and Microsoft Windows*, which is available as part of the patchset on Oracle*MetaLink*.
Review Upgrade Concepts	Before you begin, make sure you are familiar with the basic concepts of the Oracle Application Server upgrade, including typical upgrade scenarios and the differences you may encounter when upgrading to the most recent release.	Chapter 2, "Upgrade Concepts"
Review Compatibility Rules	Make sure your upgraded 10*g* Release 3 (10.1.3.1.0) components will be able to work with any other Oracle Application Server components that already exist in your environment.	Chapter 3, "10g Release 3 (10.1.3.1.0) Version Compatibility"
Review Step-By-Step Upgrade Examples	Review the provided step-by-step upgrade examples to learn about the overall upgrade process. The examples show how a very specific application deployment can be upgraded to 10*g* Release 3 (10.1.3.1.0). The examples may not apply to your specific environment, but they describe the steps required for a very specific and typical upgrade scenario.	Appendix A, "Step-By-Step Upgrade Examples"
Determine a 10.1.3.1.0 Topology to Install and Configure	Before you install 10*g* Release 3 (10.1.3.1.0), review your current 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) environment and then review the 10*g* Release 3 (10.1.3.1.0) supported topologies. You can then determine what 10*g* Release 3 (10.1.3.1.0) install types to select and how to configure your new 10*g* Release 3 (10.1.3.1.0) environment.	Task 1: Determine a 10g Release 3 (10.1.3.1.0) Topology to Install and Configure

Table 1–1 (Cont.) Description of Steps in the 10g Release 3 (10.1.3.1.0) Upgrade Process

Step	Description	More Information
Install and Configure a New 10.1.3.1.0 Environment	If you are upgrading from 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), then the first step in the upgrade process is to install and configure a new 10*g* Release 3 (10.1.3.1.0) environment that matches as closely as possible your existing Oracle Application Server environment. Review the upgrade concepts to determine which installation types and topologies to use for the 10*g* Release 3 (10.1.3.1.0) environment.	Section 4.2, "Task 2: Install and Configure Your New 10g Release 3 (10.1.3.1.0) Environment"
Perform Any Required Pre-Deployment Tasks	Before you redeploy your applications on 10*g* Release 3 (10.1.3.1.0), be sure to perform any pre-deployment tasks, such as creating data sources, deploying or configuring resource adapters, and setting OC4J server properties.	Section 4.3, "Task 3: Perform Any Required Pre-Deployment Tasks"
Redeploy Your Applications On 10.1.3.1.0	After you install and configure your new 10*g* Release 3 (10.1.3.1.0) environment, then you can redeploy your 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) applications on 10*g* Release 3 (10.1.3.1.0).	Section 4.4, "Task 4: Redeploy Your Applications on 10g Release 3 (10.1.3.1.0)"
Verify the Deployed Applications on 10.1.3.1	After you redeploy your applications on 10*g* Release 3 (10.1.3.1.0), verify that the applications and the application server itself is functioning properly.	Section 4.5, "Task 5: Verify Your Redeployed Applications on 10g Release 3 (10.1.3.1.0)"
Decommission 9.0.4 or 10.1.2 Source Oracle Homes	If you are upgrading from 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), you can decommission and deinstall your 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) Oracle homes after you verify that the upgrade is successful.	Section 4.6, "Task 6: Decommission the 10g (9.0.4) and 10g Release 2 (10.1.2) Oracle Homes"
Optionally Install Oracle SOA Components	After you upgrade to 10*g* Release 3 (10.1.3.1.0), you can optionally take advantage of the new Oracle Service-Oriented Architecture (SOA) components.	Chapter 5, "Taking Advantage of the 10g Release 3 (10.1.3.1.0) SOA Components"

2

Upgrade Concepts

This chapter introduces you to Oracle Application Server 10*g* Release 3 (10.1.3.1.0) and describes what is meant by an upgrade to 10*g* Release 3 (10.1.3.1.0).

This chapter contains the following sections:

2.1 What Is Oracle Application Server 10*g* Release 3 (10.1.3.1.0)?

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) was a significant new release that provided a complete Java 2, Enterprise Edition (J2EE) 1.4-compliant environment.

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) is an update to the 10*g* Release 3 (10.1.3.0.0) release that improves the reliability and performance of 10*g* Release 3 (10.1.3.0.0) and provides some additional functionality.

As with 10*g* Release 3 (10.1.3.0.0), 10*g* Release 3 (10.1.3.1.0) does not include all the Oracle Application Server components that were available with 10*g* (9.0.4) and 10*g* Release 2 (10.1.2).

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) is designed specifically:

- For administrators who are using OC4J to deploy and manage J2EE applications
- As a platform for developing and deploying Service Oriented Architecture (SOA) applications

Refer to the following sections for more information:

2.1.1 New Features That Were Provided With 10*g* Release 3 (10.1.3.0.0)

Table 2–1 provides a summary of the components and features that were new for 10*g* Release 3 (10.1.3.1.0). These features are available as is or improved for 10*g* Release 3 (10.1.3.1.0).

Table 2–1 Key Features of Oracle Application Server 10g Release 3 (10.1.3.0.0)

Feature Description	More Information
A new version of Oracle Containers for J2EE (OC4J) provides the containers, APIs, and services mandated by the J2EE 1.4 specification.	*Oracle Containers for J2EE Configuration and Administration Guide*
A new version of the Application Server Control that is based on the Java Management Extensions (JMX) technology, including the J2EE Management and J2EE Application Deployment specifications.	"Introduction to Administration Tools" in the *Oracle Application Server Administrator's Guide*
A new version of Oracle Process Manager and Notification Server (OPMN), which provides clustering capabilities for Oracle Application Server10*g* Release 3 (10.1.3.0.0) instances.	Appendix B, "Differences Between 10g Release 3 (10.1.3.1.0) and Previous Releases"
The new version of OC4J also includes a new, more flexible method for grouping OC4J instances to facilitate the deployment and management of your J2EE applications across a cluster topology.	Appendix B, "Differences Between 10g Release 3 (10.1.3.1.0) and Previous Releases"
Introduction of the cluster topology, which identifies the hierarchy of and relationships between your application server instances, OC4J instances, Oracle HTTP Server instances, and OC4J groups in your Oracle Application Server environment.	Section B.1, "Using the Cluster Topology Instead of an OracleAS Farm"

2.1.2 New Features Provided with 10*g* Release 3 (10.1.3.1.0)

Table 2–2 provides a summary of the additional features introduced for 10*g* Release 3 (10.1.3.1.0).

Table 2–2 New Features of Oracle Application Server 10g Release 3 (10.1.3.1.0)

Feature Description	More Information
Support for a set of Service Oriented Architecture (SOA) components (the Oracle SOA Suite) that developers can use to build and deploy industry-standard, SOA applications.	Chapter 5, "Taking Advantage of the 10g Release 3 (10.1.3.1.0) SOA Components"
New Application Server Control Console features, including the ability to create OC4J instances, organize them into OC4J groups, and the ability to review and configure ports, routing IDs, and application server instances in a cluster topology.	"Introduction to Administration Tools" in the *Oracle Application Server Administrator's Guide*
Support for OC4J Java Single Sign-On (Java SSO), a lighter-weight single sign-on solution supplied with OC4J.	"Configuring Instances to Use OC4J Java Single Sign-On" in the *Oracle Application Server Administrator's Guide*
Support for Web Services Inspection Language (WSIL), a lightweight Web services directory protocol that provides an extensible schema for a single document catalog of services	"Web Service Inspection Language 1.0" in the *Oracle Application Server Web Services Developer's Guide*

2.2 Supported Upgrade Paths

This document describes how to redeploy your Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) J2EE applications in a new Oracle Application Server 10*g* Release 3 (10.1.3.1.0) environment.

If you are using Oracle Application Server Release 2 (9.0.2) or earlier, then you must upgrade your environment to 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) before using the instructions in this guide.

> **See Also:** *Oracle Application Server Upgrade Guide* in the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2.0.2) documentation library on the Oracle Technology Network (OTN):
>
> `http://www.oracle.com/technology/documentation/`

If you are using Oracle Application Server 10*g* Release 3 (10.1.3.0.0), apply 10*g* Release 3 (10.1.3.1.0) Patch Set 1 to your exisiting 10*g* Release 3 (10.1.3.0.0) Oracle home directories.

> **See Also:** *Oracle Application Server Patchset Notes 10g Release 3 (10.1.3.1.0) Patch Set 1 (10.1.3.1.0) for Linux and Microsoft Windows* available on Oracle*MetaLink*

2.3 Reviewing Your Current Oracle Application Server Environment

Before you begin the upgrade process, you should be familiar with your current Oracle Application Server environment. For example, you should be able to answer the following questions:

- How many 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) middle tiers are installed in your environment?
- Are your middle tiers part of an Oracle Application Server Farm? In other words, are they using the same OracleAS Metadata Repository?
- Are you using Oracle Application Server Clusters to manage multiple J2EE and Web Cache installations?
- Are you using state replication for your deployed applications?
- Are you taking advantage of OracleAS Identity Management components, such as Oracle Internet Directory?
- What J2EE features do your deployed applications rely upon (for example, JDBC, JMS, Java Transaction Manager)?
- What customizations have you made to your Oracle HTTP Server and Oracle Containers for J2EE (OC4J) configuration files, such as the `httpd.conf` file and `server.xml`?

If you have answers to these types of questions, then it will be easier to select an equivalent topology for your new 10*g* Release 3 (10.1.3.1.0) environment and to perform any configuration tasks that might be required before you can deploy your 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) application on 10*g* Release 3 (10.1.3.1.0).

2.4 Upgrade Tools and Rules to Follow

Unlike the upgrade to previous versions of Oracle Application Server, such as 10*g* Release 2 (10.1.2), the upgrade to 10*g* Release 3 (10.1.3.1.0) does not involve any specific upgrade tools.

Instead, you use Oracle Universal Installer to install a new 10*g* Release 3 (10.1.3.1.0) environment, make any necessary configuration changes, verify that your applications are compatible with 10*g* Release 3 (10.1.3.1.0), and then redeploy your 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) applications on 10*g* Release 3 (10.1.3.1.0).

Before redeploying your applications on 10*g* Release 3 (10.1.3.1.0), be sure to closely follow the instructions in this guide, including the considerations listed in Appendix C, "Things to Consider When Redeploying 10g (9.0.4) and 10g Release 2 (10.1.2) Applications".

2.5 System Availability and Backup Strategies During Upgrade to 10*g* Release 3 (10.1.3.1.0)

When you upgrade from 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) to 10*g* Release 3 (10.1.3.1.0), no changes are made to your existing installations. Instead, you install and configure new 10*g* Release 3 (10.1.3.1.0) Oracle homes and redeploy your applications in the new environment.

As a result, there is no need to perform special upgrade-specific backups or to shut down your existing environment during the upgrade process. Instead, you can perform the upgrade procedures independently from the existing installations. Simply maintain your existing schedule of regular backups and maintenance for your current environment until you have cutover completely to the new 10*g* Release 3 (10.1.3.1.0) environment.

You can (and should) perform the upgrade operations on a separate host. After the upgrade is complete and you have verified that the redeployed applications are working successfully, you can then switch over and begin using the new 10*g* Release 3 (10.1.3.1.0) installations at a convenient time.

Note, however, if your applications depend upon a database or other backend components, such as an LDAP directory, you must take care to be sure that no data is lost during the transition to the new 10*g* Release 3 (10.1.3.1.0) environment.

Similarly, any state replication of applications across 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) OracleAS Clusters will be lost during the switchover to 10*g* Release 3 (10.1.3.1.0). For more information on state replication and application clustering in 10*g* Release 3 (10.1.3.1.0), refer to the *Oracle Containers for J2EE Configuration and Administration Guide*.

2.6 Documentation Road Map for Previous Oracle Application Server Users

To learn more about Oracle Application Server 10*g* Release 3 (10.1.3.1.0), take advantage of the following documentation resources. For example, consider reviewing the 10*g* Release 3 (10.1.3.1.0) books in the following order:

1. Review the *Oracle Application Server Installation Guide* for your platform.

   In particular, review Section 1.3, "Recommended Topologies," which describes common configurations to consider when installing 10*g* Release 3 (10.1.3.1.0).

2. Review the *Oracle Application Server Enterprise Deployment Guide*, which provides a detailed example of using 10*g* Release 3 (10.1.3.1.0) as the middle tier in an enterprise-wide deployment of Oracle Application Server.
3. Refer to the *Oracle Application Server Administrator's Guide* for information about the tools and procedures you can use to manage your 10*g* Release 3 (10.1.3.1.0) environment.
4. Refer to the *Oracle Containers for J2EE Configuration and Administration Guide* for general information about managing OC4J 10*g* Release 3 (10.1.3.1.0) instances.

3

10*g* Release 3 (10.1.3.1.0) Version Compatibility

Use this chapter to understand how Oracle Application Server 10*g* Release 3 (10.1.3.1.0) can work within an existing Oracle Application Server environment. Refer to the following sections for more information:

- Using OracleMetaLink to Obtain the Latest Oracle Application Server Certification Information
- General Compatibility Information
- Integrating 10g Release 3 (10.1.3.1.0) into Your Existing Environment

3.1 Using Oracle*MetaLink* to Obtain the Latest Oracle Application Server Certification Information

The compatibiliy information in this chapter was accurate at the time this manual was written. However, this information is subject to change. For the most up-to-date information about compatibility with other Oracle Application Server releases, as well as information about the latest patchsets, refer to Oracle*MetaLink*:

`http://metalink.oracle.com/`

After logging into Oracle*MetaLink*, click the **Certify** tab at the top of the page. From the resulting Web page, you can view the latest certifications by product, platform, and product availability.

3.2 General Compatibility Information

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) is designed to be installed in parallel to your existing Oracle Application Server installations. For example, you can install a new 10*g* Release 3 (10.1.3.1.0) Oracle home on a host where you have already installed Oracle Application Server 10*g* (9.0.4) or Oracle Application Server 10*g* Release 2 (10.1.2).

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) does not require an OracleAS Metadata Repository, but you can configure your 10*g* Release 3 (10.1.3.1.0) installations to take advantage of an existing 10*g* Release 2 (10.1.2) OracleAS Identity Management installation.

Table 3–1 provides an overview of Oracle Application Server 10*g* Release 3 (10.1.3.1.0) compatibility with previous Oracle Application Server releases.

For more complete information about compatibility with other Oracle software products and versions, refer to the **Certify** section on Oracle*MetaLink*:

`http://metalink.oracle.com/`

Table 3–1 10g Release 3 (10.1.3.1.0) Compatibility With Previous Versions

Component and Release	10*g* Release 3 (10.1.3.1.0) Compatibility Information
10*g* (9.0.4) Middle Tiers	■ You can install and run 10*g* (9.0.4) and 10*g* Release 3 (10.1.3.1.0) middle tiers on the same host. ■ The two Oracle homes can also share the same OracleAS Identity Management. ■ However, 10*g* (9.0.4) and 10*g* Release 3 (10.1.3.1.0) middle-tier instances cannot be in the same cluster. ■ Before you can use ORMI connections between the middle tiers, you must apply a patch on the 10*g* (9.0.4) middle tier. For more information, see Section C.7.1, "Applying Compatibility Patches for 10g (9.0.4) and 10g Release 2 (10.1.2)"
10*g* (9.0.4.0.x) OracleAS Identity Management	Not supported.
10*g* (9.0.4.1) OracleAS Identity Management	Not supported.
10*g* (9.0.4.2) OracleAS Identity Management	Not supported.
10*g* Release 2 (10.1.2) Middle Tiers	■ You can run 10*g* Release 2 (10.1.2) and 10*g* Release 3 (10.1.3.1.0) middle tiers on the same host. ■ The two Oracle homes can also share the same OracleAS Identity Management. ■ However, 10*g* Release 2 (10.1.2) and 10*g* Release 3 (10.1.3.1.0) middle-tier instances cannot be in the same cluster. ■ Before you can use ORMI connections between the middle tiers, you must apply a patch on the 10*g* (9.0.4) middle tier. For more information, see Section C.7.1, "Applying Compatibility Patches for 10g (9.0.4) and 10g Release 2 (10.1.2)"
10*g* Release 2 (10.1.2.0.0) OracleAS Identity Management	Not supported; Oracle recommends that you apply the 10*g* Release 2 (10.1.2.1.0) patchset, which is available on Oracle*MetaLink*.
10*g* Release 2 (10.1.2.0.1) Standard Edition One OracleAS Identity Management	Not supported; Oracle recommends that you apply the 10*g* Release 2 (10.1.2.1.0) patchset, which is available on Oracle*MetaLink*.
10*g* Release 2 (10.1.2.0.2) OracleAS Identity Management	You can configure your 10*g* Release 3 (10.1.3.1.0) middle tiers to use an existing 10*g* Release 2 (10.1.2.0.2) OracleAS Identity Management. See "Configuring Instances to Use 10.1.4 or 10.1.2 Oracle Identity Management" in the *Oracle Application Server Administrator's Guide*.
10*g* Release 2 (10.1.2.1.0) OracleAS Identity Management	You can configure your 10*g* Release 3 (10.1.3.1.0) middle tiers to use an existing 10*g* Release 2 (10.1.2.1.0) OracleAS Identity Management. See "Configuring Instances to Use 10.1.4 or 10.1.2 Oracle Identity Management" in the *Oracle Application Server Administrator's Guide*.

