# Oracle® Application Server

Disaster Recovery Guide Using OracleAS Guard

10*g* Release 2 (10.1.2.3)

**E11078-02**

April 2008

Oracle Application Server Disaster Recovery Guide Using OracleAS Guard, 10*g* Release 2 (10.1.2.3)

E11078-02

# Contents

## 2 Oracle Application Server Guard and asgctl

## 3 Configuring OracleAS Disaster Recovery

## 4 Disaster Recovery Special Considerations

## 5 Oracle Application Server Guard asgctl Command-line Reference

## B Oracle Application Server Guard Error Messages

## C Sync Operations Automated by OracleAS Disaster Recovery

## Glossary

## Index

# Preface

This preface contains these sections:

- Intended Audience
- Documentation Accessibility
- Related Documentation
- Conventions

## Intended Audience

The *Oracle Application Server Disaster Recovery Guide Using OracleAS Guard* is intended for administrators, developers, and others whose role is to deploy and manage Oracle Application Server in Disaster Recovery environments using the Oracle Application Server Guard (OracleAS Guard) utility.

The information in this manual replaces the Disaster Recovery chapters and appendixes in the 10.1.x releases of the *Oracle Application Server High Availability Guide*, which included information about the OracleAS Guard utility. This manual assumes that you have read and are familiar with the High Availability information in the *Oracle Application Server High Availability Guide*.

> **Note:** This manual describes how to use OracleAS Guard for disaster protection of your Oracle Application Server deployment. If you are already using OracleAS Guard for disaster protection of your present Oracle Application Server deployment, then the information in this manual can help you manage it.
>
> However, if you are planning a new Oracle Application Server deployment that needs disaster protection, Oracle recommends that you use disk replication to provide disaster protection for the Oracle Application Server deployment. For more information on how to provide disaster protection for Oracle Application Server deployments using disk replication, refer to the *Oracle Application Server Disaster Recovery Guide* for Oracle Application Server release 10.1.3.3.0.

## 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, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

## Related Documentation

For more information, see these Oracle resources:

- *Oracle Application Server High Availability Guide*
- *Oracle Application Server Concepts*
- *Oracle Application Server Installation Guide*
- *Oracle Application Server Administrator'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

# OracleAS Disaster Recovery Introduction

Disaster recovery refers to how a system recovers from catastrophic site failures caused by natural or unnatural disasters. Examples of catastrophic failures include earthquakes, tornadoes, floods, or fire. Additionally, disaster recovery can also refer to how a system is managed for planned outages. For most disaster recovery situations, the solution involves replicating an entire site, not just pieces of hardware or subcomponents. This also applies to the Oracle Application Server Disaster Recovery (OracleAS Disaster Recovery) solution.

The *Oracle Application Server Disaster Recovery Guide Using OracleAS Guard* is intended for administrators, developers, and others whose role is to deploy and manage Oracle Application Server in Disaster Recovery environments using the Oracle Application Server Guard (OracleAS Guard) utility.

The information in this manual replaces the Disaster Recovery chapters and appendixes in the 10.1.x releases of the *Oracle Application Server High Availability Guide*, which included information about the OracleAS Guard utility. This manual assumes that you have read and are familiar with the High Availability information in the *Oracle Application Server High Availability Guide*.

> **Note:** This manual describes how to use OracleAS Guard for disaster protection of your Oracle Application Server deployment. If you are already using OracleAS Guard for disaster protection of your present Oracle Application Server deployment, then the information in this manual can help you manage it.
>
> However, if you are planning a new Oracle Application Server deployment that needs disaster protection, Oracle recommends that you use disk replication to provide disaster protection for the Oracle Application Server deployment. For more information on how to provide disaster protection for Oracle Application Server deployments using disk replication, refer to the *Oracle Application Server Disaster Recovery Guide* for Oracle Application Server release 10.1.3.3.0.

This chapter describes the OracleAS Disaster Recovery solution, how to configure and set up its environment, and how to manage the solution for high availability. The discussion involves both OracleAS middle tiers and OracleAS Infrastructure tiers in two sites: production and standby. The standby site is configured either identically and symmetrically (same number of instances) or asymmetrically to the production site (fewer instances or OracleAS Disaster Recovery capability for only the Infrastructure services is supported). Under normal operation, the production site actively services requests. The standby site is maintained to mirror or closely mirror the applications and content hosted by the production site.

The term topology refers to all Oracle Application Server instances that share the same Oracle Internet Directory for an OracleAS Disaster Recovery production site. The discover topology command queries Oracle Internet Directory to determine the list of instances and then generates a `topology.xml` file that describes the production site topology. The discover topology within farm command is used in cases where Oracle Internet Directory is not available (for example, in Oracle Application Server 10.1.3 environments); in this case, Oracle Application Server Guard uses OPMN to discover the topology within the farm.

The OracleAS Disaster Recovery aspects of these sites are managed using Oracle Application Server Guard, which contains a command-line utility (asgctl) that encapsulates administrative tasks (see Chapter 5, "Oracle Application Server Guard asgctl Command-line Reference" for reference information about these administrative commands). The OracleAS Disaster Recovery solution leverages the following services among other system services that are available across the entire site. Behind the scenes Oracle Application Server Guard automates the use of OracleAS Recovery Manager (for managing configuration files in the file system) and Oracle Data Guard (for managing the OracleAS Infrastructure database) in a distributed fashion across the topology. Table 1–1 provides a summary of the OracleAS Disaster Recovery strategy and how this Oracle software is used behind the scenes:

***Table 1–1 Overview of OracleAS Disaster Recovery Strategy***

| Coverage | Procedure | Purpose |
|---|---|---|
| Middle-tier Configuration Files | OracleAS Recovery Manager | To back up and clone configuration files in the production site middle-tier nodes and restore the files to the standby site middle-tier nodes. |
| OracleAS Infrastructure Configuration Files | OracleAS Recovery Manager | To back up and clone OracleAS configuration files in the production site OracleAS Infrastructure node and restore them to the standby site OracleAS Infrastructure node. |
| OracleAS Infrastructure Database | Oracle Data Guard | To ship archive logs from production site OracleAS Infrastructure database to standby site OracleAS Infrastructure database. Logs are not applied immediately. |

You can use the add instance command to manually add any database instances besides the OracleAS Infrastructure database to the topology as part of the disaster recovery solution.

> **Note:** You must configure Oracle Data Guard for every database in your OracleAS Disaster Recovery topology. See Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology" for more information about configuring Oracle Data Guard for the databases in your OracleAS Disaster Recovery topology.

In addition to the recovery strategies, this chapter discusses configuration and installation of both sites. For these tasks, two different ways of naming the middle-tier nodes are covered as well as two ways of resolving hostnames intra-site and inter-site.

With OracleAS Disaster Recovery, planned outages of the production site can be performed without interruption of service by switching over to the standby site using the Oracle Application Server Guard switchover operation. Unplanned outages are managed by failing over to the standby site using the Oracle Application Server Guard failover operation. Procedures for switchover and failover are covered in Section 2.9, "Runtime Operations -- Oracle Application Server Guard Switchover and Failover Operations".

This chapter is organized into the following sections:

- Section 1.1, "Oracle Application Server 10g Disaster Recovery Solution"
- Section 1.2, "Preparing the OracleAS Disaster Recovery Environment"
- Section 1.4, "Overview of Installing Oracle Application Server"
- Section 1.5, "Wide Area DNS Operations"

> **See Also:** *Oracle Application Server Installation Guide* for instructions about how to install the OracleAS Disaster Recovery solution.

## 1.1 Oracle Application Server 10*g* Disaster Recovery Solution

The Oracle Application Server Disaster Recovery solution consists of two configured sites - one primary (production/active) and one secondary (standby). Both sites may or may not have the following: same number of middle tiers and the same number of OracleAS Infrastructure nodes, and the same number of components installed. In other words, the installations on both sites, middle tier and OracleAS Infrastructure could be identical (symmetrical topology) or not identical (asymmetrical topology). Both sites are usually dispersed geographically, and if so, they are connected through a wide area network.

Some important points to emphasize for the Oracle Application Server Disaster Recovery solution are the following:

- The number of instances required on the standby site to run your site can be identical to (symmetric) or fewer (asymmetric) than the production site.
- The set of instances needed must be created on the standby site for Disaster Recovery set up. You can create the instances at the standby site by installing them using Oracle Universal Installer or by cloning them from the production site using the asgctl `clone instance` command or the asgctl `clone topology` command. See Section 2.7.1, "Cloning Single or Multiple Production Instances to a Standby System" for more information on cloning.
- The standby site needs the minimum set of instances required to run your site.

This section describes the overall layout of the solution, the major components involved, and the configuration of these components. It has the following sections:

- Section 1.1.1, "OracleAS Disaster Recovery Requirements"
- Section 1.1.2, "Using Oracle Application Server Guard in an OracleAS Disaster Recovery Topology"
- Section 1.1.3, "Supported Topologies"

### 1.1.1 OracleAS Disaster Recovery Requirements

To ensure that your implementation of the OracleAS Disaster Recovery solution performs as designed, the following requirements must be adhered to:

- On each host in the standby site, make sure the following is identical to its equivalent peer in the production site:
  - For the middle-tier hosts, physical hostnames.

    **Note:** If you already have installed systems, you must only modify the physical names for the middle-tier systems at the standby site to match the physical and network hostname of the peer systems at the production site. Then create a virtual hostname for the physical hostname of the OracleAS Infrastructure at the standby site to match the virtual hostname of the OracleAS Infrastructure at the production site. See Section 1.2.1, "Planning and Assigning Hostnames" for information about physical hostnames, network hostnames, and virtual hostnames. See the Glossary for a definition of physical hostname, network hostname, and virtual hostname.

  - Port number usage. Make sure that no port number conflicts exist for production site and standby site peer hosts.
  - Hardware platform
  - Operating system release and patch levels
- All installations conform to the requirements listed in the *Oracle Application Server Installation Guide* to install Oracle Application Server.
- The following details must be the same between a host in the production site and a peer in the standby site:
  - User name and password of the user who installed Oracle Application Server must be the same between a host in the production site and its peer in the standby site.
  - Numerical user ID of the user who installed Oracle Application Server on that particular node
  - Group name of the user who installed Oracle Application Server on that particular node
  - Numerical group ID of the group for the user who installed Oracle Application Server on that particular node
  - Environment profile
  - Shell (command-line environment)
  - Directory structure, Oracle home names, and path of the Oracle home directory for each OracleAS installation on a node. Do not use symbolic links anywhere in the path.
  - Oracle Application Server installation types (Any instance installed on the standby system must be identical to that installed on the production system):
    - Middle Tier: J2EE and Web Cache, Portal and Wireless, and Business Intelligence and Forms
    - OracleAS Infrastructure: Metadata Repository (MR) and Identity Management (IM)
  - The SID names must be the same for Oracle database peers at a Disaster Recovery primary site and standby site(s).

- Entries in TNSNAMES.ORA files for databases at a production site or standby site should include the domain name.

#### 1.1.1.1 Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology

Any database in an Oracle Application Server Disaster Recovery topology must be configured by Oracle Data Guard, so that Oracle Data Guard will ship archive logs from the primary database at the production site to the standby database at the standby site.

> **Note:** Before you attempt to configure Oracle Data Guard for a database, use the asgctl `dump topology` command to ensure that the database is included in the Disaster Recovery topology at the primary site. If the database is not included in the Disaster Recovery topology at the primary site, use the asgctl `add instance` command first to add the database to the topology. Then use Table 1–2 to choose an appropriate method of configuring Oracle Data Guard for the primary database at the primary site and the standby database at the standby site.

Table 1–2 assumes that you have created the primary database at the primary site, and the database is included in the Disaster Recovery topology for the primary site.

Table 1–2 shows the supported methods of configuring Oracle Data Guard for different types of databases that can be included in your Disaster Recovery topology.

***Table 1–2 Configuring Data Guard for Databases in an OracleAS Disaster Recovery Topology***

| Database Type at Primary Site | Method(s) of Configuring Oracle Data Guard for the Database |
|---|---|
| A database with the Metadata Repository schemas that is created during an Application Server Infrastructure installation in the Application Server home[1] | You can use either of these methods:<br>▪ the `clone instance` command or `clone topology` command<br>▪ the `instantiate topology` command<br>For more information about configuring Oracle Data Guard using these commands, refer to the reference information for the clone instance, clone topology, and instantiate topology commands. |
| A single instance or RAC database installed outside an Application Server home that does *not* use the Oracle Managed Files (OMF) or Automatic Storage Management (ASM) database storage options | Use the `create standby database` command.<br>For more information about configuring Oracle Data Guard using the `create standby database` command, refer to the reference information for the create standby database command.<br>After using the `create standby database` command, perform a `sync topology` command. For more information, refer to the reference information for the sync topology command. |

*Table 1–2 (Cont.) Configuring Data Guard for Databases in an OracleAS Disaster Recovery Topology*

| Database Type at Primary Site | Method(s) of Configuring Oracle Data Guard for the Database |
|---|---|
| A single instance or RAC database installed outside an Application Server home that uses either the Oracle Managed Files (OMF) or Automatic Storage Management (ASM) database storage option | Create the standby database and configure Oracle Data Guard for the production and standby databases by following the instructions in the "Creating a Standby Database that Uses OMF or ASM" section of *Oracle Data Guard Concepts and Administration* in the Oracle Database documentation set.<br>Then, perform a `sync topology` command. For more information, refer to the reference information for the sync topology command. |

[1] The only type of database that Disaster Recovery supports in an Oracle Application Server home is a database that includes that Metadata Repository schemas and which was created during an Application Server Infrastructure installation in the Application Server home.

### 1.1.1.2 Understanding the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard

When you use the asgctl `clone instance`, `clone topology`, `instantiate topology`, or `create standby database` command to configure Oracle Data Guard for primary and standby databases in your Disaster Recovery topology, the following Oracle Data Guard configuration is set up by Oracle Application Server Guard:

- The primary database is set up with the maximum availability data protection mode.
- The standby database is set up with the LGRW SYNC and AFFIRM archive attributes for the LOG_ARCHIVE_DEST_*n* parameter.

In some cases, for example, if you are using BPEL Process Manager, you may want to change the default Oracle Data Guard configuration set up by Oracle Application Server Guard. For more information, see Section A.1.1, "Changing the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard."

## 1.1.2 Using Oracle Application Server Guard in an OracleAS Disaster Recovery Topology

Oracle Application Server Guard (ASG) supports Oracle Application Server release 10g (10.1.2.0.0, 10.1.2.0.2, 10.1.2.1, 10.1.2.2, 10.1.2.3, 10.1.3.0.0, 10.1.3.1.0, 10.1.3.2.0, and 10.1.3.3.0). By default, when you install an Oracle Application Server instance using any Oracle Application Server installation type that also installs a JDK or JRE environment, a particular release of ASG is installed into the Oracle home when the instance is installed.

You must also install ASG on standalone hosts on which external resources (such as an Oracle database) are located that you want to include in your OracleAS Disaster Recovery topology.

Multiple releases of ASG are available. It is possible (recommended) in some cases to upgrade the ASG release that was installed in an Oracle Application Server instance home when you installed that instance. To upgrade the ASG release in an Oracle Application Server instance home, download the ASG standalone kit for the recommended ASG release from the Oracle Technology Network (OTN), and then use that ASG standalone kit to install the recommended ASG release into the home.

You can access the Oracle Technology Network at:

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

Use the ASG standalone kit to install ASG on standalone hosts that you want to include in your OracleAS Disaster Recovery topology. The ASG standalone kit must be installed in its own home on any host that has a database Oracle home that you want to include in your Disaster Recovery topology (*except* on a host that has a database with the Metadata Repository schemas that is created during an Application Server Infrastructure installation in the Application Server home). After you install the ASG standalone kit on a database host, use the asgctl add instance command to add the database instance to your Disaster Recovery topology.

You must install the ASG kit on all systems with Oracle homes or Oracle Application Server instances that you want to include in your Disaster Recovery topology. See the OracleAS Disaster Recovery installation information in *Oracle Application Server Installation Guide* for more information.

Use Table 1–3 and Table 1–4 to determine whether a particular ASG release is compatible when installed into an Application Server instance home for a particular Oracle Application Server release. The left column of the table shows the different ASG releases for which an ASG standalone installation kit is available. The remaining columns show different Oracle Application Server releases for which an Oracle Application Server instance can be created.

This list describes the meaning of the entries in Table 1–3 and Table 1–4:

- **N**: This ASG release is not compatible with an instance from this Oracle Application Server release.
- **X**: This ASG release cannot be installed into the Oracle home for an instance from this Oracle Application Server release.
- **Y-NR**: This ASG release is compatible with an instance from this Oracle Application Server release, but Oracle recommends that you *do not* install this ASG release into the instance's Oracle home because another ASG release is recommended.
- **Y**: This ASG release is compatible with an instance from this Oracle Application Server release. Oracle recommends you install this ASG release into the instance's Oracle home.

Table 1–3 shows the compatible ASG releases for Oracle Application Server instances from Oracle Application Server 10.1.2.0.2 through 10.1.3.3.

***Table 1–3 Compatible ASG Releases for OracleAS Instances from Releases 10.1.2.0.2 Through 10.1.3.3***

| ASG Release | 10.1.2.0.2 OracleAS Instance | 10.1.2.1 OracleAS Instance | 10.1.2.2 OracleAS Instance | 10.1.2.3 OracleAS Instance | 10.1.3.0 OracleAS Instance | 10.1.3.1 OracleAS Instance | 10.1.3.2 OracleAS Instance | 10.1.3.3 OracleAS Instance |
|---|---|---|---|---|---|---|---|---|
| 10.1.2.0.2 | Y-NR | X | X | Y | N | N | N | N |
| 10.1.2.2 | Y | Y | Y | Y | N | N | N | N |
| 10.1.2.2.1 (ASG-only release)[1] | Y | Y | Y | Y | N | N | N | N |
| 10.1.2.3 | Y | Y | Y | Y | N | N | N | N |
| 10.1.3.0 | N | N | N | N | Y-NR | X | N | X |
| 10.1.3.1 | N | N | N | N | Y-NR | Y-NR | Y-NR | X |
| 10.1.3.3 | N | N | N | N | Y | Y | Y | Y |

[1] This is the ASG release that was provided (installed by default) with the OracleAS 10.1.4.2 release. It is compatible with the OracleAS 10.1.2.x releases. There is no OracleAS 10.1.2.2.1 release.

For example, if you have an Oracle Application Server 10.1.3.1 instance and you want to know which ASG release to install in the Oracle home for the instance, you can use Table 1–3 to determine the following:

- No ASG 10.1.2.x release is compatible with an Oracle Application Server 10.1.3.1 instance.
- The ASG 10.1.3.0 release cannot be installed into the Oracle home for an Oracle Application Server 10.1.3.1 instance.
- The ASG 10.1.3.1 release is compatible with an Oracle Application Server 10.1.3.1 instance, but Oracle recommends that you *do not* install the ASG 10.1.3.1 release into the Oracle home for an Oracle Application Server 10.1.3.1 instance.
- The ASG 10.1.3.3 release is compatible with an Oracle Application Server 10.1.3.1 instance and Oracle recommends that you install the ASG 10.1.3.3 release into the Oracle home for an Oracle Application Server 10.1.3.1 instance.

Table 1–4 shows the compatible ASG releases for Oracle Application Server instances from Oracle Application Server 10.1.4.0 through 10.1.4.2.

***Table 1–4 Compatible ASG Releases for OracleAS Instances from Releases 10.1.4.0 Through 10.1.4.2***

| ASG Release | 10.1.4.0 OracleAS Instance[1] | 10.1.4.1 OracleAS Instance[2] | 10.1.4.2 OracleAS Instance[3] |
|---|---|---|---|
| 10.1.2.0.2 | Y-NR | Y-NR | X |
| 10.1.2.2 | Y-NR | Y-NR | Y-NR |
| 10.1.2.2.1 (ASG-only release)[4] | Y | Y | Y |
| 10.1.2.3 | Y-NR | Y-NR | Y-NR |
| 10.1.3.0 | N | N | N |
| 10.1.3.1 | N | N | N |
| 10.1.3.3 | N | N | N |

[1] ASG 10.1.2.0.2 is installed by default.

[2] ASG 10.1.2.0.2 is installed by default.

[3] ASG 10.1.2.2.1 is installed by default.

[4] This is the ASG release that was provided (installed by default) with the OracleAS 10.1.4.2 release. It is compatible with the OracleAS 10.1.2.x releases. There is no OracleAS 10.1.2.2.1 release.

In addition to making sure that each of the ASG releases installed in the Oracle Application Server instance homes in your topology are compatible with those Oracle Application Server releases, you must make sure that all the ASG releases in the topology are compatible with each other.

Use Table 1–5 to determine whether two ASG releases are compatible in an OracleAS Disaster Recovery topology. Find the first ASG release in the left column of the table and then find the second ASG release in one of the other columns of the table.

This list describes the meaning of the entries in Table 1–5:

- **Y-NR**: The first ASG release is compatible with the second ASG release, but Oracle recommends that you *do not* use this ASG release combination in your topology.

- **Y**: The first ASG release is compatible with the second ASG release. Oracle recommends that you use this ASG release combination in your topology.

Table 1–5 shows which ASG releases are compatible with other ASG releases.

*Table 1–5 Compatible ASG Releases in a Topology*

| ASG Release | 10.1.2.0.2 | 10.1.2.2 | 10.1.2.2.1 | 10.1.2.3 | 10.1.3.0 | 10.1.3.1 | 10.1.3.3 |
|---|---|---|---|---|---|---|---|
| 10.1.2.0.2 | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR |
| 10.1.2.2 | Y-NR | Y | Y | Y-NR | Y-NR | Y-NR | Y |
| 10.1.2.2.1 | Y-NR | Y | Y | Y | Y-NR | Y-NR | Y |
| 10.1.2.3 | Y-NR | Y-NR | Y | Y | Y-NR | Y-NR | Y |
| 10.1.3.0 | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR |
| 10.1.3.1 | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR | Y-NR |
| 10.1.3.3 | Y-NR | Y | Y | Y | Y-NR | Y-NR | Y |

You must also make sure that each Oracle Application Server home on a standby site peer host is upgraded to the same ASG release as the equivalent Application Server home at the production site host.

## 1.1.3 Supported Topologies

OracleAS Disaster Recovery supports a number of basic topologies for the configuration of the Infrastructure and middle tier on production and standby sites. OracleAS Disaster Recovery supports these basic topologies:

- Symmetrical Topologies - Strict Mirror of the Production Site with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure
- Asymmetrical Topologies - Simple Asymmetric Standby Topology with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure
- Separate OracleAS Metadata Repository for OracleAS Portal with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure (the Departmental Topology)
- Distributed Application OracleAS Metadata Repositories with Non Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure
- Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology
- Redundant Single Oracle Application Server 10.1.3 Oracle Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology

For information on using RAC databases in your Disaster Recovery topology, refer to the following sections:

- Section 3.2.1, "Configuring OracleAS Disaster Recovery Where Both the Primary and Standby Sites Use Oracle Real Application Clusters Databases"
- Section 3.2.2, "Configuring OracleAS Disaster Recovery Where Only the Primary Site Uses Oracle Real Application Clusters Database (Standby Site Uses a Non-Real Application Clusters Database)"

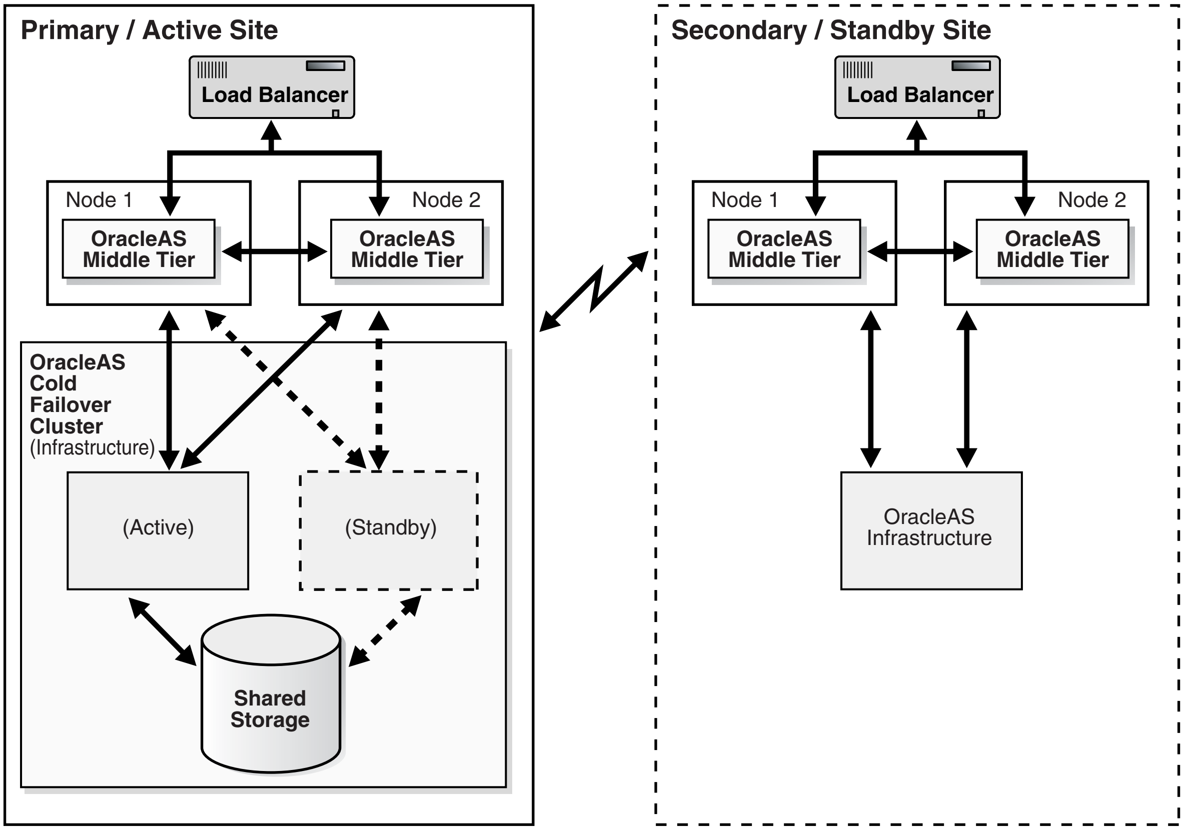
#### 1.1.3.1 Symmetrical Topologies - Strict Mirror of the Production Site with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure

For OracleAS Disaster Recovery Release 10.1.2.0.1, only the OracleAS Disaster Recovery symmetrical topology environment was supported. This OracleAS Disaster Recovery environment has two major requirements:

- The deployment must use a single default Infrastructure install that contains a collocated OracleAS Metadata Repository and Oracle Identity Management.
- The standby site has to be a strict mirror of the production site with the same number of instances (symmetrical topology).

Figure 1–1 depicts an example OracleAS Disaster Recovery solution having a symmetrical topology with a Cold Failover Cluster on the primary site. This is considered a symmetrical topology because from an Oracle Application Server perspective both sites contain two Oracle Application Server middle tiers and one Infrastructure.

***Figure 1–1 Example Oracle Application Server Site-to-Site Disaster Recovery Solution (Load Balancer Appliance is Optional If Only One Middle-Tier Node is Present)***



The procedures and steps for configuring and operating the OracleAS Disaster Recovery solution support 1 to *n* number of middle-tier hosts in the production site. The same number of middle-tier installations must exist in the standby site. The middle tiers must mirror each other in the production and standby sites.

For the OracleAS Infrastructure, a uniform number of hosts is not required (names or instances must be equal) between the production and standby sites. For example, the

OracleAS Cold Failover Cluster (Infrastructure) solution can be deployed in the production site, and a single node installation of the OracleAS Infrastructure can be deployed in the standby site as shown in Figure 1–1. This way, the production site's OracleAS Infrastructure has protection from host failure using an OracleAS Cold Failover Cluster. This solution provides hardware redundancy by utilizing a virtual hostname. Refer to Section 6.2.2, "Active-Passive High Availability Topologies" on page 6-5 of the *Oracle Application Server High Availability Guide* in the OracleAS Release 10.1.2.0.2 documentation set for more information on OracleAS Cold Failover Clusters.