Table 3–1 (Cont.) 10g Release 3 (10.1.3.1.0) Compatibility With Previous Versions

Component and Release	10*g* Release 3 (10.1.3.1.0) Compatibility Information
10*g* Release 3 (10.1.4.0.1) Oracle Identity Management	You can configure your 10*g* Release 3 (10.1.3.1.0) middle tiers to use an existing 10*g* Release 3 (10.1.4.0.1) Oracle Identity Management. See "Configuring Instances to Use 10.1.4 or 10.1.2 Oracle Identity Management" in the *Oracle Application Server Administrator's Guide*.
Oracle Access Manager 10*g* Release 3 (10.1.4.0.1)	You can use the Oracle Access Manager 10*g* Release 3 (10.1.4.0.1) security provider for 10*g* Release 3 (10.1.3.1.0) authentication, authorization, and single sign-on. For more information, refer to "Oracle Access Manager" in the *Oracle Containers for J2EE Security Guide*.

3.3 Integrating 10*g* Release 3 (10.1.3.1.0) into Your Existing Environment

The following sections describe how you can use your existing Oracle Application Server components with your new 10*g* Release 3 (10.1.3.1.0) installations:

- Installing 10g Release 3 (10.1.3.1.0) Middle Tiers in an Existing Oracle Application Server Environment
- Using Existing OracleAS Infrastructure Components
- Using Existing Oracle HTTP Server Instances
- Using an Existing Oracle Web Cache Installation As a Reverse Proxy

> **See Also:** *Oracle Application Server Enterprise Deployment Guide* for a description of a Oracle Application Server configurations that are designed to support large-scale, mission-critical business software applications

3.3.1 Installing 10*g* Release 3 (10.1.3.1.0) Middle Tiers in an Existing Oracle Application Server Environment

There are no restrictions to prevent you from installing 10*g* Release 3 (10.1.3.1.0) middle tiers in an existing 10*g* (9.0.4), 10*g* Release 2 (10.1.2), or 10*g* Release 3 (10.1.4.0.1) environment. In fact, if necessary, you can install your new 10*g* Release 3 (10.1.3.1.0) middle tiers on the same host as your existing Oracle Application Server Oracle homes.

As with other Oracle software products, Oracle Universal Installer will check for available ports and perform additional prerequisite checks to be sure the host computer meets the 10*g* Release 3 (10.1.3.1.0) hardware and software requirements.

However, while 10*g* Release 3 (10.1.3.1.0) middle tiers can co-exist with previous Oracle Application Server releases, there are fundamental differences between these releases. For example, 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0) introduce significant changes to application server clustering. For more information refer to the following:

- Appendix B, "Differences Between 10g Release 3 (10.1.3.1.0) and Previous Releases"
- Section 3.2, "General Compatibility Information"

3.3.2 Using Existing OracleAS Infrastructure Components

Your existing 10*g* Release 2 (10.1.2) or 10*g* Release 3 (10.1.4.0.1) OracleAS Infrastructure consists of an OracleAS Metadata Repository and OracleAS Identity Management.

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) does not require an OracleAS Metadata Repository because none of the components delivered with this release require specific schemas or a database to host those schemas.

However, there are many reasons to configure your 10*g* Release 3 (10.1.3.1.0) middle tier to use your existing OracleAS Identity Management installation. Table 3–2 lists the tasks you can accomplish when you use OracleAS Identity Management with 10*g* Release 3 (10.1.3.1.0).

To configure your 10*g* Release 3 (10.1.3.1.0) instance to use OracleAS Identity Management, select the **Identity Management** task on the OC4J Administration page in the Application Server Control Console.

> **See Also:** "Configuring Instances to Use 10.1.4 or 10.1.2 Oracle Identity Management" in the *Oracle Application Server Administrator's Guide*

Table 3–2 Using Oracle Identity Management with 10g Release 3 (10.1.3.1.0)

Task	For More Information
Use Oracle Internet Directory as the security provider for the J2EE applications you deploy.	"Overview of Managing Security Providers" in the Application Server Control online help
Use Oracle Internet Directory as the security provider for the 10*g* Release 3 (10.1.3.1.0) Application Server Control. You can then use Oracle Internet Directory to manage the administrator accounts that are used to log in to the Application Server Control Console.	"Configuring the Security Provider for Application Server Control" in the Application Server Control online help
Configure your J2EE applications to use Oracle Single Sign-On	"About Using OracleAS Single Sign-On" in the Application Server Control online help "Configuring Instances to Use 10.1.4 or 10.1.2 Oracle Identity Management" in the *Oracle Application Server Administrator's Guide*
Use 10*g* Release 2 (10.1.2) Single Sign-on to authenticate connections between a Web application and its Web services. In this scenario, it is assumed that both the Web application and the Web service are configured to use the Security Assertion Markup Language (SAML).	"Configuring Single Sign-on Using SAML" in the *Oracle Application Server Web Services Security Guide*

3.3.3 Using Existing Oracle HTTP Server Instances

When you install Oracle Application Server 10*g* Release 3 (10.1.3.1.0), you can choose to install Oracle HTTP Server in one or more of your 10*g* Release 3 (10.1.3.1.0) Oracle homes. You can then configure your environment to use the Oracle HTTP Server to serve as a front-end for the J2EE applications you deploy.

Alternatively, you can configure your existing 10*g* Release 2 (10.1.2) Oracle HTTP Server as the front-end to your new 10*g* Release 3 (10.1.3.1.0) instances.

For more information, see "Configuring Oracle Application Server 10.1.2 with Oracle Application Server 10.1.3" in the *Oracle Application Server Administrator's Guide*.

3.3.4 Using an Existing Oracle Web Cache Installation As a Reverse Proxy

You can use Release 2 (10.1.2) OracleAS Web Cache as a reverse proxy for your 10g Release 3 (10.1.3) middle-tier instance. As a reverse proxy server, OracleAS Web Cache acts a gateway to the middle-tier servers.

For more information, see "Configuring 10.1.2 OracleAS Web Cache as a Reverse Proxy" in the *Oracle Application Server Administrator's Guide*.

4

Performing the Upgrade to 10*g* Release 3 (10.1.3.1.0)

This chapter describes the general instructions for upgrading from Oracle Application Server 10*g* (9.0.4) or 10*g* Release 2 (10.1.2).

> **See Also:** Appendix A, "Step-By-Step Upgrade Examples" to review some specific examples of how to upgrade your environment to 10*g* Release 3 (10.1.3.1.0)

This chapter contains the following sections:

4.1 Task 1: Determine a 10*g* Release 3 (10.1.3.1.0) Topology to Install and Configure

Before you begin the upgrade to 10*g* Release 3 (10.1.3.1.0), consider the following steps that will help you prepare for the upgrade:

1. Review your current Oracle Application Server environment.

   You should have a thorough understanding of your current Oracle Application Server environment, including any requirements for clustering, load balancing, or state replication among the middle tier applications.

   > **See Also:** Section 2.3, "Reviewing Your Current Oracle Application Server Environment"

   In addition, you should be aware of any required infrastructure components, such as Oracle Internet Directory or a OracleAS Metadata Repository.

   > **See Also:** Section 3.3, "Integrating 10g Release 3 (10.1.3.1.0) into Your Existing Environment" for information about integrating 10*g* Release 3 (10.1.3.1.0) into your existing Oracle Application Server environment

2. Review the 10*g* Release 3 (10.1.3.1.0) recommended topologies, which are described in the *Oracle Application Server Installation Guide*.

   Note in particular the recommended topologies that map to your current environment.

   For example, if you are upgrading from 10*g* Release 2 (10.1.2), and you are deploying and managing your applications in an DCM-Managed OracleAS Cluster, then consider installing multiple 10*g* Release 3 (10.1.3.1.0) instances as part of a cluster topology.

   You can then use 10*g* Release 3 (10.1.3.1.0) groups to deploy your applications to multiple OC4J instances in the cluster.

   See Also: Appendix B, "Differences Between 10g Release 3 (10.1.3.1.0) and Previous Releases" for detailed information about the differences you can expect in the 10*g* Release 3 (10.1.3.1.0) environment

   Appendix A, "Step-By-Step Upgrade Examples" for examples of how you can upgrade to a similar environment in 10*g* Release 3 (10.1.3.1.0)

4.2 Task 2: Install and Configure Your New 10*g* Release 3 (10.1.3.1.0) Environment

After you decide upon a 10*g* Release 3 (10.1.3.1.0) topology, you can begin the installation. Note the following guidelines when installing and configuring your new 10*g* Release 3 (10.1.3.1.0) environment:

- Refer to the *Oracle Application Server Installation Guide* for specific instructions for installing each Oracle Application Server instance.
- If you are installing multiple Oracle Application Server instances, use the cluster topology options within the installer to configure the members of the cluster--or you can use the Application Server Control Console to configure the cluster later. For more information, see "Configuring the Cluster Topology" in the Application Server Control online help.
- Be sure to create any required additional OC4J instances and to organize your OC4J instances into groups before you continue with the upgrade process. The goal should be to configure your environment as much as possible before deploying your 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) applications.
- Consider taking advantage of the new features available with 10*g* Release 3 (10.1.3.1.0). For more information, see Appendix D, "Differences Between 10g Release 3 (10.1.3.1.0) and 10g Release 3 (10.1.3.0.0)".

4.3 Task 3: Perform Any Required Pre-Deployment Tasks

After you install and configure your 10*g* Release 3 (10.1.3.1.0) environment, perform any OC4J server configuration tasks that are required by your applications or by your organization.

For example, refer to the following sections for more information about some common predeployment tasks:

- Configuring Data Sources
- Configuring Resource Adapters

- Setting Server Properties
- Configuring Oracle HTTP Server

4.3.1 Configuring Data Sources

Most J2EE applications that you develop and deploy will require a backend database or other data source. Before you redeploy your applications on 10*g* Release 3 (10.1.3.1.0), consider creating any required data sources using the Application Server Control Console. You can create a data source for one OC4J instance or for a group of OC4J instances.

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) supports managed data sources, which rely upon JDBC connection pools, as well as native data sources.

For more information about defining data sources for 10*g* Release 3 (10.1.3.1.0), see the following:

- "Managing Data Sources and JDBC Connection Pools" in the in the Application Server Control online help
- "Data Sources" in the *Oracle Containers for J2EE Services Guide*

Before you deploy your applications that depend upon data sources, refer to Section C.3, "Data Source Considerations", which describes how Oracle Application Server support for data sources has changed since 10*g* (9.0.4) and 10*g* Release 2 (10.1.2).

4.3.2 Configuring Resource Adapters

If your deployed applications rely on any specific resource adapters, you can deploy the resource adapters to a single OC4J instance or to a group of OC4J instances. You can use the Application Server Control Console to deploy standalone resource adpaters or resource adapters that are embedded within an application.

For more information about defining data sources for 10*g* Release 3 (10.1.3.1.0), see the following:

- "Overview of Managing Resource Adapters" in the in the Application Server Control online help
- *Oracle Containers for J2EE Resource Adapter Administrator's Guide*

4.3.3 Setting Server Properties

If made any specific changes to your default 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Oracle Containers for J2EE server properties, consider using the Application Server Control Console to make the same changes to your 10*g* Release 3 (10.1.3.1.0) OC4J instances before you deploy your applications on 10*g* Release 3 (10.1.3.1.0).

For example, you can modify the number JVMs assigned for each OC4J instance, modify the port ranges for RMI, RMIS, and JMS, set the protocol to AJP or HTTP for default website, or set a specific set of Java options to process on OC4J startup.

For more information, see:

- "Setting OC4J Server Properties" in the Application Server Control online help.
- "OC4J Runtime Configuration" in the *Oracle Containers for J2EE Configuration and Administration Guide*

4.3.4 Configuring Oracle HTTP Server

If you have made any site-specific configuration changes to the Oracle HTTP Server `httpd.conf` configuration file in the 10*g* Release 2 (10.1.2) Oracle home, apply those changes to the `httpd.conf` file in the 10*g* Release 3 (10.1.3.1.0) Oracle home.

As in 10*g* (9.0.4) and 10*g* Release 2 (10.1.2), the 10*g* Release 3 (10.1.3.1.0) Oracle HTTP Server is based on the on Apache 1.3 Web Server. As a result, you can copy any configuration changes (such as modifications to the `httpd.conf` file) directly to the `httpd.conf` file 10*g* Release 3 (10.1.3.1.0) Oracle home.

4.4 Task 4: Redeploy Your Applications on 10*g* Release 3 (10.1.3.1.0)

The following sections describe how to redeploy your 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) applications on your new 10*g* Release 3 (10.1.3.1.0) environment:

- Reviewing the Considerations When Redeploying Applications on 10g Release 3 (10.1.3.1.0)
- Using Application Server Control to Redeploy Your Applications

4.4.1 Reviewing the Considerations When Redeploying Applications on 10*g* Release 3 (10.1.3.1.0)

After you have configured the 10*g* Release 3 (10.1.3.1.0) environment with any configuration settings required by your applications, review the applications themselves, as well as Appendix C, "Things to Consider When Redeploying 10g (9.0.4) and 10g Release 2 (10.1.2) Applications".

If your applications take advantage of any J2EE features or capabilities described in Appendix C, then you might have to make some changes to the application. These changes might be necessary to ensure that the application deploys successfully or to ensure that the all functionality of your application works as expected when deployed on 10*g* Release 3 (10.1.3.1.0).

4.4.2 Using Application Server Control to Redeploy Your Applications

After you have configured the new 10*g* Release 3 (10.1.3.1.0) environment appropriately for your application, and you have reviewed the conderations in Appendix C, you can then deploy your applications either to a single OC4J instance or to a group of OC4J instances.

To deploy an application to a single OC4J instance, navigate to the OC4J Home page and click **Applications**. From the OC4J Applications page, click **Deploy** and follow the instructions on the screen.

To deploy an application to an OC4J group, click the name of the group on the Cluster Topology page, and then click **Applications** on the Group page. From the Group Applications page, click **Deploy** and follow the instructions on the screen.

> **See Also:** "Deploying an Application" in the Application Server Control online help
>
> *Oracle Containers for J2EE Deployment Guide*

4.5 Task 5: Verify Your Redeployed Applications on 10*g* Release 3 (10.1.3.1.0)

After you deploy the J2EE applications on 10*g* Release 3 (10.1.3.1.0), be sure to verify that all the features of your applications work as expected. In particular, test any features that were modified due to information available in Appendix C.

The applications you deploy on the 10*g* Release 3 (10.1.3.1.0) environment can be managed from the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console. After you deploy your applications, log into the Application Server Control Console and verify that the applications are listed on the Cluster Topology page and that they are up and running. Drill down to the Application Home page to view general information about the applications, or to review performance and configuration information.

> **See Also:** "Overview of Managing Applications" in the Application Server Control online help

If you you were using state replication to maintain state across multiple instances of your applications in the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) environment, refer to "Overview of Application Clustering" in the Application Server Control online help for information about setting up a similar environment in 10*g* Release 3 (10.1.3.1.0).

Verification of an application also includes making sure that you can access and log into the application, that you can use all the features of the application, and that communications with data sources and other external resources work as expected.

4.6 Task 6: Decommission the 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) Oracle Homes

The upgrade process leaves the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Oracle homes 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.

You can continue running the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Oracle homes for as long as necessary. However, after you cutover to the new 10*g* Release 3 (10.1.3.1.0) environment and you are sure that you no longer need to support or run the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) environment, you can save disk space and other resources by decommissioning the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Oracle home.

The following sections provide more information about decommissioning an upgraded source Oracle home:

- Preserving Application Files and Log Files
- Removing the Source Oracle Home from the OracleAS Farm
- Deinstalling a 10g (9.0.4) or 10g Release 2 (10.1.2) Oracle Home

4.6.1 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 your applications or by other Oracle homes, you should move them to another location before you decommission the source Oracle home, and, where necessary, change any references to the files to the new location.

4.6.2 Removing the Source Oracle Home from the OracleAS Farm

If the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) middle tier instance is a member of an OracleAS Farm, be sure to remove the source instance from the farm before you deinstall the source Oracle home.

For example, after you upgrade an instance that was using an OracleAS Infrastructure, the source instance remains in the list of instances on the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Application Server Control Console Farm page.

To remove the source instance from the farm and from the Farm page, use the following command in the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Oracle home:

On Windows systems:

```
SOURCE_ORACLE_HOME\dcm\bin\dcmctl leavefarm
```

On UNIX systems:

```
SOURCE_ORACLE_HOME/dcm/bin/dcmctl leavefarm
```

In this example, replace *SOURCE_ORACLE_HOME* with the complete path to the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Oracle home, and replace *OC4J_Instance* with the name of the OC4J instance that you used to deploy the FAQApp application.