The OracleAS Disaster Recovery solution is an extension to various single-site Oracle Application Server architectures. Examples of such single-site architectures include the combination of OracleAS Cold Failover Cluster (Infrastructure) and active-active Oracle Application Server middle-tier architecture. For the latest information on what single-site architectures are supported, check the Oracle Technology Network (OTN) Web site for the latest certification matrix.

http://www.oracle.com/technology/products/ias/hi_av/index.html

The following are important characteristics of the symmetric OracleAS Disaster Recovery solution:

- Middle-tier hosts are identical between the production and standby sites. In other words, each middle-tier host in the production site has an identical host in the standby site. More than one middle-tier host is recommended because this enables each set of middle-tier installations on each site to be redundant. Because they are on multiple systems, problems and outages within a site of middle-tier installations are transparent to clients.
- The OracleAS Disaster Recovery solution is restricted to identical site configuration to ensure that processes and procedures are kept the same between sites, making operational tasks easier to maintain and execute. Identical site configuration also allows for a higher success rate for manually maintaining the synchronization of Oracle Application Server component configuration files between sites.
- When the production site becomes unavailable due to a disaster, the standby site can become operational within a reasonable time. Client requests are always routed to the site that is operating in the production role. After a failover or switchover operation occurs due to an outage, client requests are routed to another site that assumes the production role. For a symmetric topology, the quality of service offered by the new production site should be the same as that offered by the original production site before the outage.
- From the standpoint of a single site, the sites are set up in active-passive configuration. An active-passive setup has one site used for production and one standby site that is initially passive (on standby). The standby site is made active only after a failover or switchover operation is performed. Since the sites are symmetrical, after failover or switchover, the original standby site can be kept active as the new production site. After repairing or upgrading the original production site, it can be made into the new standby site as long as the OracleAS Disaster Recovery site requirements are maintained. Either site should offer the same level of service to clients as the other. Note that in an active-passive setup, the standby site can be comprised of different Oracle homes that can be active on the same hosts as long as the Oracle homes being used in the Disaster Recovery environment are passive (inactive).
- For a database recovery (DBR) site as explained shortly (an Oracle Application Server 10g release (10.1.3) site is most likely not involved, whereas an Oracle Application Server 10g release (10.1.2.0.2) site is involved), the site playing the

standby role contains a physical standby of the Oracle Application Server Infrastructure coordinated by Oracle Data Guard. Oracle Application Server Guard automates the configuration and use of Oracle Data Guard together with procedures for backing up and restoring OracleAS Infrastructure configuration files and provides configuration synchronization between the production and standby sites. Switchover and failover operations allow the roles to be traded between the OracleAS Infrastructures in the two sites. Refer to Section 2.7, "Oracle Application Server Guard Operations -- Standby Site Cloning of One or More Production Instances to a Standby System", Section 2.8, "Oracle Application Server Guard Operations -- Standby Instantiation and Standby Synchronization", Section 2.9, "Runtime Operations -- Oracle Application Server Guard Switchover and Failover Operations", and Section 2.11, "Using Oracle Application Server Guard Command-Line Utility (asgctl)" for information about using the asgctl command-line interface to perform Oracle Application Server Guard administrative tasks of cloning, instantiation, synchronization, switchover, and failover in the OracleAS Disaster Recovery solution.

#### 1.1.3.2 Asymmetrical Topologies - Simple Asymmetric Standby Topology with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure

Beginning with OracleAS Disaster Recovery Release 10.1.2.0.2, support for asymmetric topologies includes support for the following simple asymmetric standby topologies:

- A standby site having reduced resources (fewer middle tiers); this means support for all production services except the scaling and high availability characteristics. This approach guarantees all services are maintained, but not scaled (see Figure 1–2 for an example of this OracleAS Disaster Recovery solution).

*Figure 1–2 Simple Asymmetric Standby with Reduced Resources*

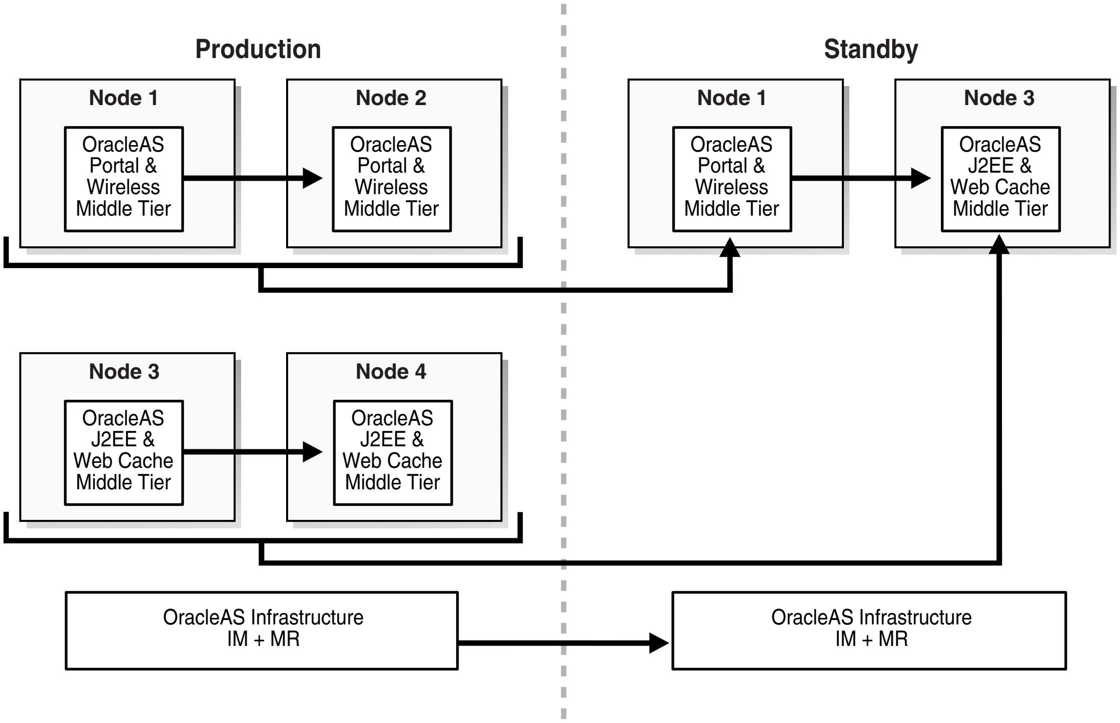

Production
Standby
Node 1
OracleAS Portal & Wireless Middle Tier
Node 2
OracleAS Portal & Wireless Middle Tier
Node 1
OracleAS Portal & Wireless Middle Tier
Node 3
OracleAS J2EE & Web Cache Middle Tier
Node 3
OracleAS J2EE & Web Cache Middle Tier
Node 4
OracleAS J2EE & Web Cache Middle Tier
OracleAS Infrastructure IM + MR
OracleAS Infrastructure IM + MR

Figure 1–2 shows a production site of four middle tier instances and one Infrastructure (collocated Oracle Identity Management and OracleAS Metadata Repository). In this example, the services and applications deployed to middle tier 1 are scaled to include middle tier 2. In addition, the services and applications deployed to middle tier 3 are scaled to include middle tier 4. To satisfy the requirements for reduced resources for disaster recovery, the scaling is not necessary at the standby site. Therefore, the services deployed at production middle tiers 1 and 2 are satisfied by a disaster recovery peer middle tier 1 at the standby site, which will be synchronized with the production middle tier 1. Likewise, the services deployed at production middle tiers 3 and 4 are satisfied by a disaster recovery peer middle tier 3 at the standby site, which will be synchronized with the production middle tier 3.

- A standby site that maintains OracleAS Disaster Recovery support for the Infrastructure services only, while the middle-tier instances are supported only through production site management. This approach guarantees that only the Infrastructure services are maintained (see Figure 1–3 for an example of this OracleAS Disaster Recovery solution).

*Figure 1–3 Simple Asymmetric Standby with Guaranteed Infrastructure*

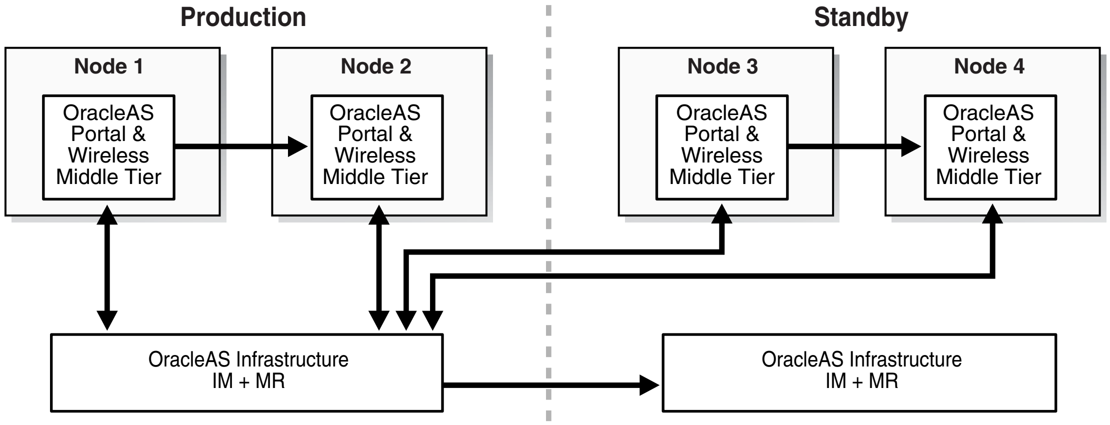

Production
Standby
Node 1
OracleAS Portal & Wireless Middle Tier
Node 2
OracleAS Portal & Wireless Middle Tier
Node 3
OracleAS Portal & Wireless Middle Tier
Node 4
OracleAS Portal & Wireless Middle Tier
OracleAS Infrastructure IM + MR
OracleAS Infrastructure IM + MR

Figure 1–3 shows a production site consisting of four middle tier instances, with two middle tier instances (1 and 2) collocated with the production Infrastructure services and two middle tier instances (3 and 4) remotely located at the standby site. In this configuration, since the middle tier instances 3 and 4 on the standby site are configured in an active/active model and are actively serving requests, they are technically members of the production topology. Only the Infrastructure services on the standby site provide passive Disaster Recovery capability.

The initial deployment of instances 3 and 4 is handled through routine production site creation. Under normal conditions, application requests are serviced by middle tier instances 1 through 4, and instances 3 and 4 have to tolerate the latency, firewall and network issues associated with this topology. After a failover, only instances 3 and 4 are available and must be able to tolerate the latency, firewall, and network issues. After a switchover, the services and applications deployed to middle tier instances 1 and 2 must be able to tolerate the latency, firewall, and network issues associated with this topology. For the Disaster Recovery instantiate and sync operations, only the Infrastructure services must be maintained using policy files.

Upon failover, instances 1 and 2 and the production Infrastructure are not available. Connection information for instances 3 and 4 would have to be updated so that requests previously routed to the production Infrastructure would be routed to the standby Infrastructure. You would use a failover policy file that ignores instances 1 and 2 to fail over the Infrastructure and start the middle tier services for instances 3 and 4 (the assumption is that instances 3 and 4 would be available at the standby site).

With this configuration, when you perform an instantiate or sync operation, use a sync policy file or instantiate policy file that ignores instances 1 through 4, since the only passive Disaster Recovery instance is the Infrastructure.

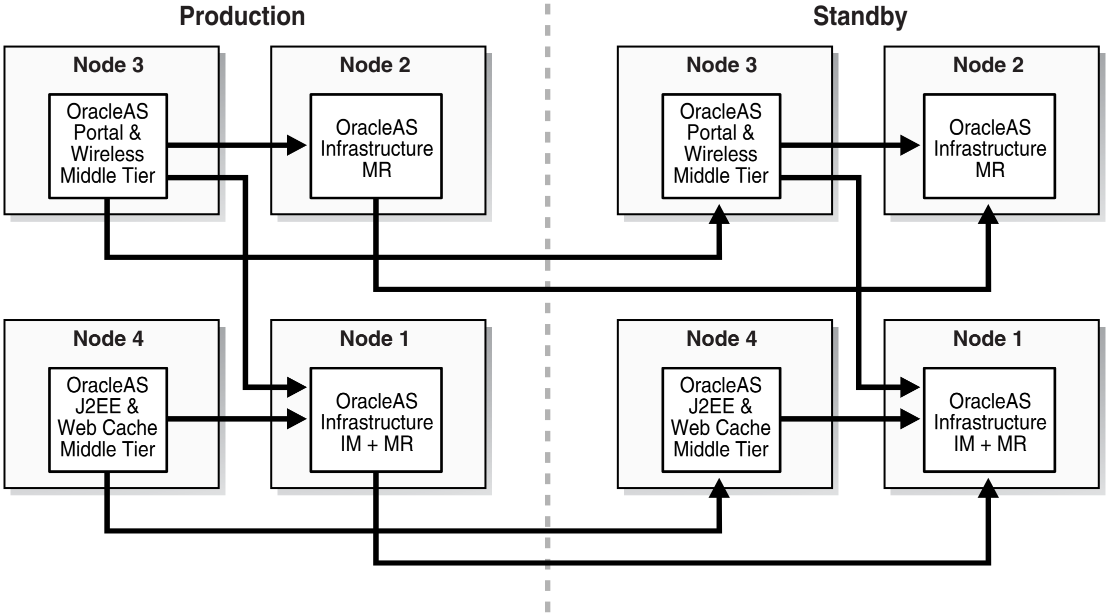
With this type of asymmetric topology, the standby site has a reduced number of Oracle homes that guarantee a certain minimum level of service capability at a reduced performance level.

#### 1.1.3.3 Separate OracleAS Metadata Repository for OracleAS Portal with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure (the Departmental Topology)

This topology (Figure 1–4), consists of an OracleAS Infrastructure with two OracleAS Metadata Repositories and multiple middle tiers. One OracleAS Metadata Repository is used by Oracle Identity Management components, such as Oracle Internet Directory and OracleAS Single Sign-On. All middle tiers use this OracleAS Metadata Repository for Oracle Identity Management services, as well as any additional middle tiers that might be added to this topology as it expands. The other OracleAS Metadata Repository is used for product metadata by the OracleAS Portal and OracleAS Wireless middle tier components. With two metadata repositories, this deployment can best be described as two DCM production farms.

An OracleAS Disaster Recovery standby configuration could be set up as either a symmetrical topology as described in Section 1.1.3.1, "Symmetrical Topologies - Strict Mirror of the Production Site with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure," thereby requiring two DCM standby farms be configured or as a simple asymmetric topology as described in Section 1.1.3.2, "Asymmetrical Topologies - Simple Asymmetric Standby Topology with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure," with service guaranteed requiring minimally that a single DCM standby farm be configured.

*Figure 1–4 Collocated Oracle Identity Management and OracleAS Metadata Repository with a Separate OracleAS Metadata Repository*


Production
Standby
Node 3
OracleAS Portal & Wireless Middle Tier
Node 2
OracleAS Infrastructure MR
Node 4
OracleAS J2EE & Web Cache Middle Tier
Node 1
OracleAS Infrastructure IM + MR
Node 3
OracleAS Portal & Wireless Middle Tier
Node 2
OracleAS Infrastructure MR
Node 4
OracleAS J2EE & Web Cache Middle Tier
Node 1
OracleAS Infrastructure IM + MR

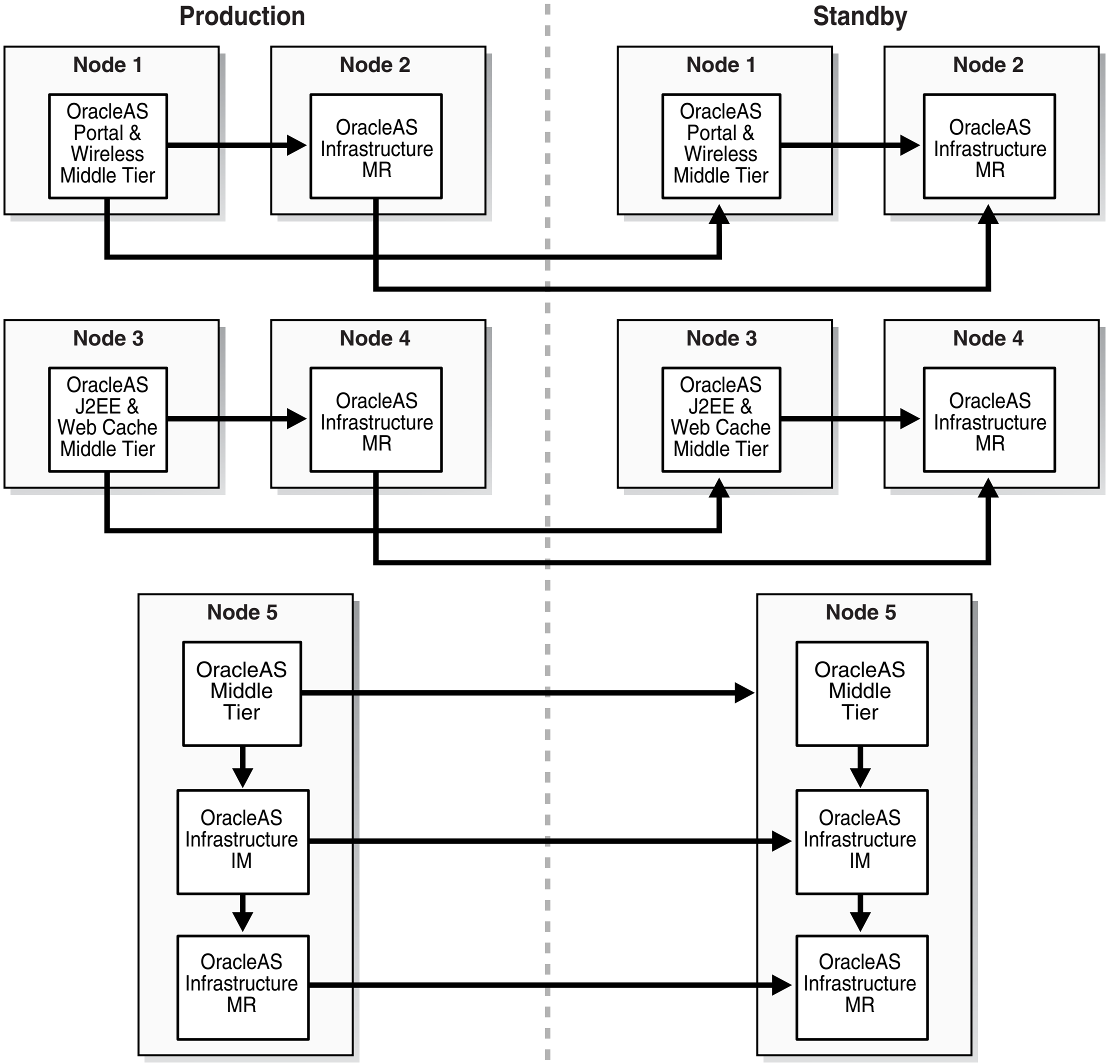
### 1.1.3.4 Distributed Application OracleAS Metadata Repositories with Non Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure

The topologies in Section 1.1.3.1, "Symmetrical Topologies - Strict Mirror of the Production Site with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure," Section 1.1.3.2, "Asymmetrical Topologies - Simple Asymmetric Standby Topology with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure," and Section 1.1.3.3, "Separate OracleAS Metadata Repository for OracleAS Portal with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure (the Departmental Topology)" describe a deployment for a default database repository collocated for both the Oracle Identity Management and OracleAS Metadata Repository Infrastructure, while Section 1.1.3.3, "Separate OracleAS Metadata Repository for OracleAS Portal with Collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure (the Departmental Topology)" also describes a topology with a separate OracleAS Metadata Repository.

In a topology with distributed application OracleAS Metadata Repositories and non collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure, the Oracle Identity Management Infrastructure and one OracleAS Metadata Repository Infrastructure are installed on separate hosts, and other OracleAS Metadata Repositories are installed to reside with respective applications on different hosts. Thus, one OracleAS Metadata Repository can be the result of a deployment using a single default Infrastructure install, while one or more OracleAS Metadata Repositories can be the result of an Oracle Application Server user using a tool, such as the OracleAS Metadata Repository Creation Assistant, to install one or more application OracleAS Metadata Repositories on one or more systems with the application data, for management or policy reasons, or both.

Figure 1–5 shows an example OracleAS Disaster Recovery solution having non collocated Oracle Identity Management and OracleAS Metadata Repository Infrastructure and distributed OracleAS Metadata Repositories.

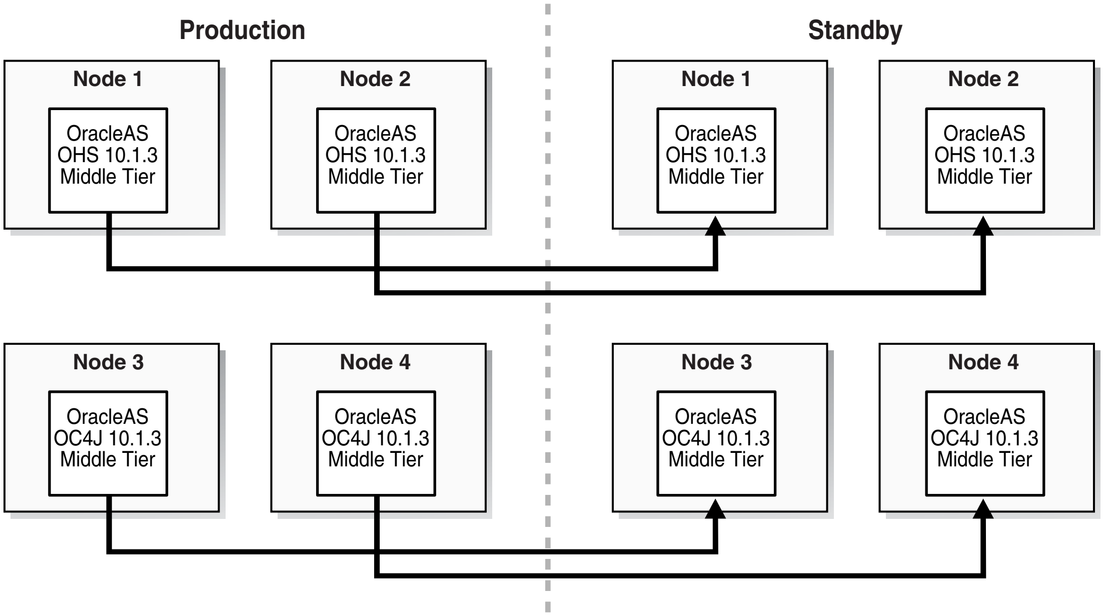
***Figure 1–5 Non-Collocated Oracle Identity Management (IM) and OracleAS Metadata Repository (MR) Infrastructure Topology with Distributed OracleAS Metadata Repositories***


Production
Standby
Node 1
OracleAS Portal & Wireless Middle Tier
Node 2
OracleAS Infrastructure MR
Node 1
OracleAS Portal & Wireless Middle Tier
Node 2
OracleAS Infrastructure MR
Node 3
OracleAS J2EE & Web Cache Middle Tier
Node 4
OracleAS Infrastructure MR
Node 3
OracleAS J2EE & Web Cache Middle Tier
Node 4
OracleAS Infrastructure MR
Node 5
OracleAS Middle Tier
OracleAS Infrastructure IM
OracleAS Infrastructure MR
Node 5
OracleAS Middle Tier
OracleAS Infrastructure IM
OracleAS Infrastructure MR

### 1.1.3.5 Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology

Figure 1–6 shows an example OracleAS Disaster Recovery solution having a configuration supporting high availability of J2EE servers in an Oracle Application Server 10.1.3 topology consisting of four nodes, known as a cluster (also called an OPMN instance). The install type of Web Server (Oracle HTTP Server or OHS) and Process Management (OPMN) is installed on nodes 1 and 2 and the install type of J2EE Server (OC4J) and Process Management (OPMN) is installed on nodes 3 and 4. There is no Identity Management.

*Figure 1–6 Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology*


Production
Standby
Node 1
OracleAS OHS 10.1.3 Middle Tier
Node 2
OracleAS OHS 10.1.3 Middle Tier
Node 1
OracleAS OHS 10.1.3 Middle Tier
Node 2
OracleAS OHS 10.1.3 Middle Tier
Node 3
OracleAS OC4J 10.1.3 Middle Tier
Node 4
OracleAS OC4J 10.1.3 Middle Tier
Node 3
OracleAS OC4J 10.1.3 Middle Tier
Node 4
OracleAS OC4J 10.1.3 Middle Tier

Beginning with Oracle Application Server 10.1.3, a new dynamic node discovery mechanism is operational within Oracle Notification Server (ONS), a component of OPMN. Dynamic node discovery enables the cluster to manage itself. When a new ONS node is added to the cluster, each existing ONS node announces its presence with a multicast message and each existing node then adds the new node and its connection information to its map of the current cluster, while at the same time the new ONS node adds all the existing nodes to its map. This fulfills one of the requirements for OracleAS Disaster Recovery that the Oracle Application Server cluster be currently configured. In this case, the OPMN configuration file, `opmn.xml`, is updated whenever a new Oracle Application Server server node is added to or removed from the cluster. This clustering configuration applies to all instances of OracleAS Server components, including OHS and OC4J, installed on the node.

By default, Oracle Application Server Guard is installed in each Oracle home on each node in the Oracle Application Server cluster. When an Oracle Application Server Guard client connects to an Oracle Application Server Guard server, and the Disaster Recovery Administrator performs a discover topology within farm command, Oracle Application Server Guard utilizes OPMN to discover all the instances on nodes within the cluster, creates the Disaster Recovery `topology.xml` file on the Oracle Application Server Guard server, and then propagates this file to all systems across the Disaster Recovery production topology. Any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby topology.

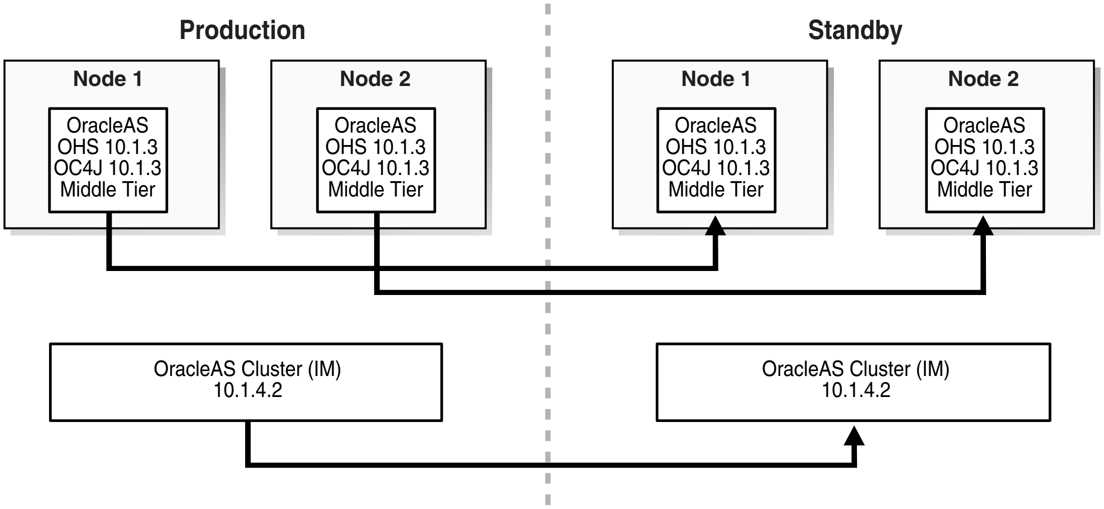
Thereafter, the Disaster Recovery Administrator can use the discover topology within farm command following the addition or removal of one or more nodes to the Oracle Application Server cluster knowing that the ONS dynamic node discovery mechanism will automatically manage the cluster configuration information and keep it current. See *Oracle Containers for J2EE Configuration and Administration Guide* for more information about the ONS dynamic discovery mechanism. You can also manage instances in the local topology file using the add instance and remove instance commands, and if specified propagate this updated local topology file to all instances in the Disaster Recovery production environment. Any Oracle Application Server

Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby environment. See Section 2.5, "Discovering Oracle Application Server 10.1.3 Instances in Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology" for a use case example. Thus, the key point is that you must perform a discover topology within farm command if any nodes have been added to the cluster and if any instances have been added to these nodes that are part of your Disaster Recovery environment.