> **See Also:** For more information about the `dcmctl` command-line utility and the Oracle Application Server Farm page, refer to the following resources in the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) documentation library:
>
> - *Distributed Configuration Management Administrator's Guide*
> - "Introduction to Administration Tools" in the *Oracle Application Server Administrator's Guide*
>
> The 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) documentation libraries are available from the main documentation page the Oracle Technology Network:
>
> `http://www.oracle.com/technology/documentation/`

4.6.3 Deinstalling a 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) 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 previous 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:** The installation guide for your platform in the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) documentation library for instructions on deinstalling Oracle Application Server Oracle homes

5

Taking Advantage of the 10*g* Release 3 (10.1.3.1.0) SOA Components

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) provides not only the core Oracle Application Server components, such as Oracle Containers for J2EE and Oracle Process Manager and Notification Server. Oracle Application Server 10*g* Release 3 (10.1.3.1.0) also provides a suite of Oracle software components that provide you with the tools you need to implement a Service Oriented Architecture (SOA) for your business.

After you upgrade your applications from 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), you can optionally take advantage of the Oracle SOA Suite.

Refer to the following sections for more information:

5.1 About Oracle SOA Suite

Oracle SOA Suite is a standards-based best-of-breed technology suite that consists of the following:

- Integrated Service Environment (ISE) to develop services
- Oracle BPEL Process Manager to orchestrate services into business processes
- Oracle Enterprise Service Bus (ESB) to connect existing IT systems and business partners as a set of services
- Oracle Business Rules for dynamic decisions at runtime that can be managed by business users or business analysts
- Oracle Business Activity Monitoring to monitor services and disparate events and provide real-time visibility into the state of the enterprise, business processes, people, and systems.
- Oracle Web Services Manager to secure and manage authentication, authorization, and encryption policies on services that is separate from your service logic
- UDDI registry to discover and manage the lifecyle of Web services.

- Oracle Application Server 10*g* Release 3 (10.1.3.1.0) to provide a complete Java 2, Enterprise Edition (J2EE) 1.4-compliant environment for your J2EE applications.

For an introduction to Oracle SOA Suite, refer to the *Oracle Application Server Quick Tour*, which introduces you to the Oracle SOA Suite and guides you on a tour of the SOA Order Booking application. The SOA Order Booking application demonstrates how a number of applications, both internal to an enterprise, and external at other sites, can be integrated using the SOA architecture paradigm to create one cohesive ordering system.

5.2 Installing the Oracle SOA Suite Components

You can install Oracle SOA Suite as part of the 10*g* Release 3 (10.1.3.1.0) installation procedure, or you can install only the Oracle Application Server core components and then install the Oracle SOA Suite later.

Refer to the following sections for more information:

- Installing Oracle SOA Suite as Part of the 10g Release 3 (10.1.3.1.0) Installation
- Installing Oracle SOA Suite Components Separately

5.2.1 Installing Oracle SOA Suite as Part of the 10*g* Release 3 (10.1.3.1.0) Installation

To install Oracle SOA Suite as part of an 10*g* Release 3 (10.1.3.1.0) installation, select one of the following options when running Oracle Universal Installer to install 10*g* Release 3 (10.1.3.1.0):

- Select the **Basic** installation type.

  OR

- Select **Advanced** and then select the **J2EE Server, Web Server, and SOA Suite** installation type.

Both of these installation options install Oracle SOA Suite; for the differences between these two installation types, as well as complete instructions for installing 10*g* Release 3 (10.1.3.1.0), see the *Oracle Application Server Installation Guide* for your platform.

5.2.2 Installing Oracle SOA Suite Components Separately

To install Oracle SOA Suite later:

1. Install 10*g* Release 3 (10.1.3.1.0) using any of the Advanced installation types, except the J2EE Server, Web Server, and SOA Suite installation type. These installation types install the core components only.
2. Later, install the SOA components, one-by-one, using the individual installation procedures for each of the SOA components.

Table 5–1 includes a list of the documentation resources available to help you install the Oracle SOA Suite components.

Table 5–1 Locating the Installation Documentation for Oracle SOA Suite Components

Component	Installation Documentation
Oracle BPEL Process Manager	*Oracle BPEL Process Manager Installation Guide*
Oracle Enterprise Service Bus (ESB)	*Oracle Enterprise Service Bus Installation Guide*

Table 5–1 (Cont.) Locating the Installation Documentation for Oracle SOA Suite

Component	Installation Documentation
Oracle Business Rules	Installed as part of the 10*g* Release 3 (10.1.3.1.0) installation; to start Oracle Business Rules, see the *Oracle Business Rules User's Guide*.
Oracle Business Activity Monitoring	*Oracle Business Activity Monitoring Installation Guide*
Oracle Web Services Manager	*Oracle Web Services Manager Installation Guide*

5.3 Where to Learn About Developing SOA Applications

After you install Oracle SOA Suite, you can begin developing applications that take advantage of the Service Oriented Architecture. To get started developing your SOA applications, refer to the following documentation:

- *Oracle Application Server Quick Tour*, which introduces you to the Oracle SOA Suite and guides you on a tour of the SOA Order Booking application
- *Oracle Application Server Tutorial*, which provides a step-by-step approach for building the SOA Order Booking application yourself
- *Oracle SOA Suite Developer's Guide*, which an in-depth description of designing and developing a SOA application, using the SOA Order Booking application as an example

5.4 Where to Learn About Upgrading From Previous Versions of the SOA Components

Table 5–2 provides a list of the resources available if you are upgrading from a previous, standalone version of an Oracle SOA component.

Table 5–2 Information About Upgrading From Previous Versions of the SOA Components

SOA Component	Upgrade Information
Oracle BPEL Process Manager	Refer to the *Oracle BPEL Process Manager Installation Guide* for information about upgrading from previous versions of Oracle BPEL Process Manager.
Oracle Enterprise Service Bus	Oracle Enterprise Service Bus is a new Service Oriented Architecture component for Oracle Application Server 10*g* Release 3 (10.1.3.1.0). As a result, no upgrade documentation is available.
Oracle Business Rules	Oracle Business Rules was a new component for Oracle Application Server 10*g* Release 3 (10.1.3.0.0). If you are using Oracle Business Rules for 10*g* Release 3 (10.1.3.0.0), then you can update your installation by applying the 10*g* Release 3 (10.1.3.1.0) patchset. For more information, see *Oracle Application Server Patchset Notes 10g Release 3 (10.1.3.1.0) Patch Set 1 (10.1.3.1.0) for Linux and Microsoft Windows*, which are available as part of the patchset download from Oracle*MetaLink*.
Oracle Business Activity Monitoring	Refer to the *Oracle Business Activity Monitoring Installation Guide* for information about upgrading from previous versions of Oracle Business Activity Monitoring.

Table 5–2 (Cont.) Information About Upgrading From Previous Versions of the SOA Components

SOA Component	Upgrade Information
Oracle Web Services Manager	Upgrading from previous versions of Oracle Web Services Manager is currently not supported.

A

Step-By-Step Upgrade Examples

This chapter provides step-by-step examples that can help you better understand the 10*g* Release 3 (10.1.3.1.0) upgrade process.

Refer to the following for more information:

- Upgrading FAQApp on a Single Oracle Application Server Instance
- Upgrading FAQApp in a Clustered Environment

A.1 Upgrading FAQApp on a Single Oracle Application Server Instance

The following sections provide an example of upgrading and redeploying an existing application on a single 10*g* Release 3 (10.1.3.1.0) OC4J instance:

- Starting Point for the FAQApp Upgrade
- Overview of the FAQApp Upgrade Procedure
- The FAQApp Upgrade Procedure

A.1.1 Starting Point for the FAQApp Upgrade

This procedure assumes the following starting point for the FAQApp upgrade:

- You have installed and configured a single 10*g* Release 2 (10.1.2) middle-tier Oracle home.

  Note: This procedure describes specifically how to upgrade from 10*g* Release 2 (10.1.2), but the procedure is also valid if you are using 10*g* (9.0.4) as your starting point.

- You have optionally made some site-specific changes to the Oracle HTTP Server configuration file (`httpd.conf`).
- You have successfully deployed the FAQApp sample application to the 10*g* Release 2 (10.1.2) middle tier.

  The FAQApp is available for download from the following location on Oracle Technology Network (OTN):

  `http://www.oracle.com/technology/tech/java/oc4j/demos/904/index.html`

See Also: "Configuring the FAQ Application Demo" in the *Oracle Application Server Containers for J2EE User's Guide* in the 10*g* Release 2 (10.1.2) documentation library, which is available from the following location on the Oracle Technology Network (OTN):

`http://www.oracle.com/technology/documentation/appserver101202.html`

- You now want to deploy the same FAQApp application on an instance of Oracle Application Server 10*g* Release 3 (10.1.3.1.0).
- You have installed and configured Apache Ant, which is a Java-based build tool that is included in the following directory of your 10*g* Release 3 (10.1.3.1.0) Oracle home:

  `(UNIX) 1013_ORACLE_HOME/ant/`
  `(Windows) 1013_ORACLE_HOME\ant\`

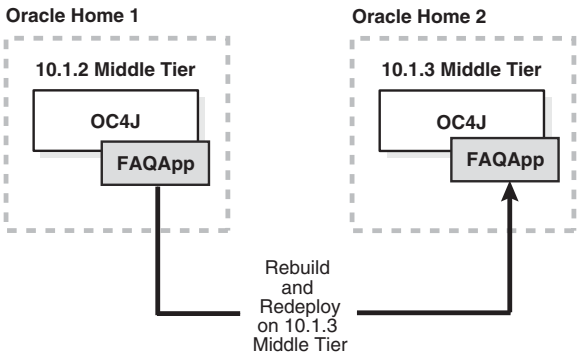
  Apache Ant, as well as documentation for using the tool, is also available from the following Web site:

  `http://ant.apache.org/`

A.1.2 Overview of the FAQApp Upgrade Procedure

The steps in the following procedure involve installing a new 10*g* Release 3 (10.1.3.1.0) Oracle home, modifying and rebuilding FAQApp, and then redeploying FAQapp on the new 10*g* Release 3 (10.1.3.1.0) OC4J instance, as shown in Figure A–1.

Figure A–1 Overview of the FAQApp Upgrade Procedure



A.1.3 The FAQApp Upgrade Procedure

The following sections describe the steps you can follow to install and configure a new 10*g* Release 3 (10.1.3.1.0) Oracle home and then recompile and redeploy the FAQApp on the new installation:

- Step 1: Install 10g Release 3 (10.1.3.1.0)
- Step 2: Apply Any Site-Specific Oracle HTTP Server Configuration Settings
- Step 3: Configure the FAQApp Data Source in the 10g Release 3 (10.1.3) Instance
- Step 4: Rebuild the FAQApp EAR File With Required JSP Library Archives
- Step 5: Deploy the Modified FAQApp EAR File on 10g Release 3 (10.1.3)
- Step 6: Test the FAQApp Sample Application on 10g Release 3 (10.1.3.1.0)

A.1.3.1 Step 1: Install 10*g* Release 3 (10.1.3.1.0)

Use the following steps to install a new 10*g* Release 3 (10.1.3.1.0) Oracle home. In this particular example, you will install a combined 10*g* Release 3 (10.1.3.1.0) Web server and OC4J instance in a single Oracle home:

1. Log in to the host computer and start Oracle Universal Installer.

   You can install the 10*g* Release 3 (10.1.3.1.0) Oracle home on the same host as the 10*g* Release 2 (10.1.2) J2EE and Web Cache Oracle home, or on a different host.

   > **See Also:** "Starting the Oracle Universal Installer" in the *Oracle Application Server Installation Guide*

2. On the Oracle Application Server 10g 10.1.3.0.0 Installation Screen, do the following:

   a. In the **Installation Directory** field, enter the path to a directory where the application server instance will be installed.

   b. Select **Advanced Install**.

   c. Click Next.

3. When you are prompted to confirm that you want to use the Advanced Install options, click **Next**.

4. On the Select Installation Type page, select **J2EE Server and Web Server**.

   This installation type installs a combined Web server and OC4J Instance in a single Oracle home.

5. Click **Next** and follow the instructions on each screen until you see the Administration Settings page.

6. On the Administration Settings page, be sure to select **Configure this as an Administration OC4J**.

7. Click **Next** and then follow the instructions on each screen to advance through the installation procedure.

   The installation is complete when the End of Installation Screen appears. This screen tells you whether or not your installation was successful. It also provides information about the default Oracle HTTP Server port, as well as the URL you can use to access the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.

A.1.3.2 Step 2: Apply Any Site-Specific Oracle HTTP Server Configuration Settings

If you have made any site-specific configuration changes to the Oracle HTTP Server `httpd.conf` configuration file in the 10*g* Release 2 (10.1.2) Oracle home, apply those changes to the `httpd.conf` file in the 10*g* Release 3 (10.1.3.1.0) Oracle home.

Like the 10*g* Release 2 (10.1.2) release, the 10*g* Release 3 (10.1.3.1.0) Oracle HTTP Server is based on the on Apache 1.3 Web Server. As a result, you can copy any configuration changes (such as modifications to the `httpd.conf` file) directly to the 10*g* Release 3 (10.1.3.1.0) Oracle home.

A.1.3.3 Step 3: Configure the FAQApp Data Source in the 10g Release 3 (10.1.3) Instance

The FAQApp uses a data source that consists of an Oracle database where the FAQ schema has been installed. This procedure assumes that you have successfully

deployed and configured the FAQApp in a previous Oracle Application Server release. As a result, the FAQ schema already exists in an existing Oracle database.

However, before you deploy the FAQApp on the new 10*g* Release 3 (10.1.3.1.0) instance, use the following procedure to configure the `OracleDS` data source, which is required by the FAQApp application, in the new 10*g* Release 3 (10.1.3.1.0) OC4J container.

Note: This procedure modifies the existing `OracleDS` datasource that is defined as part of the default application, which means that the datasource is available to all applications deployed on this instance. Alternatively, you could define the data source using a `data-sources.xml` file within the FAQApp EAR file.

For more information about defining JDBC data sources, see the *Oracle Containers for J2EE Services Guide*.

---

1. Use your browser to display the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.

   The URL for the Application Server Control Console is shown on the End of Installation Screen. This information is also saved to the following file in the 10*g* Release 3 (10.1.3.1.0) Oracle home:

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

   The first page that appears in the Application Server Control Console is the Cluster Topology page.

2. In the **Members** section of the Cluster Topology page, click **home** to display the OC4J Home page for the default `home` instance.

3. From the OC4J Home page, click **Administration**.

4. On the OC4J Administration page, click the task icon in the **JDBC Resources** row of the task table.

   Note that an "OracleDS" data source is created during the Oracle Application Server installation. However, this data source is a managed data source that relies on the "Example Connection Pool" to make its connections to the data base.

5. Click **Example Connection Pool** to display the Edit Connection Pool page.

6. Enter the JDBC URL that provides a connection to the Oracle database that hosts the FAQ schema and click **Apply**.

   Use the following format for the connection URL:

   ```
   jdbc:oracle:thin:@hostname:port:sid
   ```

   For example:

   ```
   jdbc:oracle:thin:@appserv1.acme.com:1521:orcl
   ```

Hint: If you have trouble constructing the JDBC connection URL, you can optionally do the following:

1. Click **Create** in the Connection Pool section of the JDBC Resources page to create a new connection pool.
2. On the Connection Pool - Application page, select the default application, and then select **New Connection Pool**.
3. Click **Continue** to display the Create Connection Pool page. On this page, you can enter information about your database and Application Server Control will construct the JDBC URL for you automatically. From this page, you can also test the connection to be sure it is working before you proceed.
4. Delete the existing `OracleDS` data source and create a new `OracleDS` data source that uses the connection pool you just created.

7. Navigate to the Cluster Topology page and restart the OC4J instance to load the new connection URL for the JDBC connection pool.

   Note that when you restart the `home` instance, Application Server Control is also restarted. As a result, you must wait a few seconds and then enter the URL for the Application Server Control Console again. You can then log into the restarted Application Server Control Console.

A.1.3.4 Step 4: Rebuild the FAQApp EAR File With Required JSP Library Archives

The FAQApp requires the JavaServer Pages (JSP) Standard Tag Libraries. In previous versions of Oracle Application Server, these libraries were automatically available as part of the OC4J instance. In 10*g* Release 3 (10.1.3.1.0), if an application requires the JSP tag libraries, then these two jar files must be included as part of the application EAR file before you deploy the application.

See Also: Section C.2.2, "New Location for JavaServer Pages (JSP) Standard Tag Libraries (JSTL)"

Use the following procedure to locate the JSP Standard Tag Libraries in your existing 10*g* Release 3 (10.1.3.1.0) Oracle home and include them in the FAQApp EAR file:

1. If you have not done so already, unpack the `FAQApp.ear` file into a temporary, working directory.

   Note that a copy of the `FAQApp.ear` file that you deployed on 10*g* Release 2 (10.1.2) should be available in the following directory in the 10*g* Release 2 (10.1.2) Oracle home:

   ```
   (UNIX) 1012_ORACLE_HOME/j2ee/OC4J_Instance/applications/FAQApp.ear
   (Windows) 1012_ORACLE_HOME\j2ee\OC4J_Instance\applications\FAQApp.ear
   ```

   In this example, replace *1012_ORACLE_HOME* with the complete path to the 10*g* Release 2 (10.1.2) Oracle home, and replace *OC4J_Instance* with the name of the 10*g* Release 2 (10.1.2) OC4J instance that you used to deploy the FAQApp application.