#### 1.1.3.6 Redundant Single Oracle Application Server 10.1.3 Oracle Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology

Figure 1–7 shows an example OracleAS Disaster Recovery solution supporting a mixed version topology, where Oracle Application Server 10.1.3 instances are integrated into an existing Oracle Internet Directory (OID) based topology (OracleAS Cluster (IM) 10.1.4.2). In this case, multiple Oracle Application Server 10g (10.1.3) installations of the Integrated Web server, J2EE Server, and OPMN installation option in a single Oracle home on individual systems create the redundant single Oracle Application Server 10.1.3 Oracle home J2EE topology.

***Figure 1–7 Redundant Single Oracle Homes J2EE Topology Plus OracleAS Cluster (IM) 10.1.4.2 (OracleAS Mixed Version Topology)***


Production
Standby
Node 1
OracleAS OHS 10.1.3 OC4J 10.1.3 Middle Tier
Node 2
OracleAS OHS 10.1.3 OC4J 10.1.3 Middle Tier
Node 1
OracleAS OHS 10.1.3 OC4J 10.1.3 Middle Tier
Node 2
OracleAS OHS 10.1.3 OC4J 10.1.3 Middle Tier
OracleAS Cluster (IM) 10.1.4.2
OracleAS Cluster (IM) 10.1.4.2

By default, Oracle Application Server Guard is installed in each Oracle home. When an Oracle Application Server Guard client connects to an Oracle Application Server Guard server in the IM 10.1.4.2 OracleAS topology, and performs a discover topology command, Oracle Application Server Guard utilizes OID and automatically recognizes all Oracle Application Server 10.1.4.2 instances within the existing OID based topology. The discover topology operation creates the Disaster Recovery topology file and propagates it to all instances across the production topology. Any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby topology.

A Disaster Recovery Administrator can use the asgctl add instance or remove instance command to add or remove from the local topology file single Oracle Application Server 10.1.3 J2EE instances to or from an existing OID based 10.1.4.2 production topology. With either operation, the local topology file is updated and if specified, the local updated topology file is propagated to all instances across the production topology. Any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby topology. The resulting topology is then described as a redundant single Oracle Application Server 10.1.3 Oracle home J2EE topology integrated with an existing OID based topology (OracleAS Cluster (IM) 10.1.4.2). See Section 2.6, "Adding or Removing Oracle Application Server 10.1.3 Instances to Redundant Single Oracle Application Server 10.1.3 Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology" for a use case example. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

## 1.2 Preparing the OracleAS Disaster Recovery Environment

Prior to the installation of OracleAS software for the OracleAS Disaster Recovery solution, a number of system level configurations are required or optional as specified. The tasks that accomplish these configurations are:

- Section 1.2.1, "Planning and Assigning Hostnames"
- Section 1.2.2, "Configuring Hostname Resolution"
- Chapter 7, "Secure Shell (SSH) Port Forwarding" (optional)

This section covers the steps needed to perform these tasks for the symmetrical topology. These same steps are also applicable to topologies for non collocated Oracle Identity Management and OracleAS Metadata Repository with or without distributed OracleAS Metadata Repositories.

### 1.2.1 Planning and Assigning Hostnames

Before performing the steps to set up the physical and network hostnames, plan the physical and network hostnames you want to use with respect to the entire OracleAS Disaster Recovery solution. The overall approach to planning and assigning hostnames must meet the following goals:

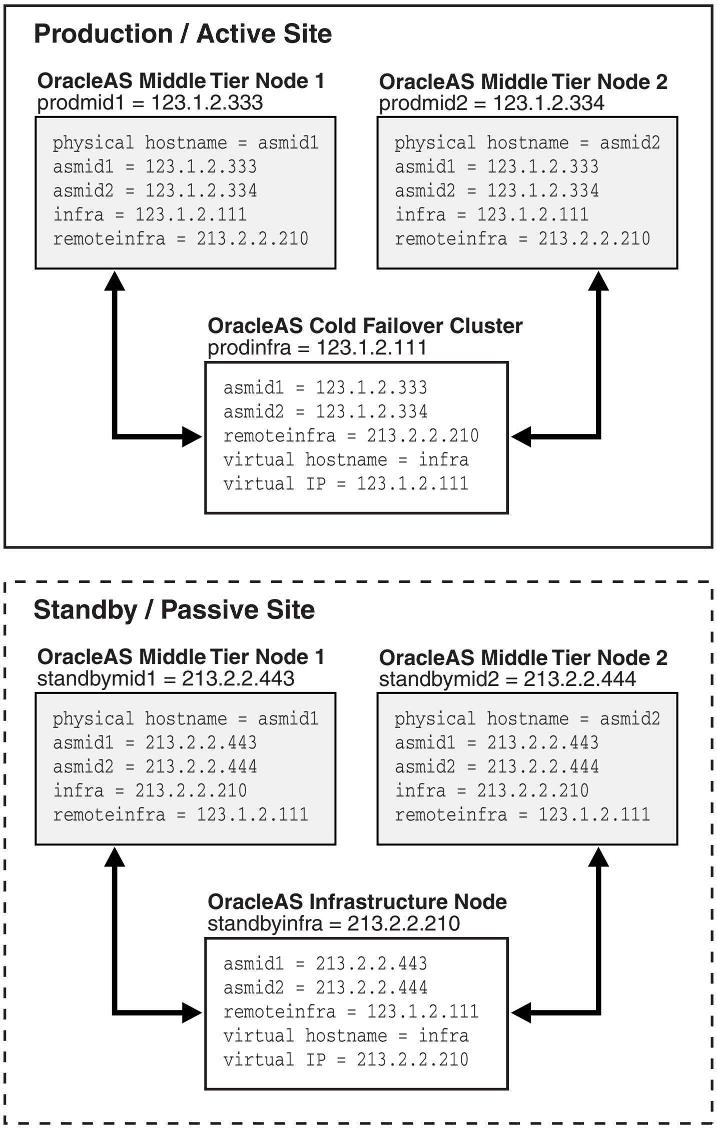
- OracleAS components in the middle tier and OracleAS Infrastructure must use the same physical hostnames in their configuration settings regardless of whether the components are in the production or standby site. In addition, you must also create a virtual hostname for the physical hostname of the OracleAS Infrastructure.

  For example, if the physical hostname asmid1 is used for a middle-tier host in the production site, the same physical hostname, asmid1, should be used for the peer middle-tier host in the standby site. Likewise, if the virtual hostname infra is used for the OracleAS Infrastructure host on the production site, then the same virtual hostname infra should be used for the OracleAS Infrastructure host on the standby site.

- No changes to hostnames (physical, network, or virtual) are required when the standby site takes over the production role. However, a DNS switchover must be performed to redirect client requests transparently to the new site that has assumed the production site. See Section 1.5, "Wide Area DNS Operations" for more information about performing a DNS switchover.

Figure 1–8 illustrates the process of planning and assigning hostnames.

*Figure 1–8 Name Assignment Example in the Production and Standby Sites*



In Figure 1–8, two middle-tier nodes exist in the production site. The OracleAS Infrastructure can be a single node or an OracleAS Cold Failover Cluster solution (represented by a single virtual hostname and a virtual IP, as for a single node OracleAS Infrastructure). The common names in the two sites are the physical hostnames of the middle-tier nodes and the virtual hostname of the OracleAS Infrastructure. Table 1–6 shows the physical, network, and virtual hostnames in the example:

*Table 1–6 Physical, Network, and Virtual Hostnames in Figure 1–8*

| Physical Hostnames | Virtual Hostname | Network Hostnames[1] |
|---|---|---|
| asmid1 | - | prodmid1, standbymid1 |
| asmid2 | - | prodmid2, standbymid2 |
| - [2] | infra | prodinfra, standbyinfra |

[1] The network hostname is the hostname defined in domain name system (DNS). A network hostname is the name by which a host is known to the network.

[2] In this example, the physical hostname for the OracleAS Infrastructure host is the network hostname.

The following sections and the Glossary explain that physical, network, and virtual hostnames have different purposes in the OracleAS Disaster Recovery solution. These hostnames are also set up differently. See the following sections for more information on physical, network, and virtual hostnames:

- Section 1.2.1.1, "Physical Hostnames"
- Section 1.2.1.2, "Network Hostnames"
- Section 1.2.1.3, "Virtual Hostname"

### 1.2.1.1 Physical Hostnames

When you are installing an Oracle Application Server middle-tier instance (non-Infrastructure instance) on a host, there are different ways of specifying a physical hostname for that host, depending on the release of Oracle Application Server that you are installing. The Oracle Universal Installer will use the hostname string that is discovered during the installation in the OracleAS Disaster Recovery configuration.

The subsections in this section describe the different ways of specifying a physical hostname for a host prior to installing an Oracle Application Server middle-tier instance for a particular Oracle Application Server release on the host.

Oracle recommends that you use the same physical hostname for a production site middle-tier host and its peer middle-tier host at the standby site. However, if the middle-tier hosts at the production site already have installed software, changing the physical hostnames of those hosts may break the software. To avoid breaking already installed software, you can optionally continue to use the network hostnames for the production site middle-tier hosts and not define physical hostnames for them. In this case, though, you must change the physical hostnames for the standby site middle-tier hosts to match the network hostnames for their production site peer hosts.

**1.2.1.1.1 Creating Physical Hostnames Before Installing Application Server Releases 10.1.2.0.2 and 10.1.2.1** The installation procedures for Oracle Application Server middle-tier instances (non-Infrastructure instances) for releases 10.1.2.0.2 and 10.1.2.1 do not allow you to specify a physical hostname as part of the installation. Therefore, you must follow the steps below to specify a physical hostname for middle-tier hosts prior to these installations:

Follow these steps to change the physical hostname of a Solaris host:

1. On a Solaris middle-tier host, check the setting for the existing hostname as follows:

   ```
   prompt> hostname
   ```

2. Use a text editor, such as `vi`, to edit the name in `/etc/nodename` to your planned physical hostname.
3. For each middle-tier host, restart it for the change to take effect.
4. Repeat Step 1 to verify the correct physical hostname has been set. For example:

   ```
   prompt> hostname
   asmid1
   ```

5. Repeat the previous steps to create a new physical hostname for each Solaris middle-tier host at the production and standby sites.

Follow these steps to change the physical hostname of a Linux host:

1. On a Linux middle-tier host, check the setting for the existing hostname by issuing the `hostname` command as follows:

   ```
   prompt> hostname
   ```

2. Use the `hostname` command and specify the new physical hostname you want to create for the host (you must be root to issue the `hostname` command). The following example shows how to create a physical hostname of asmid1 for the host:

   ```
   prompt> hostname asmid1
   ```

3. Repeat Step 1 to verify the correct hostname has been set:

   ```
   prompt> hostname
   asmid1
   ```

4. Add the new physical hostname as an entry in the `/etc/hosts` file for the host. An `/etc/hosts` file is used for local hostnaming file resolution. The supported methods for configuring OracleAS Disaster Recovery hostname resolution are described in Section 1.2.2, "Configuring Hostname Resolution." In this example, adding the following `/etc/hosts` file entry for the host with an IP address of 123.1.2.333 creates a physical hostname of asmid1 for that host:

   ```
   123.1.2.333 asmid1.oracle.com asmid1
   ```

5. Repeat the previous steps to create a new physical hostname for each Linux middle-tier host at the production and standby sites.

> **Note:** For other UNIX variants, consult your system administrator for equivalent commands.

Follow these steps to change the physical hostname of a Windows host:

> **Note:** The user interface elements in your version of Windows may vary from those described in the following steps.

1. In the **Start** menu, select **Control Panel**.
2. Double-click the **System** icon.
3. Select the **Advanced** tab.
4. Select **Environment Variables**.
5. Under the **User variables** for the installer account, select **New** to create a new variable.
6. Enter the name of the variable as "`_CLUSTER_NETWORK_NAME_`".
7. For the value of this variable, enter the planned physical hostname, for example, admid1.
8. Add the new physical hostname as an entry in the `C:\WINDOWS\system32\drivers\etc\hosts` file for the host. For example, adding the following entry in the `\etc\hosts` file for the host with an IP address of 123.1.2.333 creates a physical hostname of asmid1 for that host:

```
123.1.2.333 asmid1.oracle.com asmid1
```

9. Repeat the previous steps to create a new physical hostname for each Windows middle-tier host at the production and standby sites.

##### 1.2.1.1.2 Creating Physical Hostnames Before Installing Application Server Releases 10.1.2.2 and 10.1.2.3

To create a new physical hostname for a host prior to installing Oracle Application Server middle-tier instances (non-Infrastructure instances) for releases 10.1.2.2 and 10.1.2.3 on the host, you can specify the physical hostname as an entry in the `/etc/hosts` file for the host on which you are installing the middle-tier instance. For example, the following entry in the `/etc/hosts` file for the host with an IP address of 123.1.2.333 creates a physical hostname of asmid1 for that host:

```
123.1.2.333 asmid1.oracle.com asmid1
```

If you do not create a new physical hostname by specifying it as an entry in the `/etc/hosts` file for the host, then the Oracle Universal Installer will use the name returned by the `hostname` command on the host as the physical hostname for the host.