2. Locate the following archives in the Oracle home of a 10*g* Release 3 (10.1.3.1.0) installation:

   UNIX:

   ```
   1013_ORACLE_HOME/j2ee/home/default-web-app/WEB-INF/lib/standard.jar
   1013_ORACLE_HOME/j2ee/home/default-web-app/WEB-INF/lib/jstl.jar
   ```

   Windows:

```
1013_ORACLE_HOME\j2ee\home\default-web-app\WEB-INF\lib\standard.jar
1013_ORACLE_HOME\j2ee\home\default-web-app\WEB-INF\lib\jstl.jar
```

3. Copy these two .jar files to the following directory in the FAQApp working directory:

   ```
   faq/lib/
   ```

4. Delete any existing archives (.ear, .war, or other files) from the FAQApp working directory:

   ```
   faq/dist/
   ```

   Note that this step is important because the `ant all` command (which you run in the next step) will not remove or overwrite any existing archive files in the `dist` directory. As a result a new EAR file will not be created if a previous version of the EAR file exists in the `faq/dist` directory.

5. Use Apache Ant to rebuild the jar file using the following command:

   ```
   ant all
   ```

   When you enter this command, Apache Ant compiles and builds the FAQApp application EAR file, based on the information provided in the `build.xml` file that is included in the `faq` directory.

   See Also: Apache Ant Project Web Site at:

   ```
   http://ant.apache.org/
   ```

   Note: This example uses Apache Ant to rebuild the `FAQApp.ear` file. In fact, you can use other tools to perform this task. For more information, see the *Oracle Containers for J2EE Deployment Guide*.

6. Locate the new FAQApp EAR file in the following location in the FAQApp working directory:

   ```
   faq/dist/FAQApp.ear
   ```

A.1.3.5 Step 5: Deploy the Modified FAQApp EAR File on 10g Release 3 (10.1.3)

Use the following procedure to deploy the modified FAQApp sample application on your new 10*g* Release 3 (10.1.3.1.0) OC4J instance.

Note: This procedure describes how to use the Application Server Control Console to deploy the FAQApp sample application. Note, however, that OC4J 10*g* Release 3 (10.1.3.1.0) provides a number of options for deploying your applications.

For more information, see "Deployment Tool Options Provided with OC4J" in the *Oracle Containers for J2EE Deployment Guide*.

1. Navigate to the OC4J Home page in the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.
2. Click **Applications** to display the Applications page.
3. Click **Deploy**.

4. On the Deploy: Select Archive page, click **Browse** and select the modified `FAQApp.ear` file.

   Do not change the other default settings on the page.

5. On the Deploy: Application Attributes page, enter `FAQApp` in the **Application Name** field.

   Do not change the other default settings on the page.

6. On the Deploy: Deployment Settings page, click **Deploy** to deploy the FAQApp application on the 10*g* Release 3 (10.1.3.1.0) OC4J instance.

   Application Server Control displays the progress of the deployment and then displays a message indicating whether or not the deployment was successful.

   If the deployment was not successful, review the progress messages for any specific error messages.

   If any error messages mention SQL or database connections, the problem is likely with the data source configuration. Review the steps you performed in Section A.1.3.3, "Step 3: Configure the FAQApp Data Source in the 10g Release 3 (10.1.3) Instance". Verify that the data source is valid and that the FAQ schema exists in the database. Verify that the database user name you used to connect to the database has access rights to the FAQ schema.

A.1.3.6 Step 6: Test the FAQApp Sample Application on 10*g* Release 3 (10.1.3.1.0)

If the deployment was successful, you can verify that the FAQApp application is working properly by accessing the following URL in your Web browser:

`http://host.domain:port/FAQApp/`

For example:

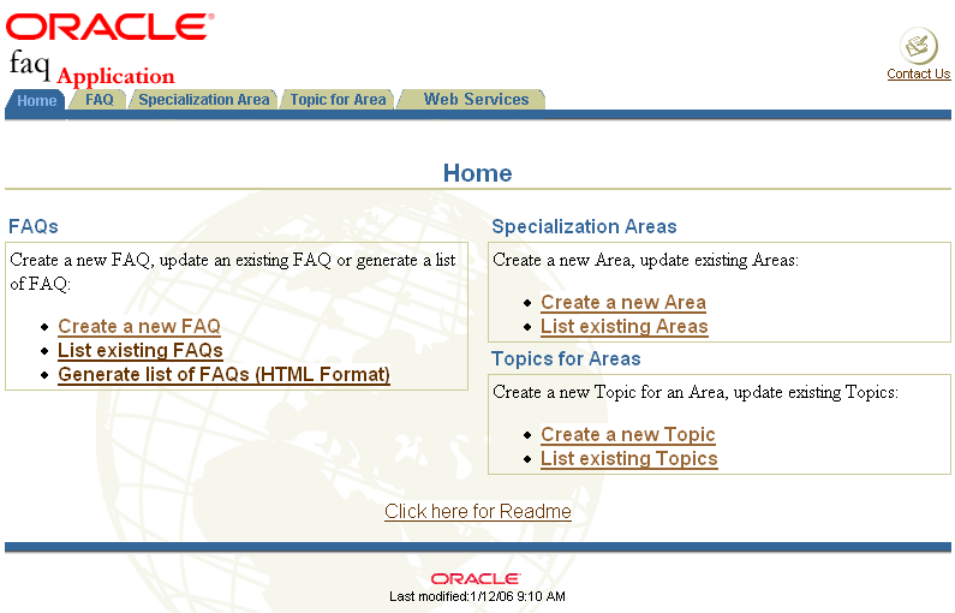
`http://appserv1.acme.com:7779/FAQApp/`

The FAQApp prompts you for a username and password. It is assumed you configured the users and roles for the application when you deployed FAQApp on your previous Oracle Application Server installation.

Enter `faq` as the username and `faq` as the password. The FAQApp home page should appear as shown in Figure A–2. If the application does not display, note any error messages that appear in the browser window.

If any error messages mention SQL or database connections, the problem is likely with the data source configuration. Review the steps you performed in Section A.1.3.3, "Step 3: Configure the FAQApp Data Source in the 10g Release 3 (10.1.3) Instance". Verify that the data source is valid and that the FAQ schema exists in the database. Verify that the database user name you used to connect to the database has access rights to the FAQ schema.

Figure A–2 Successfully Deployed FAQApp Sample Application



A.2 Upgrading FAQApp in a Clustered Environment

The following sections provide an example of upgrading to a 10*g* Release 2 (10.1.2) clustered environment and redeploying an existing application on a new 10*g* Release 3 (10.1.3.1.0) cluster:

- Starting Point for Upgrading FAQApp in a Clustered Environment
- Overview of Upgrading FAQApp in a Clustered Environment
- The FAQApp Upgrade Procedure in a Clustered Environment

A.2.1 Starting Point for Upgrading FAQApp in a Clustered Environment

This procedure assumes the following starting point for the FAQApp upgrade:

- You have installed and configured a 10*g* Release 2 (10.1.2) OracleAS Farm and created an OracleAS Cluster within that farm.

  Note: This procedure describes specifically how to upgrade from 10*g* Release 2 (10.1.2), but the procedure is also valid if you are using 10*g* (9.0.4) as your starting point.

- You have successfully deployed the FAQApp sample application to the 10*g* Release 2 (10.1.2) OracleAS Cluster.

  The FAQApp is available for download from the following location on Oracle Technology Network (OTN):

  `http://www.oracle.com/technology/tech/java/oc4j/demos/904/index.html`

  See Also: "Configuring the FAQ Application Demo" in the *Oracle Application Server Containers for J2EE User's Guide* in the 10*g* Release 2 (10.1.2) documentation library

- You now want to deploy the same FAQApp application on a similar, Oracle Application Server 10*g* Release 3 (10.1.3.1.0) clustered environment.
- You have installed and configured Apache Ant, which is a Java-based build tool that is included in the following directory of your 10*g* Release 3 (10.1.3.1.0) Oracle home:

  ```
  (UNIX) 1013_ORACLE_HOME/ant/
  (Windows) 1013_ORACLE_HOME\ant\
  ```

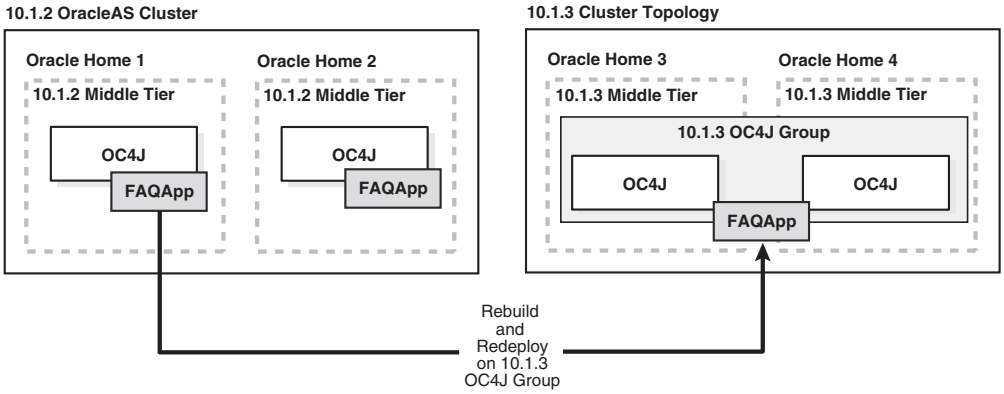
  Apache Ant, as well as documentation for using the tool, is also available from the following Web site:

  ```
  http://ant.apache.org/
  ```

A.2.2 Overview of Upgrading FAQApp in a Clustered Environment

The steps in the following procedure involve installing a new 10*g* Release 3 (10.1.3.1.0) Oracle home, modifying and rebuilding FAQApp, and then redeploying FAQapp on the new 10*g* Release 3 (10.1.3.1.0) OC4J instance, as shown in Figure A–3.

Figure A–3 Overview of Upgrading FAQApp in a Clustered Environment



A.2.3 The FAQApp Upgrade Procedure in a Clustered Environment

The following sections describe the steps you can follow to install and configure a 10*g* Release 3 (10.1.3.1.0) clustered environment and then redeploy the FAQApp on the new cluster:

- Step 1: Install and Configure a 10g Release 3 (10.1.3.1.0) Cluster
- Step 2: Apply Any Site-Specific Oracle HTTP Server Configuration Settings
- Step 3: Create an FAQApp Group
- Step 4: Configure the FAQApp Data Source for the Group
- Step 5: Rebuild the FAQApp EAR File With Required JSP Library Archives
- Step 6: Deploy FAQApp and Configure Application Clustering
- Step 7: Test the FAQApp Sample Application on 10g Release 3 (10.1.3.1.0)

A.2.3.1 Step 1: Install and Configure a 10*g* Release 3 (10.1.3.1.0) Cluster

Use the following steps to install a new 10*g* Release 3 (10.1.3.1.0) clustered environment:

1. For the first middle tier Oracle home, install an Administration OC4J instance.

   After you start Oracle Universal Installer, select **Advanced Install** and then select **J2EE Server and Web Server**.

   During the installation procedure, follow the prompts, ensuring you do the following:

   - In the Administration Instance Settings page, select **Configure this as an Administration OC4J Instance**.
   - In the Administration Settings page, make a note of the `oc4jadmin` password you enter in the **Administrator Account Password** fields; you will need that password later.
   - In the Cluster Topology Configuration screen, select **Configure this HTTP Server instance to be part of an Oracle Application Server cluster topology** and specify a multicast discovery address for the cluster.

     Make a note of the address and port that you enter on this page; you will need it later.

     The multicast address you enter must be within the valid address range, which is `224.0.1.0` to `239.255.255.255`. There are no restrictions on the four-digit port number.

2. For the second middle tier, perform another **J2EE Server and Web Server** advanced installation.

   During the installation procedure, follow the prompts, ensuring you perform the following:

   - In the Administration Instance Settings page, *de*select **Configure this as an Administration OC4J Instance**. Only one administration OC4J is required for each cluster.
   - In the Administration Settings page, enter the same `oc4jadmin` password in the **Administrator Account Password** fields that you entered in Step 1.

     It is important that you enter the same `oc4jadmin` password that you entered during the first installation. In order to use groups, each OC4J instance in the group must have the same `oc4jadmin` password. If they do not, then you will have to reset the password after the installation.
   - In the Cluster Topology Configuration screen, select **Configure this HTTP Server instance to be part of an Oracle Application Server cluster topology** and enter the same multicast address and port you entered in Step 1. Instances that share the same multicast address are automatically configured as a cluster.

     See Also: "Configuring Multiple OC4J Middle Tiers in a Cluster," in the *Oracle Application Server Administrator's Guide* for further information about configuring this topology

3. Verify that the installation and formation of the cluster was successful; do this by using your Web browser to display the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.

   The URL for the Application Server Control Console is shown on the End of Installation Screen. This information is also saved to the following file in the first 10*g* Release 3 (10.1.3.1.0) Oracle home that you installed:

   `(UNIX) ORACLE_HOME/install/readme.txt`

(Windows) *ORACLE_HOME*\install\readme.txt

When you first display the Application Server Control Console, both of the 10*g* Release 3 (10.1.3.1.0) installations should appear on the Cluster Topology page.

See Also: Section B.1, "Using the Cluster Topology Instead of an OracleAS Farm"

A.2.3.2 Step 2: Apply Any Site-Specific Oracle HTTP Server Configuration Settings

If you have made any site-specific configuration changes to the Oracle HTTP Server `httpd.conf` configuration file in the 10*g* Release 2 (10.1.2) OracleAS Cluster, apply those changes to the `httpd.conf` file in each of the 10*g* Release 3 (10.1.3.1.0) Oracle homes.

Like the 10*g* Release 2 (10.1.2) release, the 10*g* Release 3 (10.1.3.1.0) Oracle HTTP Server is based on the on Apache 1.3 Web Server. As a result, you can copy any configuration changes (such as modifications to the `httpd.conf` file) directly to the 10*g* Release 3 (10.1.3.1.0) Oracle homes.

See Also: *Oracle HTTP Server Administrator's Guide*

A.2.3.3 Step 3: Create an FAQApp Group

Within a 10*g* Release 3 (10.1.3.1.0) cluster, you can organize multiple OC4J instances into a single group. You can then perform specific deployment and configuration tasks on the group.

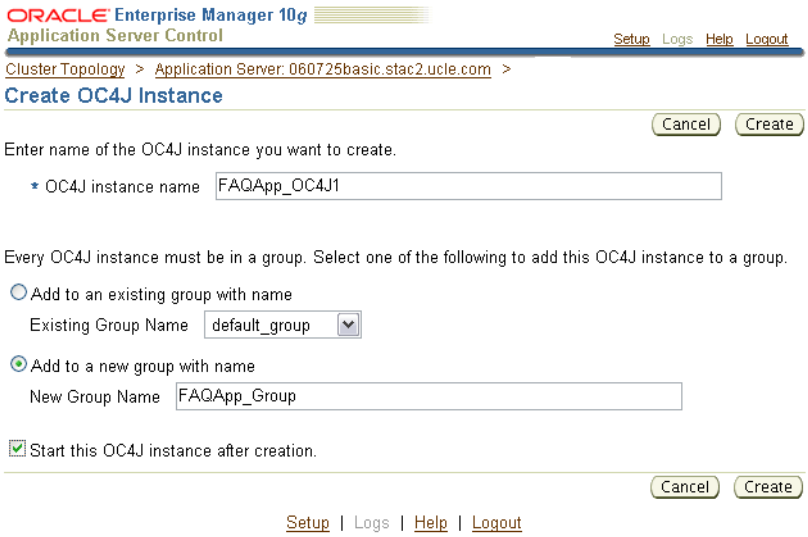
To create a group for the FAQApp:

1. Navigate to the Cluster Topology page in the Application Server Control Console.
2. In the Members section of the page, click the name of the first application server instance you installed.
3. From the Application Server page, click **Create OC4J Instance**.

   Application Server Control displays the Create OC4J Instance page (Figure A–4).
4. On the Create OC4J Instance page, do the following:
   a. Enter `FAQApp_OC4J1` in the **OC4J instance name** field.
   b. Select **Add to a new group with name**.
   c. Enter FAQApp_Group in the **New Group Name** field.
   d. Select **Start this OC4J instance after creation**.
   e. Click **Create**.
5. Return to the Cluster Topology page and click the name of the second application server instance you installed.
6. From the Application Server page, click **Create OC4J Instance**.
7. On the Create OC4J Instance page, do the following:
   a. Enter `FAQApp_OC4J2` in the **OC4J instance name** field.
   b. Select **Add to an existing group with name**.
   c. Select **FAQApp_Group** from the **Existing Group Name** drop-down menu.
   d. Select **Start this OC4J instance after creation**.
   e. Click **Create**.

8. Return to the Cluster Topology page and scroll to the Groups section of the page.
9. Note that the new FAQApp group contains both the FAQApp_OC4J1 instance and the FAQApp_OC4J2 instance.

Figure A–4 Create OC4J Instance Page



A.2.3.4 Step 4: Configure the FAQApp Data Source for the Group

The FAQApp uses a data source that consists of an Oracle Database where the FAQ schema has been installed. This procedure assumes that you have successfully deployed and configured the FAQApp in a previous Oracle Application Server release. As a result, the FAQ schema already exists in an existing Oracle database.

However, before you deploy the FAQApp on the 10*g* Release 3 (10.1.3.1.0) group, you must use the following procedure to configure the `OracleDS` data source, which is required by the FAQApp application.

Use the following procedure to define the data source for all the OC4J instances in the FAQApp group:

1. From the Cluster Topology page in the Application Server Control Console, scroll to the Groups section of the page.
2. Click the name of the FAQApp group.
3. From the Group page, click **Administration**.
4. On the Group Administration page, click the task icon in the **JDBC Resources** row of the task table.

   Note that an "OracleDS" data source has already been created. However, this data source is a managed data source that relies on the "Example Connection Pool" to make its connections to the data base.
5. Click **Example Connection Pool** to display the Edit Connection Pool page.
6. Enter the JDBC URL that provides a connection to the Oracle database that hosts the FAQ schema and click **Apply**.

   Use the following format for the connection URL:

   `jdbc:oracle:thin:@`*`hostname`*`:`*`port`*`:`*`sid`*

For example:

```
jdbc:oracle:thin:@appserv1.acme.com:1521:orcl
```

Hint: If you have trouble constructing the JDBC connection URL, you can optionally do the following:

1. Click **Create** in the Connection Pool section of the JDBC Resources page to create a new connection pool.
2. On the Connection Pool - Application page, select the default application, and then select **New Connection Pool**.
3. Click **Continue** to display the Create Connection Pool page. On this page, you can enter information about your database and Application Server Control will construct the JDBC URL for you automatically. From this page, you can also test the connection to be sure it is working before you proceed.
4. Delete the existing OracleDS data source and create a new `OracleDS` data source that uses the connection pool you just created.

7. Restart the FAQApp group to load the new JDBC connection pool URL:
   a. Navigate to the Cluster Topology page.
   b. Select the FAQApp group.
   c. Click **Stop**.
   d. After Application Server Control confirms that the group has been stopped, select the FAQApp group again and click **Start**.

A.2.3.5 Step 5: Rebuild the FAQApp EAR File With Required JSP Library Archives

The FAQApp requires the JavaServer Pages (JSP) Standard Tag Libraries. In previous versions of Oracle Application Server, these libraries were automatically available as part of the OC4J instance. In 10*g* Release 3 (10.1.3.1.0), if an application requires the JSP tag libraries, then these two jar files must be included as part of the application EAR file before you deploy the application.

See Also: Section C.2.2, "New Location for JavaServer Pages (JSP) Standard Tag Libraries (JSTL)"

Use the following procedure to locate the JSP Standard Tag Libraries in your existing 10*g* Release 3 (10.1.3.1.0) Oracle home and include them in the FAQApp EAR file:

1. If you have not done so already, unpack the `FAQApp.ear` file into a temporary, working directory.

   Note that a copy of the FAQApp.ear file that you deployed on 10*g* Release 2 (10.1.2) should be available in the following directory in the 10*g* Release 2 (10.1.2) Oracle home:

   ```
   (UNIX) 1012_ORACLE_HOME/j2ee/OC4J_Instance/applications/FAQApp/FAQApp.ear
   (Windows) 1012_ORACLE_HOME\j2ee\OC4J_Instance\applications\FAQApp\FAQApp.ear
   ```

   In this example, replace *1012_ORACLE_HOME* with the complete path to the 10*g* Release 2 (10.1.2) Oracle home, and replace *OC4J_Instance* with the name of the OC4J instance that you used to deploy the FAQApp application.

2. Locate the following archives in the Oracle home of a 10*g* Release 3 (10.1.3.1.0) installation:

   UNIX:

```
1013_ORACLE_HOME/j2ee/home/default-web-app/WEB-INF/lib/standard.jar
1013_ORACLE_HOME/j2ee/home/default-web-app/WEB-INF/lib/jstl.jar
```

Windows:

```
1013_ORACLE_HOME\j2ee\home\default-web-app\WEB-INF\lib\standard.jar
1013_ORACLE_HOME\j2ee\home\default-web-app\WEB-INF\lib\jstl.jar
```

3. Copy these two .jar files to the following directory in the FAQApp working directory:

   ```
   faq/lib/
   ```

4. Delete any existing archives (.ear, .war, or other files) from the FAQApp working directory:

   ```
   faq/dist/
   ```

   Note that this step is important because the `ant all` command (which will be used in the next step) will not remove or overwrite any existing archive files in the dist directory. As a result a new EAR file will not be created if a previous version of the EAR file exists in the `faq/dist` directory.

5. Use Apache Ant to rebuild the jar file using the following command:

   ```
   ant all
   ```

   When you enter this command, Apache Ant compiles and builds the FAQApp application EAR file, based on the information provided in the build.xml file that is included in the `faq` directory.

   See Also: Apache Ant Project Web Site at:

   ```
   http://ant.apache.org/
   ```

   Note: This example uses Apache Ant to rebuild the `FAQApp.ear` file. In fact, you can use other tools to perform this task. For more information, see the *Oracle Containers for J2EE Deployment Guide*.

6. Locate the new FAQApp EAR file in the following location in the FAQApp working directory:

   ```
   faq/dist/FAQApp.ear
   ```

A.2.3.6 Step 6: Deploy FAQApp and Configure Application Clustering

Use the following procedure to deploy the modified FAQApp sample application on the 10*g* Release 3 (10.1.3.1.0) FAQApp group.

During the deployment, you can also configure application clustering for the FAQApp, which enables session state replication across the application instances in the cluster. Examples of session state information include whether or not a user is logged in and the contents of a shopping cart.

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) offers three protocols for session state replication. In this example, you enable the peer-to-peer replication protocol. When you select this protocol, OPMN automatically replicates state information to the other OC4J instances in the 10*g* Release 3 (10.1.3.1.0) cluster.

See Also: Section B.3, "Using Application Clustering for State Replication"

To deploy the FAQApp on the FAQApp group and configure application clustering:

1. Navigate to the Cluster Topology page in the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.
2. Click the name of the FAQApp group to display the FAQApp Group page.
3. Click **Applications** to display the Group Applications page.
4. Click **Deploy**.
5. On the Deploy: Select Archive page, click **Browse** and select the updated `FAQApp.ear` file.

   Do not change the other default settings on the page.
6. On the Deploy: Application Attributes page, enter `FAQApp` in the **Application Name** field.

   Do not change the other default settings on the page.
7. On the Deploy: Deployment Settings page, click **Configure Clustering** and use the resulting page to enable session state replication for the FAQApp application:
   a. Select **Override parent application clustering settings**.
   b. Select **Enable** from the **Clustering** drop-down menu.
   c. Select **Peer-Peer Replication**; do not enter a value in the **Bind Address** field unless the host computer has multiple network cards.
   d. Click **OK**.
8. Click **Deploy** to deploy FAQApp to both OC4J instances in the FAQApp group.

   Application Server Control displays the progress of the deployment and then displays a message indicating whether or not the deployment was successful.

   If the deployment was not successful, review the progress messages for any specific error messages.

   If any error messages mention SQL or database connections, the problem is likely with the data source configuration. Review the steps you performed in Section A.2.3.4, "Step 4: Configure the FAQApp Data Source for the Group". Verify that the data source is valid and that the FAQ schema exists in the database. Verify that the database user name you used to connect to the database has access rights to the FAQ schema.

A.2.3.7 Step 7: Test the FAQApp Sample Application on 10*g* Release 3 (10.1.3.1.0)

If the deployment was successful, you can verify that the FAQApp application is working properly by accessing the following URL in your Web browser:

```
http://host.domain:port/FAQApp/
```

For example:

```
http://appserv1.acme.com:7779/FAQApp/
```

The FAQApp prompts you for a username and password. It is assumed you configured the users and roles for the application when you deployed FAQApp on your previous Oracle Application Server installation.

Enter `faq` as the username and `faq` as the password. The FAQApp home page should appear as shown in Figure A–2. If the application does not display, note any error messages that appear in the browser window.

If any error messages mention SQL or database connections, the problem is likely with the data source configuration. Review the steps you performed in Section A.2.3.4, "Step 4: Configure the FAQApp Data Source for the Group". Verify that the data source is valid and that the FAQ schema exists in the database. Verify that the database user name you used to connect to the database has access rights to the FAQ schema.

B

Differences Between 10*g* Release 3 (10.1.3.1.0) and Previous Releases

Use this appendix to learn about the key differences between Oracle Application Server 10*g* Release 3 (10.1.3.1.0) and the following Oracle Application Server 10*g* releases:

- Oracle Application Server 10*g* (9.0.4)
- Oracle Application Server 10*g* Release 2 (10.1.2)

> **Note:** For information about the differences between 10*g* Release 3 (10.1.3.1.0) and 10*g* Release 3 (10.1.3.0.0), refer to:
>
> - The *Oracle Application Server Patchset Notes 10g Release 3 (10.1.3.1.0) Patch Set 1 (10.1.3.1.0) for Linux and Microsoft Windows*
> - Section 2.1.2, "New Features Provided with 10g Release 3 (10.1.3.1.0)".

This chapter contains the following sections:

B.1 Using the Cluster Topology Instead of an OracleAS Farm

With previous releases of Oracle Application Server, you can configure a set of Oracle Application Server instances so they use a common OracleAS Metadata Repository. The instances that share the common OracleAS Metadata Repository are members of the same OracleAS Farm. From the Farm page in the 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) Application Server Control Console, you can view all the application servers that are members of the OracleAS Farm. The Distributed Configuration Management (DCM) software provides the underlying technology for managing the farm.

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) does not include an OracleAS Metadata Repository or the DCM software. As a result, there is no concept of an OracleAS Farm. Instead, in 10*g* Release 3 (10.1.3.1.0), you configure your application

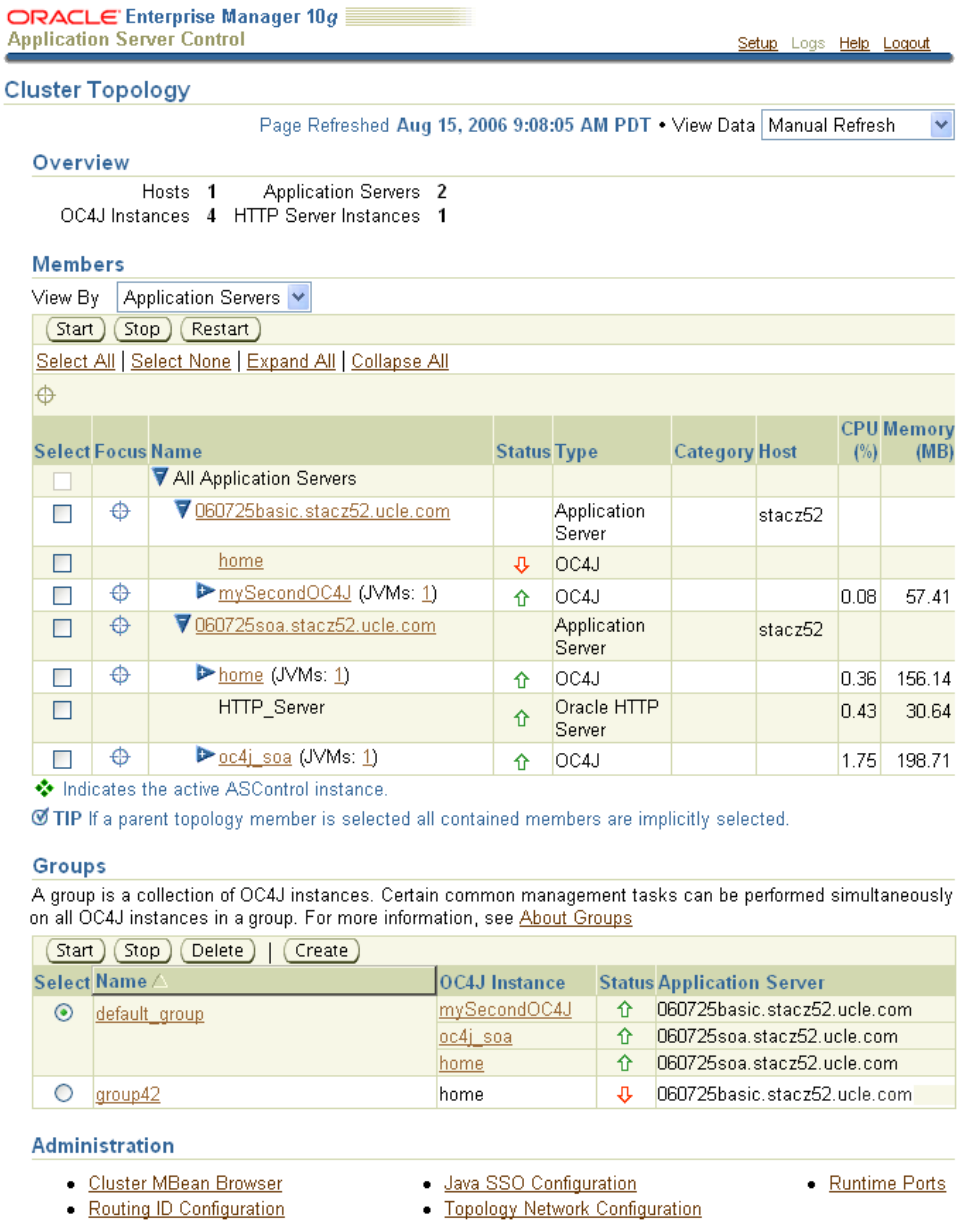
server instances so they can communicate via Oracle Process Manager and Notification Server (OPMN).

When you configure two or more 10*g* Release 3 (10.1.3.1.0) instances in this manner, the instances can be managed from the Cluster Topology page in the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.

Figure B–1 shows the 10*g* Release 3 (10.1.3.1.0) Cluster Topology page, which includes two Oracle Application Server instances that have been configured to communicate via the same multicast address and port.

> **See Also:** "Configuring and Managing Clusters" in the *Oracle Containers for J2EE Configuration and Administration Guide*
>
> "Configuring the Cluster Topology" in the in the Application Server Control online help

Figure B–1 Oracle Application Server 10g Release 3 (10.1.3.1.0) Cluster Topology Page



B.2 Using Groups Instead of OracleAS Clusters

With previous releases of Oracle Application Server, you can create and manage OracleAS Clusters. OracleAS Clusters consist of identically configured J2EE and Web Cache installations that are part of the same OracleAS Farm. Distributed Configuration Management (DCM) is then used to keep the instances within the cluster in synch. Configuration changes made to one instance in the cluster are automatically applied to other instances in the cluster.

In 10*g* Release 3 (10.1.3.1.0), there is no OracleAS Farm and there is no DCM. However, you can still group multiple Oracle Containers for J2EE (OC4J) instances that are part of the same cluster topology. These groups of OC4J instances can be used in a similar manner to OracleAS Clusters.

Refer to the following sections for more information:

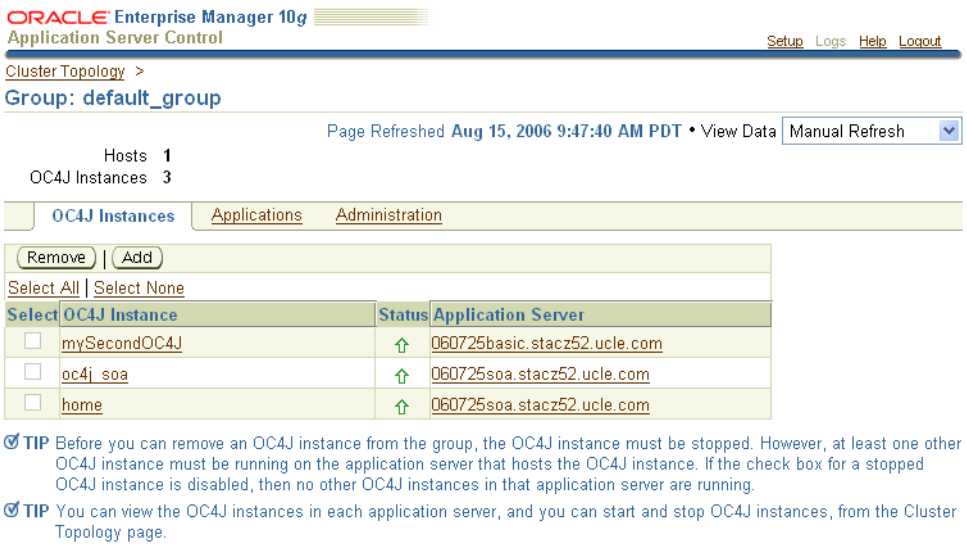
- How Are Groups Similar to OracleAS Clusters?
- How Are Groups Different from OracleAS Clusters?

B.2.1 How Are Groups Similar to OracleAS Clusters?