##### 1.2.1.1.3 Creating Physical Hostnames Before Installing Application Server Releases 10.1.3.0 through 10.1.3.3

Use one of these options to create a new physical hostname for a host prior to installing Oracle Application Server middle-tier instances (non-Infrastructure instances) for releases 10.1.3.0 through 10.1.3.3 on the host:

- On UNIX platforms, you can set the VIRTUAL_HOST_NAME environment variable prior to the installation to create a new physical hostname for the middle-tier system on which you are installing Oracle Application Server (even though you use the VIRTUAL_HOST_NAME environment variable, this variable creates a physical hostname on a middle-tier host). For example, to specify a physical hostname of asmid1 or a middle-tier host Linux host, log into the host and issue the following command prior to installing Oracle Application Server:

  ```
  setenv VIRTUAL_HOST_NAME asmid1
  ```

- You can create a new physical hostname for the host prior to the installation by specifying it as an entry in the `/etc/hosts` file for the host on which you are installing the middle-tier instance. For example, adding the following entry in the `/etc/hosts` file for the host with an IP address of 123.1.2.333 creates a physical hostname of asmid1 for that host:

  ```
  123.1.2.333 asmid1.oracle.com asmid1
  ```

If you do not create a new physical hostname by using either the VIRTUAL_HOST_NAME variable or by specifying the new physical hostname as an entry in the `/etc/hosts` file for the middle-tier host, then the Oracle Universal Installer will use the name returned by the `hostname` command on the host as the physical hostname for the host.

#### 1.2.1.2 Network Hostnames

The network hostnames used in the OracleAS Disaster Recovery solution are defined in domain name system (DNS). These hostnames are visible in the network that the solution uses and are resolved through DNS to the appropriate hosts by the assigned IP address in the DNS system. You must add these network hostnames and their corresponding IP addresses to the DNS system.

Using the example in Figure 1–8, the following additions should be made to the DNS system serving the entire network that encompasses the production and standby sites:

```
prodmid1.oracle.com        IN     A     123.1.2.333
```

```
prodmid2.oracle.com        IN     A     123.1.2.334
prodinfra.oracle.com       IN     A     123.1.2.111
standbymid1.oracle.com     IN     A     213.2.2.443
standbymid2.oracle.com     IN     A     213.2.2.444
standbyinfra.oracle.com    IN     A     213.2.2.210
```

#### 1.2.1.3 Virtual Hostname

As defined in this section and the Glossary, virtual hostname applies to the OracleAS Infrastructure only. It is specified during installation of the OracleAS Infrastructure. When you run the OracleAS Infrastructure installation type, a screen called **Specify Virtual Hostname** appears to provide a text box to enter the virtual hostname of the OracleAS Infrastructure that is being installed. Refer to the *Oracle Application Server Installation Guide* for more details.

For the example in Figure 1–8, when you install the production site's OracleAS Infrastructure, enter its virtual hostname, infra, on the **Specify Virtual Hostname** screen. Enter the same virtual hostname when you install the standby site's OracleAS Infrastructure.

**Note:** If the OracleAS Infrastructure is installed in an OracleAS Cold Failover Cluster solution, the virtual hostname is the name that is associated with the virtual IP of the OracleAS Cold Failover Cluster.

For high availability deployment with multiple Oracle Application Server instances using a load balancer on the production site, but no load balancer on the standby site, the virtual hostname is the DNS based virtual hostname for the load balancer, for example `lbr01.us.oracle.com`. For more information, see the white paper on using load balancers and OracleAS High Availability on the Oracle Technology Network (OTN).

#### 1.2.1.4 Virtual Hostname Aliases

When setting up a Disaster Recovery solution in environments involving both Oracle RAC databases and non-RAC databases, it is recommended to define a virtual hostname that can be used as an alias at both the production site and standby site. The alias must be defined in the `/etc/hosts` file on Windows and UNIX platforms on each node running a database instance.

In a Disaster Recovery environment, the site that actively accepts connections is the production site. At the completion of a successful failover or switchover operation, the standby site becomes the new production site.

This section includes an example of defining an alias for database hosts stajo01 and stajo02. Table 1–7 shows the database hostnames and the connect strings for the databases before the alias is defined:

*Table 1–7 Database Hostnames and Connect Strings*

| Site | Database Hostname | Database Connect String |
|---|---|---|
| Primary | stajo01.us.oracle.com | stajo01.us.oracle.com:1521:orcl |
| Standby | stajo02.us.oracle.com | stajo02.us.oracle.com:1521:orcl |

In this example, all database connect strings on the production site take the form "stajo01.us.oracle.com:1521:orcl." After a failover or switchover operation, this connect string must be changed to "stajo02.us.oracle.com:1521:orcl." However, by creating an alias of "proddb1" for the database hostname as shown in Table 1–8, you can avoid manually changing the connect strings, which enables seamless failovers and switchovers:

***Table 1–8 Specifying an Alias for a Database Host***

| Site | Database Hostname | Alias | Database Connect String |
|---|---|---|---|
| Production | stajo01.us.oracle.com | proddb1.us.oracle.com | proddb1.us.oracle.com:1521:orcl |
| Standby | stajo02.us.oracle.com | proddb1.us.oracle.com | proddb1.us.oracle.com:1521:orcl |

In this example, the production site database hostname and the standby site database hostname are aliased to "proddb1.us.oracle.com" and the connect strings on the production site and the standby site can take the form "proddb1.us.oracle.com:1521:orcl". On failover and switchover operations, the connect string does not need to change, thus enabling a seamless failover and switchover.

The format for specifying virtual hostnames as aliases in /etc/hosts file entries is:

```
<IP>    <ALIAS WITH DOMAIN> <ALIAS>    <HOSTNAME WITH DOMAIN> <HOSTNAME>
```

In this example, you create an alias with the virtual hostname of proddb1 for host stajo01 at the production site and for host stajo02 at the standby site. The hosts file entry should specify the fully qualified virtual hostname with the `<ALIAS WITH DOMAIN>` parameter, the short virtual hostname with the `<ALIAS>` parameter, the fully qualified physical hostname with the `<HOSTNAME WITH DOMAIN>` parameter, and the short physical hostname with the `<HOSTNAME>` parameter.

So, in the /etc/hosts files at the production site, make sure the entry for host stajo01 looks like this:

```
152.68.196.213   proddb1.us.oracle.com proddb1   stajo01.us.oracle.com stajo01
```

And, in the /etc/hosts files at the standby site, make sure the entry for host stajo02 looks like this:

```
140.87.25.40   proddb1.us.oracle.com proddb1   stajo02.us.oracle.com stajo02
```

## 1.2.2 Configuring Hostname Resolution

In the OracleAS Disaster Recovery solution, you must configure hostname resolution in one of two ways to resolve the hostnames you planned and assigned in Section 1.2.1, "Planning and Assigning Hostnames." The two methods of configuring hostname resolution are:

- Section 1.2.2.1, "Using Local Hostnaming File Resolution"
- Section 1.2.2.2, "Using DNS Resolution"

In UNIX, the order of the method of hostname resolution is specified using the "`hosts`" parameter in the file `/etc/nsswitch.conf`. The following is an example of the `hosts` entry:

```
hosts:     files dns nis
```

In the previous statement, local hostname file resolution is preferred over DNS and NIS (Network Information Service) resolutions. When a hostname is required to be resolved to an IP address, the /etc/hosts file (UNIX) or C:\WINDOWS\system32\drivers\etc\hosts file is consulted first. In this example, if a hostname cannot be resolved using local hostnaming resolution, then DNS is used. (NIS resolution is not used for the OracleAS Disaster Recovery solution.) Refer to your UNIX system documentation to find out more about name resolution using the file /etc/nsswitch.conf.

After you choose and configure either local hostname file resolution or DNS resolution for Disaster Recovery on UNIX, make sure that the resolution method you configured is the first entry for the hosts parameter in the /etc/nsswitch.conf file on each host in your Disaster Recovery production site and standby site. For example, if you chose local hostname file resolution, the hosts parameter must look like this in the /etc/nsswitch.conf file on each host:

```
hosts:     files dns nis
```

If you chose DNS resolution, the hosts parameter must look like this in the /etc/nsswitch.conf file on each host:

```
hosts:     dns files nis
```

In Windows, the method of ordering hostname resolution varies depending on the Windows version. Refer to the documentation for your version of Windows for the appropriate steps.

#### 1.2.2.1 Using Local Hostnaming File Resolution

This method of hostname resolution relies on a local hostnaming file to contain the requisite hostname-to-IP address mappings. In UNIX, this file is /etc/hosts. In Windows, this file is C:\WINDOWS\system32\drivers\etc\hosts.

To use the local hostnaming file to resolve hostnames for the OracleAS Disaster Recovery solution in UNIX for each middle tier and OracleAS Infrastructure host in both the production and standby sites, perform the following steps on each host at the production and standby sites:

1. Use a text editor, such as vi, to edit the /etc/nsswitch.conf file. With the "hosts:" parameter, specify "files" as the first choice for hostname resolution.
2. Edit the /etc/hosts file to include the following:
   - The physical hostnames and the correct IP addresses for all middle-tier nodes in the current site. The first entry must be the hostname and IP address of the current node.

     **Note:** When making entries in the hosts file, make sure the intended hostname is positioned in the second column of the hosts file; otherwise, an asgctl verify topology with <host> operation will fail indicating that the production topology is not symmetrical with the standby topology. See Section A.1.7, "Failure of Farm Verification Operation with Standby Farm" for more information about troubleshooting and resolving this type of problem.

     For example, if you are editing the /etc/hosts file of a middle-tier node in the production site, enter all the middle-tier physical hostnames and their IP addresses in the production site beginning the list with the current host. (You should also include fully qualified hostnames in addition to the abbreviated hostnames. See Table 1–9.)

- The virtual hostname of the OracleAS Infrastructure in the current site.

  For example, if you are editing the `/etc/hosts` of a middle-tier node in the standby site, enter the virtual hostname, fully qualified and abbreviated, and the IP address of the OracleAS Infrastructure host in the standby site.

3. Restart each host after editing the files mentioned in the previous steps.
4. From each host, use the `ping` command for each physical hostname that is valid in its particular site to ensure that the IP addresses have been assigned correctly.

   For the example in Figure 1–8, on the asmid1 host, use the following commands in succession:

   ```
   ping asmid1
   ```

   The returned IP address should be 123.1.2.333.

   ```
   ping asmid2
   ```

   The returned IP address should be 123.1.2.334.

   ```
   ping infra
   ```

   The returned IP address should be 123.1.2.111.

   > **Note:** Some UNIX variants, such as Solaris, require the `-s` option to return an IP address.

Using the example in Figure 1–8, Table 1–9 shows that the `/etc/hosts` file entries on each production node contains the required entries for each UNIX host. The entries in the Windows `C:\WINDOWS\system32\drivers\etc\hosts` file should be similar.

***Table 1–9 Network and Virtual Hostname Entries in Each `/etc/hosts` File of Example in Figure 1–8***

| Host | Entries in /etc/hosts |
|---|---|
| `asmid1` in production site | `123.1.2.333 asmid1.oracle.com asmid1`<br>`123.1.2.334 asmid2.oracle.com asmid2`<br>`123.1.2.111 infra.oracle.com infra`<br>`213.2.2.210 remoteinfra.oracle.com remoteinfra` |
| `asmid2` in production site | `123.1.2.334 asmid2.oracle.com asmid2`<br>`123.1.2.333 asmid1.oracle.com asmid1`<br>`123.1.2.111 infra.oracle.com infra`<br>`213.2.2.210 remoteinfra.oracle.com remoteinfra` |
| `infra` in production site | `123.1.2.111 infra.oracle.com infra`<br>`123.1.2.333 asmid1.oracle.com asmid1`<br>`123.1.2.334 asmid2.oracle.com asmid2`<br>`213.2.2.210 remoteinfra.oracle.com remoteinfra` |
| `asmid1` in standby site | `213.2.2.443 asmid1.oracle.com asmid1`<br>`213.2.2.444 asmid2.oracle.com asmid2`<br>`213.2.2.210 infra.oracle.com infra`<br>`123.1.2.111 remoteinfra.oracle.com remoteinfra` |
| `asmid2` in standby site | `213.2.2.444 asmid2.oracle.com asmid2`<br>`213.2.2.443 asmid1.oracle.com asmid1`<br>`213.2.2.210 infra.oracle.com infra`<br>`123.1.2.111 remoteinfra.oracle.com remoteinfra` |

*Table 1–9 (Cont.) Network and Virtual Hostname Entries in Each `/etc/hosts` File of Example in Figure 1–8*

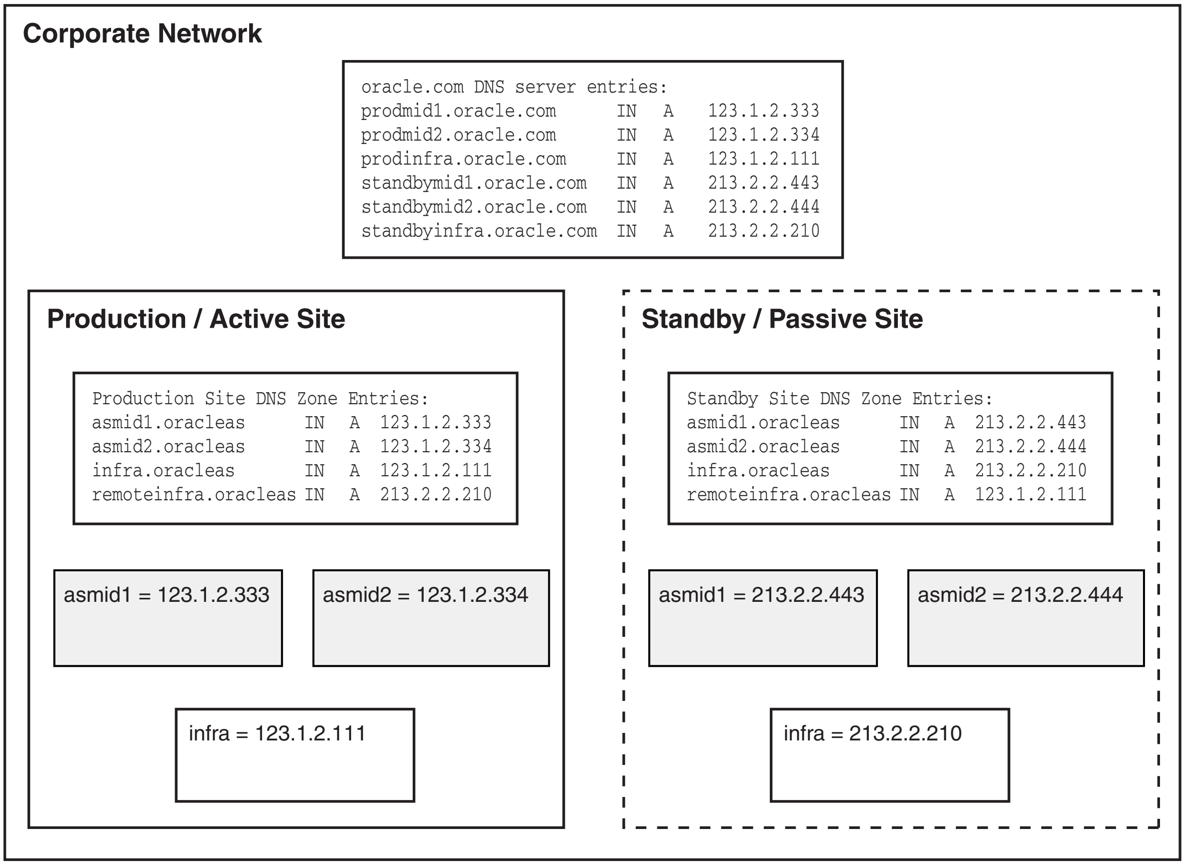
| Host | Entries in /etc/hosts |
|---|---|
| `infra` in standby site | `213.2.2.210 infra.oracle.com infra`<br>`213.2.2.443 asmid1.oracle.com asmid1`<br>`213.2.2.444 asmid2.oracle.com asmid2`<br>`123.1.2.111 remoteinfra.oracle.com remoteinfra` |

### 1.2.2.2 Using DNS Resolution

To set up the OracleAS Disaster Recovery solution to use DNS hostname resolution, you must set up site-specific DNS servers in the production and standby sites in addition to the overall corporate DNS servers (usually more than one DNS server exists in a corporate network for redundancy). Figure 1–9 provides an overview of this setup.