Like OracleAS Clusters, groups make it easy to deploy your applications to more than one OC4J instance at a time:

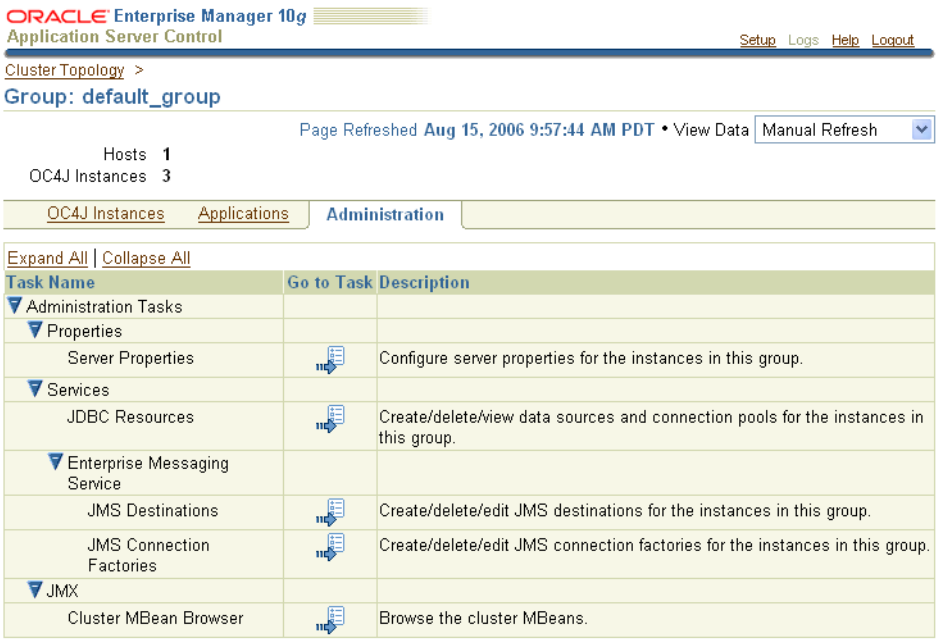
- With OracleAS Clusters, changes made to one instance in a cluster are automatically propogated to other instances in the cluster. For example, if you deploy an application to one instance in the cluster, the application is automatically deployed to the other instances.
- With groups, you deploy your J2EE applications to all the OC4J instances in the group using the Group page (Figure B–2). The Group page is available from the Cluster Topology page.

Figure B–2 Oracle Application Server 10g Release 3 (10.1.3.1.0) Group Page



Like OracleAS Clusters, groups allow you to make specific configuration changes to all the OC4J instances in the group. For example, you can use the Group Administration page (Figure B–3) to set server properties, configure JDBC data sources, configure the Enterprise Messaging Service, and to access the cluster MBean browser.

Figure B–3 Oracle Application Server 10g Release 3 (10.1.3.1.0) Group Administration Page



B.2.2 How Are Groups Different from OracleAS Clusters?

The following sections describe some key differences between 10*g* Release 3 (10.1.3.1.0) groups and OracleAS Clusters:

- Configuration Changes Are Not Propogated Automatically to Other Members of the Group
- OC4J Instances Within a Group Are Not Always Identical

B.2.2.1 Configuration Changes Are Not Propogated Automatically to Other Members of the Group

Unlike OracleAS Clusters, configuration changes made to an individual OC4J instance in a group (from the OC4J instance Home page or from the command line) are not automatically applied to other OC4J instances in the cluster.

Instead, if you want to make a configuration change to all the OC4J instances in a group, you must either make the change from the Group page, or you must use the Application Server Control Console or command line tools to make the change to each OC4J instance in the cluster.

Similarly, if you add a new OC4J instance to the group, configuration changes are not automatically applied to the new instance. Instead, you must use the OC4J home page for that instance, or the command-line tools, to apply any required configuration changes to the new instance.

> **See Also:** "Replicating Changes Across a Cluster" in the *Oracle Containers for J2EE Configuration and Administration Guide*

B.2.2.2 OC4J Instances Within a Group Are Not Always Identical

In some ways, groups provide more flexibility than OracleAS Clusters. For example, when you add an OC4J instance to an cluster in 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), the

instance can be used for cluster operations only. Any changes you make to the instance are automatically applied to the other instances in the cluster.

In 10*g* Release 3 (10.1.3.1.0), you have the flexibility to deploy an application to just one OC4J instance in the group, or to adjust the attributes of one instance without impacting the other instances in the group.

Note, however, that if you make changes to one member of the group and not to another, some operations performed on the group could succeed on one instance and fail on another.

Also, unlike OracleAS Clusters, actions you perform on a group do not affect OC4J instances within the group that are not up and running when the operation is performed.

B.3 Using Application Clustering for State Replication

In addition to clusters and groups, Oracle Application Server 10*g* Release 3 (10.1.3.1.0) introduces the concept of application clustering, which provides state replication and load balancing for applications within your cluster topology.

The following sections provide more information:

- Clustering Features and Concepts That Are No Longer Supported
- About 10g Release 3 (10.1.3.1.0) Application Clustering

B.3.1 Clustering Features and Concepts That Are No Longer Supported

Application Clustering provides a simpler, more efficient method of replicating application state, which replaces the following concepts and features that are no longer supported in 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0):

- Islands

  In previous releases, an island was essentially a group of OC4J instances within a cluster across which HTTP session data was replicated. Although islands reduced overhead by not replicating data across the entire cluster, they increased configuration and management overhead. In addition, islands were only applicable to Web applications; EJB applications could not utilize the island configuration.

  In 10*g* Release 3 (10.1.3.1.0), you replicate HTTP session data using Application Clustering, which can be configured using the Application Server Control Console during or after the deployment of your application.

- loadbalancer.jar

  The `loadbalancer.jar` file, which provided load balancing functionality in previous OC4J releases, was deprecated in the previous release of OC4J and has been removed from the current release.

- Deprecated Clustering-Specific XML Elements

  The following XML elements are deprecated in OC4J 10g (10.1.3) and should no longer be used to configure clustering. The new `<cluster>` element is now used for all cluster management:

  – The `<cluster-config>` element in `server.xml`

  – The `cluster-island` attribute of the `<web-site>` element in the `*-web-site.xml` configuration file

B.3.2 About 10*g* Release 3 (10.1.3.1.0) Application Clustering

Within a 10*g* Release 3 (10.1.3.1.0) cluster, you can configure clustering for selected applications that are deployed across the cluster. Application clustering offers the following features:

- You can configure clustering for specific applications, or globally by configuring clustering for the `default` application in an OC4J instance.

  Other applications deployed to the instance automatically inherit the clustering characteristics of the `default` application.
- You can configure clustering for an application at deployment time, or later, after you deploy the application.
- You can select from the following replication methods:
  - Peer-to-peer replication
  - Multicast replication
  - Database replication

See Also: "Application Clustering in OC4J" in the *Oracle Containers for J2EE Configuration and Administration Guide* for more detailed information about the supported replication methods

B.4 Using the admin_client.jar Utility to Manage OC4J Instances, Groups, and Clusters

OC4J 10*g* Release 3 (10.1.3.1.0) also provides a command-line utility— `admin_client.jar`—that can be used to perform operations on active OC4J instances.

For many functions, the `admin_client.jar` utility replaces the `admin.jar` utility, which is used exclusively for the standalone configuration of the 10*g* Release 3 (10.1.3.1.0) OC4J server.

Unlike the `admin.jar` utility, you can use the `admin_client.jar` utility to manage OC4J instances in a managed, Oracle Application Server environment, as well as OC4J instances in a standalone OC4J environment.

You can perform the following tasks with the `admin_client.jar` utility:

- Deploy applications to a specific OC4J instance or to all instances within a cluster
- Undeploy an application
- Incrementally update a deployed EJB module with modified classes
- Create a new shared library for a specific OC4J instance or for all instances within a cluster
- Create data sources and JMS queues and topics
- Stop, start or restart a specific application, on a specific OC4J instance or cluster-wide
- Manage the creation, deletion, and membership of OC4J groups

See Also: "Using the admin_client.jar Utility" in the *Oracle Containers for J2EE Configuration and Administration Guide*

B.5 Summary of Equivalent Features in 10*g* Release 3 (10.1.3.1.0)

Table B–1 describes how some common Oracle Application Server management tasks were performed in Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2). It then shows how those same tasks are performed in 10*g* Release 3 (10.1.3.1.0).

See Also: Appendix D, "Differences Between 10g Release 3 (10.1.3.1.0) and 10g Release 3 (10.1.3.0.0)"

Table B–1 Summary of Changed Features for 10g Release 3 (10.1.3.1.0)

Task or Feature	In Prior Releases...	In 10*g* Release 3 (10.1.3.1.0)...
Clustering Oracle Application Server instances	Configure multiple Oracle Application Server instances so they use the same OracleAS Metadata Repository. This creates an OracleAS Farm, which can be viewed from the Application Server Control Console Farm page.	Use the Topology Network Configuration page to configure the cluster, or perform the equivalent task during the installation. This causes the selected Oracle Application Server instances to appear on the Cluster Topology page of the Application Server Control Console.
Performing management tasks simultaneously on multiple OC4J instances	Add selected J2EE and Web Cache instances within an OracleAS Farm to an OracleAS Cluster. Perform this task from the Farm page in the Application Server Control Console.	Create multiple OC4J instances and organzie them into a group. Use the Group page in the Application Server Control Console to manage the group and to perform administration tasks on all OC4J instances in the group.
Replicating application state across a cluster	OC4J processes and islands within OracleAS Clusters.	Application clustering, which can be configured from the Application Server Control Console during deployment or post-deployment.
Creating new OC4J instances	Click **Create Instance** on the OC4J Home page in the Application Server Control Console.	Click Create OC4J Instance on the Application Server page, or use the `createinstance` command in the `bin` directory of the Oracle Application Server Oracle home.
Using command-line tools to manage instances and clusters	Use one of the following: ▪ Distributed Configuration Management (DCM) command line (dcmctl) ▪ Oracle Process Manager and Notification Server (OPMN) command line (opmnctl)	DCM is not available in 10*g* Release 3 (10.1.3.1.0), but new `opmnctl` commands and the new `admin_client.jar` utility provide additional management capabilities.
Using OracleAS Identity Management	Configure OracleAS Identity Management using the Application Server Infrastructure page in the Application Server Control Console.	Configure OracleAS Identity Management using the Identity Management task on the OC4J Administration page in the Application Server Control Console.

C

Things to Consider When Redeploying 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) Applications

If you are upgrading from Oracle Application Server 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), Oracle Application Server 10*g* Release 3 (10.1.3.1.0) introduces support for the the latest J2EE 1.4 technologies and application programming interfaces (APIs). These new capabilities allow you to deploy J2EE applications that take advantage of the newest J2EE features and capabilities.

This chapter provides important considerations to review before you deploy your Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) applications on Oracle Application Server 10*g* Release 3 (10.1.3.1.0).

Note that many of these features were introduced with Oracle Application Server 10*g* Release 3 (10.1.3.0.0), but they also apply to 10*g* Release 3 (10.1.3.1.0).

This chapter includes the following sections:

C.1 Overview of Redeploying Applications on 10*g* Release 3 (10.1.3.1.0)

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) supports functionality outlined in the J2EE Application Deployment API (JSR-88), which defines a standard API for configuring and deploying J2EE applications and modules into a J2EE-compatible environment.

Specifically, the JSR-88 compliant features in OC4J provide the ability to:

- Start an application immediately upon deployment, making it available to clients

- Stop an application, making it unavailable to clients
- Undeploy an application or module
- Redeploy an application or module, essentially updating the currently installed application with an updated version
- Create a deployment plan containing the aggregated OC4J-specific configuration data needed to deploy a component into OC4J.

  See Also: "Working With Deployment Plans" in the *Oracle Containers for J2EE Deployment Guide* for details on the JSR-88 implementation in OC4J.

To deploy an application, you use one of two management tools:

- The new Application Server Control Console provided with 10*g* Release 3 (10.1.3.1.0)

  See Also: "Introduction to Administration Tools" in the *Oracle Application Server Administrator's Guide*

- The `admin_client.jar` command-line utility, which is new for 10*g* Release 3 (10.1.3.1.0)

  See Also: Section B.4, "Using the admin_client.jar Utility to Manage OC4J Instances, Groups, and Clusters"

For complete information about deploying your J2EE applications on 10*g* Release 3 (10.1.3.1.0), see the *Oracle Containers for J2EE Deployment Guide*.

For a step-by-step example of upgrading to 10*g* Release 3 (10.1.3.1.0) and redeploying the FAQApp sample application on 10*g* Release 3 (10.1.3.1.0), see Appendix A, "Step-By-Step Upgrade Examples".

C.2 General Considerations

The following sections describe general information that you should consider before redeploying your applications on 10*g* Release 3 (10.1.3.1.0):

- Classloading and Shared Library Support
- New Location for JavaServer Pages (JSP) Standard Tag Libraries (JSTL)
- Oracle JSP Markup Language (JML) Tag Library No Longer Supported
- New OC4J Instances No Longer Include http-web-site

C.2.1 Classloading and Shared Library Support

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced significant improvements in the areas of class loading and shared library support. These improvements are also available in 10*g* Release 3 (10.1.3.1.0).

For complete information about how you can take advantage of these new features, see "Utilizing the OC4J Class Loading Framework" in the *Oracle Containers for J2EE Developer's Guide*.

That chapter in the *Oracle Containers for J2EE Developer's Guide* contains an overview of the new class loading framework, information about using shared libraries, as well as classloading best practices and troubleshooting information.

C.2.2 New Location for JavaServer Pages (JSP) Standard Tag Libraries (JSTL)

In previous versions of Oracle Application Server, the JavaServer Pages (JSP) Standard Tag Libraries were automatically available as part of the OC4J instance. However, application developers often want to include their own custom version of the libraries, or a newer version of the tag libraries. In previous versions of OC4J, errors could result if you included custom tag libraries in addition to the pre-packaged libraries.

As a result, for 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0), the tag libraries (`standard.jar` and `jstl.jar`) are now installed in a new location in the Oracle Application Server Oracle home. If your application depends upon these libraries, you must now include the tag libraries in the `WEB-INF/lib` directory of your application EAR file.

Specifically, the libraries are now installed in the following directory. You can copy these libraries from this location and include them into your application before you deploy the application on 10*g* Release 3 (10.1.3.1.0):

1013_ORACLE_HOME`/j2ee/home/default-web-app/WEB-INF/lib`

> **See Also:** "Support for the JavaServer Pages Standard Tag Library" in the *Oracle Containers for J2EE JSP Tag Libraries and Utilities Reference*

C.2.3 Oracle JSP Markup Language (JML) Tag Library No Longer Supported

The Oracle JSP Markup Language (JML) tag library is officially de-supported as of Oracle Application Server 10*g* Release 3 (10.1.3.0.0). This applies to 10*g* Release 3 (10.1.3.1.0) as well.

Developers are advised to use tags provided with the JavaServer Pages Standard Tag Library (JSTL), which provide similar functionality in a standardized implementation.

For more information, see Chapter 2, "Support for the JavaServer Pages Standard Tag Library" in the *Oracle Containers for J2EE Support for JavaServer Pages Developer's Guide*.

C.2.4 New OC4J Instances No Longer Include http-web-site

In previous versions of Oracle Containers for J2EE, newly created OC4J instances included a predefined Web site named `http-web-site`. For 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0), the `http-web-site` has been removed.

As as a result, if you are deploying an application that was previously configured to bind to the `http-web-site`, you must modify the application to bind to the new `default-web-site`, or create a new Web site for the OC4J instance called `http-web-site`.

The configuration file for the `default-web-site` is stored in the following location within a 10*g* Release 3 (10.1.3.1.0) Oracle home:

On UNIX systems:

ORACLE_HOME`/j2ee/`*OC4J_instance*`/config/default-web-site.xml`

On Windows systems:

ORACLE_HOME`/j2ee/`*OC4J_instance*`/config/default-web-site.xml`

In this example, replace *OC4J_instance* with the name of the OC4J instance.

> **See Also:** "New and Changed Features in OC4J" in the *Oracle Containers for J2EE Configuration and Administration Guide*

C.3 Data Source Considerations

The following sections provide information about using data sources in 10*g* Release 3 (10.1.3.1.0):

- New Features for Data Sources in 10g Release 3 (10.1.3.1.0)
- Converting data-sources.xml to the New 10g Release 3 (10.1.3.1.0) Format
- Using Oracle JDBC-OCI Drivers with 10g Release 3 (10.1.3.1.0)

C.3.1 New Features for Data Sources in 10*g* Release 3 (10.1.3.1.0)

The following OC4J Data Source features and behaviors were introduced in the 10*g* Release 3 (10.1.3.0.0) release and are also available for 10*g* Release 3 (10.1.3.1.0):

- Data source configuration can be performed entirely in the Oracle Enterprise Manager 10g Application Server Control Console.
- The OC4J Data Source types are **managed data sources** and **native data sources**, replacing emulated, non-emulated, and native.
- New connection caching mechanism that is uniform across Oracle data sources and offers integrated Real Application Clusters (RAC) failover support.

> **See Also:** "Data Sources" in the *Oracle Containers for J2EE Services Guide*
>
> "Managing Data Sources and JDBC Connection Pools" in the Application Server Control online help

C.3.2 Converting data-sources.xml to the New 10*g* Release 3 (10.1.3.1.0) Format

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced a new format for the `data-sources.xml` file, which defines the data sources for your application, OC4J instance, or group. This change also applies to 10*g* Release 3 (10.1.3.1.0).

However, you can still use your existing `data-source.xml` files. OC4J will convert the data sources to the new format at runtime. Note, however, that if you deploy an EAR file that contains a `data-sources.xml` file in the previous format, OC4J will convert the `data-sources.xml` file that is expanded on disk. It will not modify the `data-sources.xml` file contained within the EAR file.

Note that when you use the Application Server Control Console to create or modify your OC4J data sources, Application Server Control saves the updates in the `data-sources.xml` using the new format.

Alternatively, if you are using standalone OC4J, you can use the `admin.jar` utility to convert the `data-sources.xml` file to the new format.

Note: The `admin.jar` utility can only be used to manage a single OC4J instance in a standalone OC4J installation.

For more information, see "Converting Existing Data Sources to the New Configuration" in the *Oracle Containers for J2EE Configuration and Administration Guide*.

C.3.3 Using Oracle JDBC-OCI Drivers with 10*g* Release 3 (10.1.3.1.0)

If your existing applications use the Oracle JDBC Oracle Call Interface (OCI) driver, be sure to review the section, "Oracle JDBC Drivers" in the *Oracle Containers for J2EE Services Guide* for information on configuration and upgrade requirements.

C.4 Web Services Considerations

For backward compatibility, Oracle Application Server 10*g* Release 3 (10.1.3.1.0) includes the underlying software required to run 10*g* Release 2 (10.1.2) Web services. As a result, Web services applications designed and packaged to run with Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) can be used without modification with Release 3.

However, there are significant advantages to recreating your Web services for 10*g* Release 3 (10.1.3.1.0). For complete information on creating Web services for 10*g* Release 3 (10.1.3.1.0), refer to the *Oracle Application Server Web Services Developer's Guide*.

In addition, refer to the following sections for specific considerations when using 10*g* Release 3 (10.1.3.1.0) to recreate Web services that were originally created against 10*g* (9.0.4) or 10*g* Release 2 (10.1.2):

- New Web Services Assembler (wsa.jar)
- Assembling Web Services From Java Classes in 10g Release 3 (10.1.3.1.0)
- Developing Database Web Services in 10g Release 3 (10.1.3.1.0)
- Assembling Web Services from an Existing WSDL File

C.4.1 New Web Services Assembler (wsa.jar)

If you re-create your Web services for the 10*g* Release 3 (10.1.3.1.0), note that the Web Services Assembler tool for 10*g* Release 3 (10.1.3.1.0) is now called `wsa.jar`, and it is not compatible with the Web Services Assembler tool used for previous releases (`WebServicesAssembler.jar`). Web services and clients created with `wsa.jar` will be different and incompatible with Web services created with `WebServicesAssembler.jar`.

> **See Also:** "Using WebServicesAssembler" in the *Oracle Application Server Web Services Developer's Guide*

C.4.2 Assembling Web Services From Java Classes in 10*g* Release 3 (10.1.3.1.0)

If you created a Web service based on a Java class for Oracle Application Server 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), you can do the same using the new Web Services Assembler (`wsa.jar`) available with 10*g* Release 3 (10.1.3.1.0). However, you must be aware of the following:

- In 10*g* Release 2 (10.1.2), it was possible to publish a class by itself without providing an interface. In 10*g* Release 3 (10.1.3.1.0), you must provide an interface (specifically, the Service Endpoint Interface) to publish a class.

  > **See Also:** "Writing Java Class-Based Web Services" in the *Oracle Application Server Web Services Developer's Guide*

- The set of Java types that are natively supported has changed with the 10*g* Release 3 (10.1.3.1.0) release. For a list of the supported data types, see the JAX-RPC 1.1 specification available from the following URL:

  `http://java.sun.com/xml/jaxrpc/index.jsp`

  See Also: "Assembling a Web Service with Java Classes" in the *Oracle Application Server Web Services Developer's Guide*

C.4.3 Developing Database Web Services in 10*g* Release 3 (10.1.3.1.0)

If you created database Web services with Oracle Application Server 10*g* (9.0.4) or 10*g* Release 2 (10.1.2), you can do the same using the new Web Services Assembler (`wsa.jar`) available with 10*g* Release 3 (10.1.3.1.0).

Note, however, that in 10*g* (9.0.4) and in 10*g* Release 2 (10.1.2), database Web services were always created using the RPC-encoded message format. In 10*g* Release 3 (10.1.3.1.0), database Web services are by default created using the document-literal message format.

See Also: "Supported Message Formats" in the *Oracle Application Server Web Services Developer's Guide*

As a result, if you use the RPC-encoded message format when you create a 10*g* Release 3 (10.1.3.1.0) database Web service, the Web service will not be interchangeable between 10*g* Release 3 (10.1.3.1.0) and previous Oracle Application Server Web services clients.

Specifically, a Web service client written for a database Web service generated under 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) will fail if you try to use it against a database Web service generated under 10*g* Release 3 (10.1.3.1.0). This will be true even if the PL/SQL structures have remained the same.

One of the reasons for this is that the SQL collection type was mapped into a complex type with a single array property in 10*g* (9.0.4) and 10*g* Release 2 (10.1.2). In release 10*g* Release 3 (10.1.3.1.0), it is mapped directly into an array instead.

If you regenerate the Web service client, you will have to rewrite the client code. This is because the regenerated code will now be employing an `array[]` instead of a `BeanWrappingArray`.

See Also: "Developing Database Web Services" in the *Oracle Application Server Web Services Developer's Guide*

C.4.4 Assembling Web Services from an Existing WSDL File

If you have existing clients for your 10*g* Release 2 (10.1.2) Web service, and you want to keep the same contract (or WSDL file), you can generate the 10*g* Release 3 (10.1.3.1.0) Web service by assembling the Web service from the WSDL file, and then reuse the implementation code for your Web service. This is also known as top down Web service generation.

For detailed information about top down Web service assembly, see "Assembling a Web Service from WSDL" in the *Oracle Application Server Web Services Developer's Guide*.

C.5 Java Messaging Service (JMS) Considerations

The following sections provide information about using JMS 10*g* Release 3 (10.1.3.1.0):

C.5.1 Nomenclature Changes for 10g Release 3 (10.1.3) JMS Support

In past releases, Oracle used the terms "OracleAS JMS" and "OJMS" when describing the In-Memory, File-Based, and Database persistence options. "OracleAS JMS" referred to the In-Memory and File-Based options; "OJMS" referred to JMS interface to Streams Advanced Queuing (AQ).

For this release, the "OracleAS JMS" and "OJMS" nomenclature is not used. The "Oracle Enterprise Messaging Service (OEMS) JMS" reference is used instead. This change reflects the fact that Oracle offers a single Java Messaging Service (JMS) interface to the three message persistence options. As a result, you do not have to change your JMS application code if you decide to change message persistence between any of the three quality of service choices.

C.5.2 Using the JMS Connector Provided by 10*g* Release 3 (10.1.3.1.0)

The following sections describe the JMS Connector resource adapter available for Oracle Application Server 10*g* Release 3 (10.1.3.1.0):

C.5.2.1 About the 10*g* Release 3 (10.1.3.1.0) JMS Connector

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) provides a J2CA 1.5-compliant resource adapter called the JMS Connector that allows OC4J-managed applications to have a unified mechanism to access any JMS provider that implements JMS 1.1 or 1.02b.

Out-of-the-box, this release provides `OracleASjms`, which is an instance of the JMS Connector that is pre-configured for use with the OEMS JMS In-Memory and File-Based options.

Oracle recommends that new JMS applications be deployed using the JMS Connector. The JMS Connector provides the new features introduced in 10*g* Release 3 (10.1.3.0.0) and that are still available in 10*g* Release 3 (10.1.3.1.0). Oracle will continue to support JMS applications deployed using the older proprietary OC4J Resource Provider supported in Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2), but you are strongly encouraged to use the JMS Connector.

> **See Also:** "JMS Connector" in the "Java Message Service (JMS)" chapter of the *Oracle Containers for J2EE Services Guide*

C.5.2.2 Considerations for Global (XA) Transactions

Before you redeploy 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) J2EE applications that use JMS in global transactions, you must modify the corresponding deployment descriptors to

use OEMS JMS In-Memory and File-Based options via the JMS Connector. For 10*g* Release 3 (10.1.3.1.0), OEMS JMS In-Memory and File-Based options cannot be used for global transactions without the JMS Connector.

If you do not modify your deployment descriptors to use OEMS JMS In-Memory and File-Based options, Oracle provides a JMS property you can use as a temporary solution until you are able to update your deployment descriptors.

If the `oc4j.jms.pseudoTransactionEnlistment` property is set to TRUE, then all JMS Sessions (XA and non-XA) will be enlisted into global transactions, which is in violation of J2EE 1.4 and associated specs. This allows pre-existing applications that are using unsupported, non-compliant JMS semantics to continue to run, even after the implementation of fully JMS-compliant behavior in 10*g* Release 3 (10.1.3.1.0).

Note that enabling this flag will have uncertain results in other JMS applications that expect compliant behavior (for example, Oracle BPEL Process Manager, Oracle Enterprise Service Bus, and other SOA applications).

You can modify JMS properties by using the using the `JMSAdministrator` MBean, which is available from the System MBean Browser in the Application Server Control Console.

> **See Also:** "About the MBean Browser" in the Application Server Control online help
>
> "JMS Configuration Properties" in the *Oracle Containers for J2EE Services Guide*

C.5.3 Using the Application Server Control Console to Configure OEMS JMS

Unlike the Application Server Control Console in 10*g* (9.0.4) and 10*g* Release 2 (10.1.2), you can use the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console to manage the OC4J-provided OEMS In-Memory and File-Based resource provider. For example, you can use Application Server Control to create connection factories and destinations, as well as modify specific OEMS configuration properties.

Note that in the Application Server Control Console, the OEMS JMS In-Memory and File-Based resource provider is still referred to as the OracleAS JMS provider.

> **See Also:** "Managing the OracleAS JMS Provider" in the Application Server Control online help

C.5.4 Changes to the jms.xml Configuration File

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced additional elements to the `jms.xml` configuration file, as well as changes to the format of the `jms.xml` file so it is compliant with the latest schema. These changes also apply to 10*g* Release 3 (10.1.3.1.0).

If you redeploy a JMS application on 10*g* Release 3 (10.1.3.1.0), Oracle Application Server automatically rewrites the `jms.xml` file to use the new configuration file format and to add additional queues if they do not exist already.

Specifically, Oracle Application Server adds additional queues that are required by the scheduler and router, and one queue that is added for demonstration purposes. The new queues defined in the updated `jms.xml` file include:

- `jms/RAExceptionQueue`
- `jms/events`
- `jms/jobstore`

- `jms/notifications`

> **See Also:** "Configuration Elements" in the *Oracle Containers for J2EE Services Guide*

C.5.5 List of JAR Files Required for OEMS JMS Lookup

When you redeploy JMS applications on Oracle Application Server 10*g* Release 3 (10.1.3.1.0), note the following.

When using OEMS JMS In-Memory and File-Based options directly from an application client, the JAR files that must be included in the class path are listed in the section, "Required Class Path for Application Clients Using Direct OEMS JMS In-Memory and File-Based Lookup" in the *Oracle Containers for J2EE Services Guide*.

When using OEMS JMS Database option directly from an application client, the JAR files that must be included in the class path are listed in the section, "Required Class Path for Application Clients Using Direct OEMS JMS Database Lookup," in the *Oracle Containers for J2EE Services Guide*.

C.5.6 Database Version Support for OEMS JMS Database

Refer to the "OEMS JMS Database Certification Matrix" in the*Oracle Containers for J2EE Services Guide* for information on which versions of the Oracle database work with the Oracle Application Server when the OJMS client is running in OC4J.

C.5.7 Additional Data Source Requirement for OEMS JMS Database Applications

If you are deploying an OEMS JMS Database application on Oracle Application Server 10g Release (10.1.3), note that you must verify that the manage-local-transactions attribute in the data-sources.xml file is set to false.

The following example shows the managed-data-source element in the data-sources.xml file with the required attribute for OEMS JMS Database applications:

```
<managed-data-source name="OracleDS" connection-pool-name="Example
Connection Pool" jndi-name="jdbc/OracleDS" *manage-local-transactions="false"*/>.
```

C.6 Java Transaction API (JTA) Considerations

The Java Transaction API (JTA) is a specification developed by Sun Microsystems to provide support for global (distributed) transactions in the J2EE environment. Global transactions combine multiple enterprise systems - such as databases and message queues - into a single unit of work. The JTA maps the specifications based on the Open Group Distributed Transaction Processing model into the Java environment.

> **See Also:** "OC4J Transaction Support" in the *Oracle Containers for J2EE Services Guide*

The following sections highlight key changes to the OC4J JTA Support for 10*g* Release 3 (10.1.3.0.0). These changes also apply to 10*g* Release 3 (10.1.3.1.0). You should review these sections before deploying your existing 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) J2EE applications on 10*g* Release 3 (10.1.3.1.0):

- Using the New Middle-Tier Two-Phase Commit (2PC) Coordinator Instead of the Database Transaction Coordinator
- New Support for Transaction Propagation

C.6.1 Using the New Middle-Tier Two-Phase Commit (2PC) Coordinator Instead of the Database Transaction Coordinator

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced the Middle-Tier Two-Phase Commit (2PC) Coordinator that supports all XA-compatible resources, not just those from Oracle. This feature is referred to as a "heterogeneous middle tier coordinator," and it is also available in 10*g* Release 3 (10.1.3.1.0).

As a result, you are encouraged to use this new 2PC coordinator, instead of the deprecated in-database two-phase commit coordinator.

> **See Also:** "Middle-Tier Two-Phase Commit (2PC) Coordinator" in the *Oracle Containers for J2EE Services Guide*

C.6.2 New Support for Transaction Propagation

OC4J 10*g* Release 3 (10.1.3.0.0) introduced JTA transaction propagation. Transaction context propagation makes it possible for multiple OC4J instances to participate in a single global transaction. This feature is also available in 10*g* Release 3 (10.1.3.1.0).

Previous versions of Oracle Application Server did not support transaction propagation. As a result, when an OC4J instance that supports transaction propagation makes a remote method invocation on a bean that is deployed on an older version of OC4J that does not support transaction propagation, no transaction context is propagated.

> **See Also:** "Transaction Propagation Between OC4J Processes Over ORMI" in the *Oracle Containers for J2EE Services Guide*

C.7 Remote Method Invocation (RMI) Considerations

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced several new features and changes to the OC4J Remote Method Invocation (RMI) implementation. These changes are also supported by 10*g* Release 3 (10.1.3.1.0). For more information, see the following sections:

- Applying Compatibility Patches for 10g (9.0.4) and 10g Release 2 (10.1.2)
- New System Property for Configuring ORMI Request Load Balancing
- New Implementation of ORMI Tunnelling through HTTP
- Configuring Secure Connections with RMIS and SSL

C.7.1 Applying Compatibility Patches for 10*g* (9.0.4) and 10*g* Release 2 (10.1.2)

To use ORMI to invoke a method on a remote object when the invoking object and the invoked object are running on different OC4J versions, you must install a patch on the older version. This applies when the newer version is 10*g* Release 3 (10.1.3.1.0) and the older version is 10*g* (9.0.4) or 10*g* Release 2 (10.1.2).

For more information, see "Compatibility Patches for 9.0.4.x and 10.1.2.x" in the *Oracle Containers for J2EE Services Guide*.

C.7.2 New System Property for Configuring ORMI Request Load Balancing

In Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2), when two or more clients in the same process retrieved an `InitialContext`, you could use the `dedicated.connection` or `dedicated.rmicontext` properties to be sure that

each client received its own `InitialContext` instead of a shared context. When each client had its own `InitialContext`, then the clients could be load balanced.

These properties are deprecated in 10*g* Release 3 (10.1.3.1.0). Instead, you should use the new `oracle.j2ee.rmi.loadBalance` system property to specify load balancing in an application cluster. This property can be set in the client's `jndi.properties` file or in a Hashtable in the client code. The values for this property are:

- `client` — The client interacts with the OC4J process that was initially chosen at the first lookup (this is the default setting).
- `context` — The client goes to a new server when a separate context is used (this is similar to the deprecated `dedicated.rmicontext` property).
- `lookup` — The client goes to a new (randomly selected) server for every request.

C.7.3 New Implementation of ORMI Tunnelling through HTTP

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced a new implementation for ORMI tunneling through HTTP, which also applies to 10*g* Release 3 (10.1.3.1.0). For complete information, see "Configuring ORMI Tunneling through HTTP" in the *Oracle Containers for J2EE Services Guide*.

C.7.4 Configuring Secure Connections with RMIS and SSL

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) supports the use of Secure Socket Layer (SSL) for RMI connections. Complete instructions for configuring RMIS for your OC4J instances is included in the *Oracle Containers for J2EE Security Guide*.

Besides securing the RMI connections for your deployed applications, you can also secure the RMI management connections between the Administration OC4J instance (which is used to deploy the Application Server Control Console) and the other OC4J instances you are managing. For more information, see "Configuring Security for the Application Server Control Console" in the *Oracle Application Server Administrator's Guide*.

C.8 Java Naming and Directory Interface (JNDI) Considerations

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) introduced several new features and changes to JNDI. These changes also apply to 10*g* Release 3 (10.1.3.1.0). For a complete list of the new and changed JNDI features, see "Oracle JNDI" in the *Oracle Containers for J2EE Services Guide*.