> **See Also:** Chapter 6, "Setting Up a DNS Server" for instructions on how to set up a DNS server in a UNIX environment.

*Figure 1–9 DNS Resolution Topology Overview*



```
oracle.com DNS server entries:
prodmid1.oracle.com      IN   A   123.1.2.333
prodmid2.oracle.com      IN   A   123.1.2.334
prodinfra.oracle.com     IN   A   123.1.2.111
standbymid1.oracle.com   IN   A   213.2.2.443
standbymid2.oracle.com   IN   A   213.2.2.444
standbyinfra.oracle.com  IN   A   213.2.2.210
```

```
Production Site DNS Zone Entries:
asmid1.oracleas      IN  A  123.1.2.333
asmid2.oracleas      IN  A  123.1.2.334
infra.oracleas       IN  A  123.1.2.111
remoteinfra.oracleas IN  A  213.2.2.210
```

```
Standby Site DNS Zone Entries:
asmid1.oracleas      IN  A  213.2.2.443
asmid2.oracleas      IN  A  213.2.2.444
infra.oracleas       IN  A  213.2.2.210
remoteinfra.oracleas IN  A  123.1.2.111
```

For the topology in Figure 1–9 to work, the following requirements and assumptions must be made:

- The DNS servers for the production and standby sites must not be aware of each other. They make non authoritative lookup requests to the overall corporate DNS servers if they fail to resolve any hostnames within their specific sites.

- The production site and standby site DNS servers must contain entries for middle-tier physical hostnames and OracleAS Infrastructure virtual hostnames. Each DNS server contains entries of only the hostnames within its own site. The sites have a common domain name that is different from that of the overall corporate domain name.
- The overall corporate DNS servers contain network hostname entries for the middle-tier hosts and OracleAS Infrastructure hosts of both production and standby sites.
- In UNIX, the `/etc/hosts` file in each host does not contain entries for the physical, network, or virtual hostnames of any host in either the production or standby site. In Windows, this applies to the file `C:\WINDOWS\system32\drivers\etc\hosts`.

To set up the OracleAS Disaster Recovery solution for DNS resolution, follow these steps:

1. Configure each of the overall corporate DNS servers with the network hostnames of all the hosts in the production and standby sites. Using the example presented in Figure 1–8, the following entries are made:

```
prodmid1.oracle.com      IN    A    123.1.2.333
prodmid2.oracle.com      IN    A    123.1.2.334
prodinfra.oracle.com     IN    A    123.1.2.111
standbymid1.oracle.com   IN    A    213.2.2.443
standbymid2.oracle.com   IN    A    213.2.2.444
standbyinfra.oracle.com  IN    A    213.2.2.210
```

2. For each site, production and standby, create a unique DNS zone by configuring a DNS server as follows:

   a. Select a unique domain name to use for the two sites that is different from the corporate domain name. As an example, use the name "`oracleas`" for the domain name for the two sites in Figure 1–8. The high level corporate domain name is `oracle.com`.

   b. Configure the DNS server in each site to point to the overall corporate DNS servers for unresolved requests.

   c. Populate the DNS servers in each site with the physical hostnames of each middle-tier host and the virtual hostname of each OracleAS Infrastructure host. Include the domain name selected in the previous step.

   For the example in Figure 1–8, the entries are as follows:

   For the DNS server on the production site:

```
asmid1.oracleas      IN    A    123.1.2.333
asmid2.oracleas      IN    A    123.1.2.334
infra.oracleas       IN    A    123.1.2.111
```

   For the DNS server on the standby site:

```
asmid1.oracleas      IN    A    213.2.2.443
asmid2.oracleas      IN    A    213.2.2.444
infra.oracleas       IN    A    213.2.2.210
```

**Note:** If you are using a load balancer, you must alias the IP addresses inside the host file with DNS based virtual hostnames. This is essential for Oracle Application Server Guard on the local host to perform a local write of the topology file from a discover topology within farm command and to correctly perform an add instance command and update the topology file.

If you are using the OracleAS Cold Failover Cluster solution for the OracleAS Infrastructure in either site, enter the cluster's virtual hostname and virtual IP address. For example, in the previous step, `infra` is the virtual hostname and `123.1.2.111` is the virtual IP of the cluster in the production site. For more information on the OracleAS Cold Failover Cluster solution, see Section 6.2.2, "Active-Passive High Availability Topologies" on page 6-5 of the *Oracle Application Server High Availability Guide* in the Oracle Application Server Release 10.1.2.0.2 documentation set.

#### 1.2.2.2.1 Additional DNS Server Entries for Oracle Data Guard

Because Oracle Application Server Guard automates the use of Oracle Data Guard technology, which is used to synchronize the production and standby OracleAS Infrastructure databases, the production OracleAS Infrastructure must be able to reference the standby OracleAS Infrastructure and conversely.

For this to work, the IP address of the standby OracleAS Infrastructure host must be entered in the production site's DNS server with a hostname that is unique to the production site. Similarly, the IP address of the production OracleAS Infrastructure host must be entered in the standby site's DNS server with the same hostname. These DNS entries are required because Oracle Data Guard uses TNS Names to direct requests to the production and standby OracleAS Infrastructures. Hence, the appropriate entries must also be made to the `tnsnames.ora` file. Additionally, Oracle Application Server Guard asgctl command-line commands must reference the network hostnames.

Using the example in Figure 1–8 and assuming that the selected name for the remote OracleAS Infrastructure is "remoteinfra," the entries for the DNS server in the production site are:

```
asmid1.oracleas        IN    A    123.1.2.333
asmid2.oracleas        IN    A    123.1.2.334
infra.oracleas         IN    A    123.1.2.111
remoteinfra.oracleas   IN    A    213.2.2.210
```

And, in the standby site, the DNS server entries should be as follows:

```
asmid1.oracleas        IN    A    213.2.2.443
asmid2.oracleas        IN    A    213.2.2.444
infra.oracleas         IN    A    213.2.2.210
remoteinfra.oracleas   IN    A    123.1.2.111
```

## 1.3 Preparing the OracleAS Disaster Recovery Environment for an Asymmetrical Standby Site

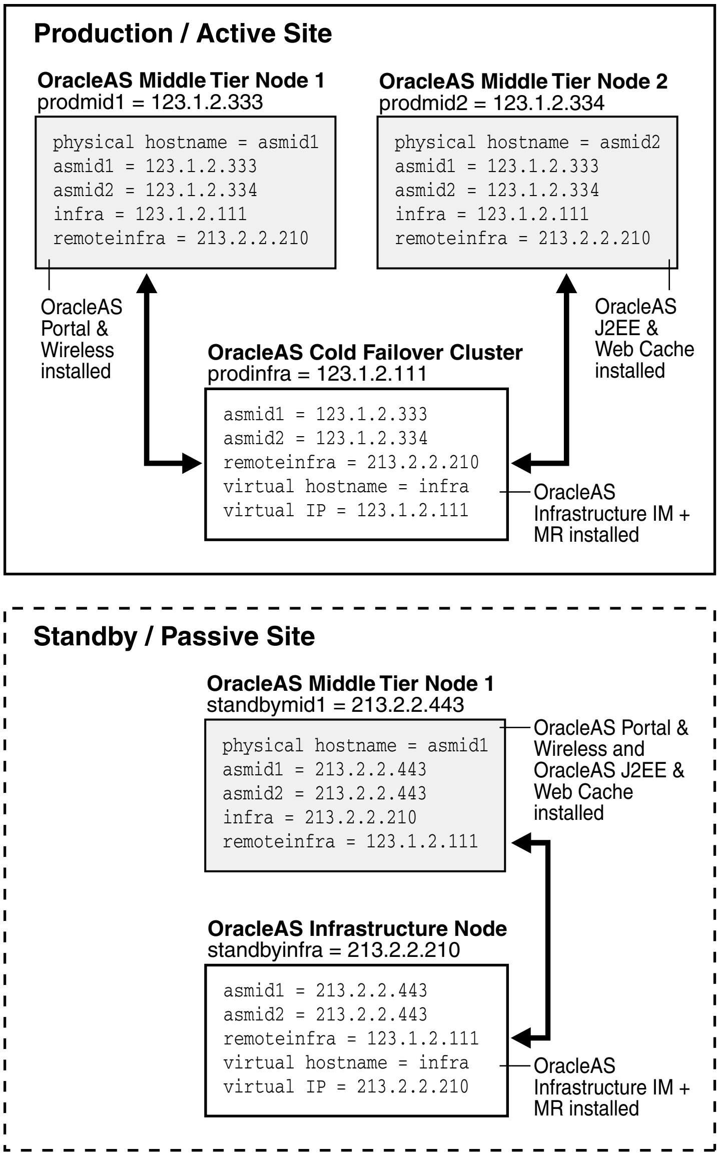
Section 1.1.3.2, "Asymmetrical Topologies - Simple Asymmetric Standby Topology with Collocated Oracle Identity Management and OracleAS Metadata Repository

Infrastructure" describes some of the asymmetrical topologies supported by OracleAS Disaster Recovery.

This section describes the system level configuration you should perform prior to installing the OracleAS software for an asymmetrical OracleAS Disaster Recovery solution where the standby site has fewer hosts than the production site. Much of the information in Section 1.3, "Preparing the OracleAS Disaster Recovery Environment for an Asymmetrical Standby Site," which is written for symmetrical topologies, is also true for asymmetrical topologies. Therefore, this section describes only the differences you need to be aware of when setting up an asymmetrical standby site topology.

Figure 1–10 shows the asymmetrical topology being described in this section and illustrates the name assignments for the production site and standby site:

***Figure 1–10 Name Assignment Example for an Asymmetrical Standby Site Topology***


Production / Active Site
OracleAS Middle Tier Node 1
prodmid1 = 123.1.2.333
physical hostname = asmid1
asmid1 = 123.1.2.333
asmid2 = 123.1.2.334
infra = 123.1.2.111
remoteinfra = 213.2.2.210
OracleAS Portal & Wireless installed
OracleAS Middle Tier Node 2
prodmid2 = 123.1.2.334
physical hostname = asmid2
asmid1 = 123.1.2.333
asmid2 = 123.1.2.334
infra = 123.1.2.111
remoteinfra = 213.2.2.210
OracleAS J2EE & Web Cache installed
OracleAS Cold Failover Cluster
prodinfra = 123.1.2.111
asmid1 = 123.1.2.333
asmid2 = 123.1.2.334
remoteinfra = 213.2.2.210
virtual hostname = infra
virtual IP = 123.1.2.111
OracleAS Infrastructure IM + MR installed
Standby / Passive Site
OracleAS Middle Tier Node 1
standbymid1 = 213.2.2.443
physical hostname = asmid1
asmid1 = 213.2.2.443
asmid2 = 213.2.2.443
infra = 213.2.2.210
remoteinfra = 123.1.2.111
OracleAS Portal & Wireless and OracleAS J2EE & Web Cache installed
OracleAS Infrastructure Node
standbyinfra = 213.2.2.210
asmid1 = 213.2.2.443
asmid2 = 213.2.2.443
remoteinfra = 123.1.2.111
virtual hostname = infra
virtual IP = 213.2.2.210
OracleAS Infrastructure IM + MR installed

In Figure 1–10, two middle-tier nodes and an OracleAS Infrastructure exist at the production site. OracleAS Portal & Wireless is installed in an Oracle home on Node 1 and OracleAS J2EE & Web Cache is installed in an Oracle home on Node 2. The OracleAS Infrastructure can be a single node or an OracleAS Cold Failover Cluster solution (represented by a single virtual hostname and a virtual IP, as for a single node OracleAS Infrastructure). At the standby site, only one middle-tier node (with collocated Application Server instances) and an OracleAS Infrastructure node exists. OracleAS Portal & Wireless is installed in an Oracle home on Node 1 and OracleAS J2EE & Web Cache is installed in another Oracle home on Node 1. The OracleAS Infrastructure is a single node at the standby site.

Table 1–10 shows the physical, network and virtual hostnames to use for the asymmetrical standby site shown in Figure 1–10:

***Table 1–10 Physical, Network, and Virtual Hostnames in Figure 1–10***

| Physical Hostnames | Virtual Hostname | Network Hostnames |
|---|---|---|
| asmid1 | - | prodmid1, standbymid1 |
| asmid2 | - | prodmid2 |
| - [1] | infra | prodinfra, standbyinfra |

[1] In this example, the physical hostname is the network hostname.