In particular, before you deploy your J2EE applications on 10*g* Release 3 (10.1.3.1.0), review the following sections:

- New Package Names for Initial JNDI Context Factories
- JNDI-Related MBeans Now Available in the Application Server Control Console
- Performing Inter-Application JNDI Lookups
- Browsing the JNDI Context in the Application Server Control Console

C.8.1 New Package Names for Initial JNDI Context Factories

Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2) package names for OC4J initial context factories are deprecated. They will no longer be supported in future releases. Specifically, the following context factories are deprecated:

```
com.evermind.server.rmi.RMIInitialContextFactory
com.evermind.server.ApplicationClientInitialContextFactory
com.oracle.iiop.server.IIOPInitialContextFactory
```

Instead, you should use the following settings when using the `java.naming.factory.initial` property:

```
oracle.j2ee.rmi.RMIInitialContextFactory
oracle.j2ee.naming.ApplicationClientInitialContextFactory
oracle.j2ee.iiop.IIOPInitialContextFactory
```

> **See Also:** "Initial Context" in the *Oracle Containers for J2EE Services Guide*

C.8.2 JNDI-Related MBeans Now Available in the Application Server Control Console

The following JNDI-related MBeans are now registered with OC4J and are available for use within the MBean browser in the Application Server Control Console:

- JNDIResource
- JNDINamespace

> **See Also:** "About the MBean Browser" in the Application Server Control online help

C.8.3 Performing Inter-Application JNDI Lookups

It is now possible to configure JNDI to perform inter-application lookups. This in contrast to the default behavior, where lookups within an application are bound to be available within the current application's namespace.

Note that for global lookup to work properly, the target application's classes must be in the classpath of the application attempting the lookup.

> **See Also:** "Configuring JNDI for Deployment" in the *Oracle Containers for J2EE Services Guide*

C.8.4 Browsing the JNDI Context in the Application Server Control Console

You can browse the JNDI context for a selected application with the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console.

To browse the JNDI context, select the **JNDI Browser** task on the OC4J Administration page in the Application Server Control Console.

> **See Also:** "Browsing the JNDI Namespace for an OC4J Instance" in the Application Server Control online help

C.9 Security Considerations

Review the following sections for information on providing security for the J2EE applications you deploy on 10*g* Release 3 (10.1.3.1.0):

- List of Significant Changes in OC4J Security for 10g Release 3 (10.1.3.0.0) and 10g Release 3 (10.1.3.1.0)
- Converting principals.xml to the New JAAS Security Model
- Using Oracle Internet Directory as a Security Provider
- New DBTableOraDataSourceLoginModule OC4J Login Module

- New Default OC4J Administration Users and Roles

C.9.1 List of Significant Changes in OC4J Security for 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0)

The security features of Oracle Application Server and Oracle Containers for J2EE have been updated significantly since Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2). Additional changes were implemented for 10*g* Release 3 (10.1.3.1.0).

Before you redeploy your applications on 10*g* Release 3 (10.1.3.1.0), be sure to review the changes to the OC4J security features for security changes that could affect your application deployments.

Refer to the *Oracle Containers for J2EE Security Guide* for complete information about the security changes introduced in Oracle Application Server 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0).

The following sections provide an overview of the significant changes introduced for the 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0):

C.9.1.1 Changes in Release 10.1.3.1

This section notes changes and updated deprecation notices for the OC4J 10.1.3.1 implementation. Also review the section that follows, Section C.9.1.1, "Changes in Release 10.1.3.1".

C.9.1.1.1 Noteworthy Changes and Additions Note the following key additions in the OC4J 10.1.3.1 implementation:

- Identity management framework to support heterogeneous third-party identity management systems
- Java SSO, an alternative single sign-on solution packaged with OC4J that does not require additional infrastructure (such as Oracle Single Sign-On and Oracle Access Manager single sign-on do) and decouples OC4J from any identity management system that you use
- `DBTableOraDataSourceLoginModule` to replace `DataSourceUserManager` functionality. For more information, see Section C.9.4, "New DBTableOraDataSourceLoginModule OC4J Login Module".
- A new user and role API framework, particularly for use with supported LDAP servers. This includes replacement functionality for the deprecated `UserManager`, `User`, and `Group` classes

In addition, note the following changes:

- The JDK-default JSSE implementation is now the default SSL implementation for `HTTPClient`. (This is a step toward deprecating OracleSSL, the previous default SSL implementation, for use with `HTTPClient`.)

C.9.1.1.2 Updated Deprecation Notices Note the following deprecations in the OC4J 10.1.3.1 implementation:

- The `com.evermind.security` package and its classes are deprecated. **They will no longer be supported in the 11*g* release.**
  - `UserManager` class: Use JAAS custom login modules instead of custom `com.evermind.security.UserManager` implementations.
  - `User` class: Use standard JAAS APIs instead.

  - `Group` class: Use standard JAAS APIs instead.

- The `XMLUserManager` class and its data store, `principals.xml`, are deprecated. **They will no longer be supported in the 11*g* release.** For more information, see Section C.9.2, "Converting principals.xml to the New JAAS Security Model".

C.9.1.2 Changes in Release 10.1.3.0.0

The following security features and enhancements were added for the OC4J 10.1.3.0.0 implementation:

- Support for the COREid Access (now Oracle Access Manager) security provider
- Support for the LDAP-based provider in standalone OC4J
- Digest authentication support, and client certification authentication and authorization support
- Implementation of the Java Authorization Contract for Containers (JSR-115)
- JAAS integration with EJBs
- ORMI enhancements for SSL (ORMIS)
- Support for subject propagation (with ORMI or ORMIS)
- JMX and MBeans support (JSR-77) for security configuration
- New OC4J user and role accounts
- Enhanced Java 2 security support
- Web services security

In addition, note the following changes since the OC4J 10.1.2 implementation:

- There is a new consolidated "JAAS mode" for authorization, for both servlets and EJBs. This replaces previous `runas-mode` and `dosasprivileged-mode` functionality for servlets, and `USE_JAAS` functionality (introduced in preliminary 10.1.3 releases) for EJBs. The previous functionality is supported but deprecated in OC4J 10.1.3.x implementations.
- The instance-level `jazn-data.xml` configuration file used in previous releases to store user and role configuration (for the file-based provider), policy configuration (for the file-based, external LDAP, or custom security provider), and login module configuration (for all security providers) has been renamed `system-jazn-data.xml`. However, an application can optionally use an application-specific `jazn-data.xml` repository file to store user and role configuration for the file-based provider.
- The `XMLUserManager` class and its data store, `principals.xml`, are deprecated and will no longer be supported at a future release. We strongly encourage you to migrate your existing applications. For more information, see Section C.9.2, "Converting principals.xml to the New JAAS Security Model".
- Custom `UserManager` classes are still supported at this release, but will be deprecated at a future release. We recommend that you use JAAS custom login modules instead of custom `com.evermind.security.UserManager` implementations.
- For the Oracle Identity Management security provider, the application realm and external realm are deprecated.
- The `external.synchronization` property is no longer supported.

- The default setting of the `jaas.username.simple` property is now "`true`"; in the 10.1.2 implementation, the default setting was "`false`". This now means that by default, realm names are omitted from the names of authenticated principals returned by such methods as `getUserPrincipal()` and `getRemoteUser()` for servlets, and `getCallerPrincipal()` for EJBs.
- There have been some OC4J account name changes: the `admin` account is now `oc4jadmin`; the `administrators` role is now `oc4j-administrators`; the `jmx-users` role is now `oc4j-app-administrators`. For the file-based provider in standalone OC4J, `oc4jadmin` is initially deactivated. See Section C.9.5, "New Default OC4J Administration Users and Roles".
- Required OC4J accounts are created automatically in Oracle Internet Directory when you associate an OC4J instance with an OID instance. See "Required Accounts Created in Oracle Internet Directory" in the *Oracle Containers for J2EE Security Guide*.
- Setting `LD_LIBRARY_PATH` is no longer necessary in 10.1.3.x implementations.
- The `jazn.debug.log.enable` flag is no longer supported for logging. Use regular OC4J logging features. See "Logging" in the *Oracle Containers for J2EE Security Guide*.

C.9.2 Converting principals.xml to the New JAAS Security Model

For Oracle Application Server 10*g* Release 3 (10.1.3.1.0), the `XMLUserManager` class and its datastore, `principals.xml`, are supported, but you are strongly encouraged to migrate to the new JAAS security model.

If an application that you want to redeploy on 10*g* Release 3 (10.1.3.1.0) was previously using the `XMLUserManager` class, you can use the JAZN Admintool to migrate the data in the principals defined in the `principals.xml` file to the new JAAS security model.

For more information, see "Migrating Principals from the principals.xml File" in the *Oracle Containers for J2EE Security Guide*.

C.9.3 Using Oracle Internet Directory as a Security Provider

Oracle Application Server 10*g* Release 3 (10.1.3.1.0) supports the use of Oracle Internet Directory as a security provider and Oracle Single Sign-On for the applications you deploy.

Before you deploy an application that requires Oracle Internet Directory or Oracle Single Sign-On, see "Oracle Identity Management" in the *Oracle Containers for J2EE Security Guide* for complete instructions.

> **See Also:** Section 3.3.2, "Using Existing OracleAS Infrastructure Components"

Note that when you associate an OC4J instance with an Oracle Internet Directory instance, a set of required OC4J accounts are created automatically in Oracle Internet Directory. For more information, see "Required Accounts Created in Oracle Internet Directory" in the *Oracle Containers for J2EE Security Guide*.

C.9.4 New DBTableOraDataSourceLoginModule OC4J Login Module

For 10*g* Release 3 (10.1.3.1.0), the OC4J implementation supplies a login module you can use if you have a user identity store in a database:

```
oracle.security.jazn.login.module.db.DBTableOraDataSourceLoginModule
```

This replaces previous functionality of the following class, now deprecated (but still supported for backward compatibility):

```
com.evermind.sql.DataSourceUserManager
```

The new login module also replaces some authentication functionality of the deprecated `com.evermind.security.User` class.

Once you have created your database schema and an Oracle datasource to connect to the database, you are ready to configure the login module.

`DBTableOraDataSourceLoginModule` supports a number of options for specifying such items as data location (table and column names) and password encryption. You can set these options through Application Server Control or the OracleAS JAAS Provider Admintool, with the settings being reflected in the `<jazn-loginconfig>` element of the `system-jazn-data.xml` file.

> **See Also:** "DBTableOraDataSourceLoginModule" in the *Oracle Containers for J2EE Security Guide*

C.9.5 New Default OC4J Administration Users and Roles

Refer to the following sections for information about the new OC4J default administrator accounts available with 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0):

- Summary of the New Administrate Users and Roles
- Using Security Role Mapping to Access New Users and Roles
- Example: Using Security Role Mapping for the helloworld Sample Application

C.9.5.1 Summary of the New Administrate Users and Roles

Oracle Application Server 10*g* Release 3 (10.1.3.0.0) and 10*g* Release 3 (10.1.3.1.0) introduce changes to the following default OC4J users and roles that were previously defined by default in Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2):

- The `admin` account is now `oc4jadmin`
- The `administrators` role is now `oc4j-administrators`
- The `jmx-users` role is now `oc4j-app-administrators`

C.9.5.2 Using Security Role Mapping to Access New Users and Roles

If any of your 10*g* (9.0.4) or 10*g* Release 2 (10.1.2) applications depend upon the `admin` account or the administrators role, you can map your application-specific security roles to the new OC4J default security roles.

You can perform this task from the Application Server Control Console during the deployment of your application, or you can modify the appropriate `orion-application.xml`, `orion-ejb-jar.xml`, or `orion-web.xml` configuration file after deployment.

C.9.5.3 Example: Using Security Role Mapping for the helloworld Sample Application

The `helloworld` EJB sample application, available on the Oracle Technology Network, is an example of an application that depends upon the `admin` user account.

The application was designed to work with the default administration users and roles defined for OC4J 10*g* (9.0.4) and 10*g* Release 2 (10.1.2).

As a result, if you deploy the `helloworld` application on 10*g* Release 3 (10.1.3.1.0) without mapping the security roles, you will be unable to log in to the sample application. To remedy this problem, map the `users` role of the `helloworld` application to the `oc4j-administrators` role available in 10*g* Release 3 (10.1.3.1.0).

> **Note:** The following procedure assumes you have a copy of the `helloworld.ear` application archive available on your local hard disk. The sample application is available from the Enterprise JavaBeans Samples page on the Oracle Technology Network.

If you perform the following steps, then you can log in to the `helloworld` application using the 10*g* Release 3 (10.1.3.1.0) `oc4jadmin` account.

Task 1 Use Application Server Control to Map the Security Roles and Deploy the Application

1. Log in to the 10*g* Release 3 (10.1.3.1.0) Application Server Control Console and navigate to the OC4J home page.
2. Click **Applications** to display the OC4J Applications page.
3. Click **Deploy** to start the Application Server Control deployment wizard.
4. Follow the instructions on the screen and proceed through the wizard until Application Server Control displays the Deployment Settings page.
5. Use the **Map Security Roles** task to map the helloworld security roles to the security roles defined for the 10*g* Release 3 (10.1.3.1.0) OC4J instance:
   a. On the Deployment Tasks page, click the Go to Task icon for the **Map Security Roles** deployment task.

      On the Map Security Roles page, notice that a security role called `users` is defined for the `helloworld` application. The `users` role is configured for the `hello` EJB module, as well as for the `helloworld` web application.

   b. Click the **Map Role** icon for the `hello` EJB module.

      Application Server Control displays the Map Security Role page (Figure C–1).

   c. Select **Map selected users and groups to this role**.

   d. In the **Map Roles to Groups** section of the page, enter `oc4j-administrators` in the **Group** field and click **Add**.

      The `oc4j-administrators` role appears in the table above the field. This indicates that Application Server Control will map the `users` role, which is defined for the application, to the `oc4jadmin-administrators` role, which is defined in the `system-jazn-data.xml` of the 10*g* Release 3 (10.1.3.1.0) OC4J instance.

Figure C–1 Deployment Settings: Map Security Roles Page

e. Click **Continue** to return to the Map Security Role page.

f. Repeat steps b through e for the `helloworld` web application.

g. Click **OK** to return to the Deployment Tasks page.

6. Click **Deploy** to deploy the `helloworld` application.

   Application Server Control shows the status of the deployment and displays a success message when the deployment is complete.

7. Click **Return** to return to the OC4J Applications page.

Task 2 Verify that Security Roles are Mapped Correctly

1. Click the name of the `helloworld` application in the list of deployed applications.

   Application Server Control displays the Application Home page for the `helloworld` application.

2. Click **helloworld-web** to display the helloworld web application page.

3. Test the Web module to verify that you can log into the `helloworld` application using the oc4jadmin user credentials:

   a. Click the **Test Web Module** icon to display the Test Web module page.

   b. Select a valid listener from the table and click **Test Web Module**.

   c. On the first test page, click the word **here** to display the login prompt for the helloworld application.

   d. Enter `oc4jadmin` and the password you defined for the account during the Oracle Application Server installation.

      If the security roles are mapped correctly, the helloworld application displays the following, which confirms that the application is working successfully:

      ```
      Hello James Earl
      ```

Task 3 Verify the Changes Made to the orion-web.xml Configuration File

1. Return to the Web Module page for the helloworld-web Web module.
2. Click **Administration** to display the list of tasks you can perform for a selected Web module.
3. Click the task icon for the View Proprietary Deployment Descriptor task.
4. Scroll through the `orion-web.xml` configuration file and locate the following entry:

```
<security-role-mapping name="users">
    <group name="oc4j-administrators" />
</security-role-mapping>
```

This is the entry that is added when you map the users security role to the `oc4j-administrators` group defined by the `system-jazn-data.xml` file for the 10*g* Release 3 (10.1.3.1.0) OC4J instance.

> **See Also:** "Overview of Security Role Mapping" in the *Oracle Containers for J2EE Security Guide*

C.10 Oracle TopLink and EJB Considerations

Use the following sections to take advantage of Oracle TopLink for your 10*g* Release 3 (10.1.3.1.0) applications:

- Configuring CMP Entity Beans to Use Oracle TopLink Persistence Manager
- Upgrading TopLink Workbench Projects

C.10.1 Configuring CMP Entity Beans to Use Oracle TopLink Persistence Manager

In Oracle Application Server 10*g* (9.0.4) and 10*g* Release 2 (10.1.2), the default persistence manager was Orion CMP. In 10*g* Release 3 (10.1.3.0.0) and 10g Release 3 (10.1.3), OC4J is configured by default to use Oracle TopLink as its default persistence manager.

> **Note:** OrionCMP was deprecated in 10*g* Release 3 (10.1.3.0.0) 10*g* Release 3 (10.1.3.1.0), and it will be desupported in future versions of Oracle Application Server.

As a result, before you redeploy your EJB applications that use Orion CMP, you must migrate persistence configuration from your original `orion-ejb-jar.xml` file to the `toplink-ejb-jar.xml` file.

Oracle provides a TopLink migration tool that you can use to automate this migration.

> **See Also:** "Migrating OC4J Orion Persistence to OC4J TopLink Persistence" in the *Oracle TopLink Developer's Guide*

C.10.2 Upgrading TopLink Workbench Projects

If you have used Oracle TopLink with previous versions of Oracle Application Server, you can migrate your existing Oracle TopLink projects to TopLink 10g Release 3 (10.1.3).

For more information, see "Migrating to 10g Release 3 (10.1.3)" in the *Oracle TopLink Getting Started Guide*.

Index

Numerics

A

B

C

D

E

F

G

H

I

J

L

M

O

P

R

S

T

U

V

W

X