See the following sections for general information on setting up hostnames and configuring hostname resolution:

- Section 1.2.1.1, "Physical Hostnames" describes setting up physical hostnames.
- Section 1.2.1.2, "Network Hostnames" describes setting up network hostnames.
- Section 1.2.1.3, "Virtual Hostname" describes setting up virtual hostnames.
- Section 1.2.1.4, "Virtual Hostname Aliases" describes setting up virtual hostname aliases for Oracle database hosts.
- Section 1.2.2, "Configuring Hostname Resolution" describes how to use local hostnaming file resolution (`/etc/hosts` files) or DNS resolution for hostname resolution.

Before you perform any installations, follow these steps to prepare for setting up the OracleAS Disaster Recovery asymmetrical standby site topology in Figure 1–10:

> **Note:** When you do perform the installations required to set up the OracleAS Disaster Recovery topology in Figure 1–10, you must use only Oracle Universal Installer to install all the Oracle software on the hosts at the production site and standby site. Do not use the asgctl `clone instance` or `clone topology` commands when setting up this asymmetrical standby site topology. The asgctl `clone instance` and `clone topology` commands are not supported for this asymmetrical standby site topology.
>
> None of the required installations are done as part of the steps below. See Section 1.4, "Overview of Installing Oracle Application Server" for a description of the order in which the required installations for the asymmetrical topology should be performed.

1. Requirements for the OracleAS Infrastructure host at the production site:

   a. Make sure the network hostname prodinfra is set up for the host prior to installing the OracleAS Infrastructure on the host. You do this by confirming that the corporate DNS server includes the entries shown in Figure 1–11 and Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology."

- **b.** When you install the OracleAS Infrastructure on the host later, create the virtual hostname infra for the host by following the instructions in Section 1.2.1.3, "Virtual Hostname."

2. Requirements for the OracleAS Infrastructure host at the standby site:

   - **a.** Make sure the network hostname standbyinfra is set up for the host prior to installing the OracleAS Infrastructure on the host. You do this by confirming that the corporate DNS server includes the entries shown in Figure 1–11 and Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology."
   - **b.** When you install the OracleAS Infrastructure on the host later, create the virtual hostname infra for the host by following the instructions in Section 1.2.1.3, "Virtual Hostname."
   - **c.** When you install the OracleAS Infrastructure on the host later, install it into an Oracle home directory with the same path as the Oracle home directory that the OracleAS Infrastructure was installed into on the Infrastructure host at the production site. Use the `portlist.ini` file from the OracleAS Infrastructure installation at the production site as input to the OracleAS Infrastructure installation at the standby site.

3. Requirements for Node 1 at the production site:

   - **a.** Make sure the network hostname prodmid1 is set up for the host prior to installing OracleAS Portal & Wireless on the host. You do this by confirming that the corporate DNS server includes the entries shown in Figure 1–11 and Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology."
   - **b.** When you install OracleAS Portal & Wireless on the host later, make sure that the physical hostname asmid1 is set up for the host by following the instructions in Section 1.2.1.1, "Physical Hostnames."

4. Requirements for Node 2 at the production site:

   - **a.** Make sure the network hostname prodmid2 is set up for the host prior to installing OracleAS Portal & Wireless on the host. You do this by confirming that the corporate DNS server includes the entries shown in Figure 1–11 and Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology."
   - **b.** When you install OracleAS J2EE & Web Cache on the host later, make sure that the physical hostname asmid2 is set up for the host by following the instructions in Section 1.2.1.1, "Physical Hostnames."
   - **c.** Make sure that you do not install the OracleAS J2EE & Web Cache instance on Node 2 at the production site into an Oracle home directory with the same path as the Oracle home directory you installed the OracleAS Portal & Wireless instance into on Node 1 at the production site. For example, do not install the OracleAS Portal & Wireless instance into the `/app/instance1` Oracle home directory path on Node 1 at the production site and then install the OracleAS J2EE & Web Cache instance into the `/app/instance1` Oracle home directory path on Node 2 at the production site. The relative paths to the Oracle homes for these instances must be different to enable a successful failover or switchover from the production site to the standby site. The reason for this is because it is not supported for both the OracleAS Portal & Wireless instance and the OracleAS J2EE & Web Cache instance to be installed in a single Oracle home directory (the `/app/instance1` directory, for example) on Node 1 at the standby site.

d. Plan the ports for the entire topology so that the port each OracleAS instance at the standby site listens on is the same port as its peer instance listens on at the production site peer host.

5. Requirements for Node 1 at the standby site:

    a. Make sure the network hostname standbymid1 is set up for the host prior to installing either the OracleAS Portal & Wireless instance or the OracleAS J2EE & Web Cache instance on the host. You do this by confirming that the corporate DNS server includes the entries shown in Figure 1–11 and Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology."

    b. When you install the OracleAS Portal & Wireless instance on the host later, make sure that the physical hostname asmid1 is set up for the host by following the instructions in Section 1.2.1.1, "Physical Hostnames." Make sure that the OracleAS Portal & Wireless instance is installed into the same Oracle home directory as the peer instance was installed into on Node 1 at the production site. Use the `portlist.ini` file from the OracleAS Portal & Wireless installation at Node 1 at the production site as input to the OracleAS Portal & Wireless installation at Node 1 at the standby site.

    c. When you install the OracleAS J2EE & Web Cache instance on the host later, make sure that the physical hostname asmid2 is set up for the host by following the instructions in Section 1.2.1.1, "Physical Hostnames." Make sure that the OracleAS J2EE & Web Cache instance is installed into the same Oracle home directory as the peer instance was installed into on production site Node 2. Use the `portlist.ini` file from the OracleAS J2EE & Web Cache installation at Node 2 at the production site as input to the OracleAS J2EE & Web Cache installation at Node 1 at the standby site.

6. General standby site requirements:

    a. If you are using local hostnaming file resolution, confirm that both asmid1 and asmid2 are in the `/etc/hosts` files for the standby site hosts and that the entries show the 213.2.2.443 IP address.

    b. If you are using DNS hostname resolution, confirm that the entries shown in Figure 1–11 and Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology" have been made.

After performing the preparatory steps in this section, you are ready to begin the installations required for the asymmetrical topology in Figure 1–10. Perform the required installations for the topology in the order described in Section 1.4, "Overview of Installing Oracle Application Server."

After completing the installations required for the topology, use the asgctl `discover topology` command and the asgctl `instantiate topology` command to perform the remaining setup for the site.

## 1.3.1 Configuring Hostname Resolution for the Asymmetrical Topology

General information on configuring hostname resolution is provided in Section 1.2.2, "Configuring Hostname Resolution." This section describes how to configure hostname resolution for the OracleAS Disaster Recovery asymmetrical standby site topology shown in Figure 1–10.

Section 1.3.1.1, "Using Local Hostnaming File Resolution for the Asymmetrical Topology" describes how to configure hostname resolution for this asymmetrical standby site topology using local hostnaming file resolution.

Section 1.3.1.2, "Using DNS Resolution for the Asymmetrical Topology" describes how to configure hostname resolution for this asymmetrical standby site topology using DNS resolution.

#### 1.3.1.1 Using Local Hostnaming File Resolution for the Asymmetrical Topology

This method of hostname resolution relies on a local hostnaming file to contain the requisite hostname-to-IP address mappings. In UNIX, this file is `/etc/hosts`. In Windows, this file is `C:\WINDOWS\system32\drivers\etc\hosts`.

To use the local hostnaming file to resolve hostnames for the OracleAS Disaster Recovery solution in UNIX for each middle tier and OracleAS Infrastructure host in both the production and standby sites, perform the following steps:

1. Use a text editor, such as `vi`, to edit the `/etc/nsswitch.conf` file. With the "`hosts:`" parameter, specify "`files`" as the first choice for hostname resolution.
2. Edit the `/etc/hosts` file to include the following:
   - The physical hostnames and the correct IP addresses for all middle-tier nodes in the current site. The first entry must be the hostname and IP address of the current node.

     **Note:** When making entries in the hosts file, make sure the intended hostname is positioned in the second column of the hosts file; otherwise, an asgctl `verify topology with <host>` operation will fail indicating that the production topology is not symmetrical with the standby topology. See Section A.1.7, "Failure of Farm Verification Operation with Standby Farm" for more information about troubleshooting and resolving this type of problem.

     For example, if you are editing the `/etc/hosts` file of a middle-tier node in the production site, enter all the middle-tier physical hostnames and their IP addresses in the production site beginning the list with the current host. (You should also include fully qualified hostnames in addition to the abbreviated hostnames. See Table 1–11.)
   - The virtual hostname of the OracleAS Infrastructure in the current site.

     For example, if you are editing the `/etc/hosts` of a middle-tier node in the standby site, enter the virtual hostname, fully qualified and abbreviated, and the IP address of the OracleAS Infrastructure host in the standby site.
3. Restart each host after editing the files mentioned in the previous steps.
4. From each host, use the `ping` command for each physical hostname that is valid in its particular site to ensure that the IP addresses have been assigned correctly.

   For the example in Figure 1–10, on the asmid1 host at the production site, use the following commands in succession:

   ```
   ping asmid1
   ```

   The returned IP address should be 123.1.2.333.

   ```
   ping asmid2
   ```

   The returned IP address should be 123.1.2.334.

   ```
   ping infra
   ```

The returned IP address should be 123.1.2.111.

> **Note:** Some UNIX variants, such as Solaris, require the `-s` option to return an IP address.

Using the example in Figure 1–10, Table 1–11 shows that the `/etc/hosts` file entries on each production node contains the required entries for each UNIX host. The entries in the Windows `C:\WINDOWS\system32\drivers\etc\hosts` file should be similar.

***Table 1–11 Network and Virtual Hostname Entries in /etc/hosts File of Example in Figure 1–10***

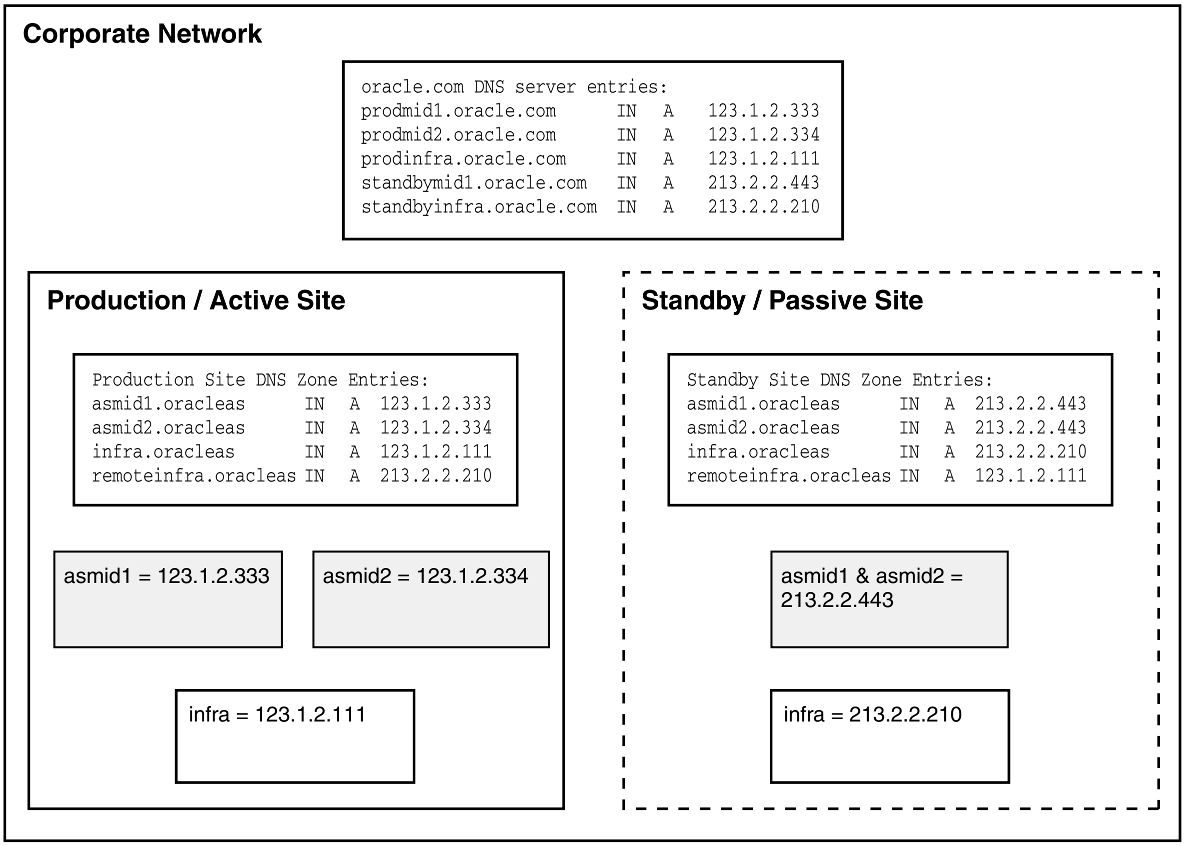
| Host | Entries in /etc/hosts |
|---|---|
| asmid1 in production site | `123.1.2.333 asmid1.oracle.com asmid1`<br>`123.1.2.334 asmid2.oracle.com asmid2`<br>`123.1.2.111 infra.oracle.com infra`<br>`213.2.2.210 remoteinfra.oracle.com remoteinfra` |
| asmid2 in production site | `123.1.2.334 asmid2.oracle.com asmid2`<br>`123.1.2.333 asmid1.oracle.com asmid1`<br>`123.1.2.111 infra.oracle.com infra`<br>`213.2.2.210 remoteinfra.oracle.com remoteinfra` |
| infra in production site | `123.1.2.111 infra.oracle.com infra`<br>`123.1.2.333 asmid1.oracle.com asmid1`<br>`123.1.2.334 asmid2.oracle.com asmid2`<br>`213.2.2.210 remoteinfra.oracle.com remoteinfra` |
| asmid1 in standby site | `213.2.2.443 asmid1.oracle.com asmid1`<br>`213.2.2.443 asmid2.oracle.com asmid2`<br>`213.2.2.210 infra.oracle.com infra`<br>`123.1.2.111 remoteinfra.oracle.com remoteinfra` |
| infra in standby site | `213.2.2.210 infra.oracle.com infra`<br>`213.2.2.443 asmid1.oracle.com asmid1`<br>`213.2.2.443 asmid2.oracle.com asmid2`<br>`123.1.2.111 remoteinfra.oracle.com remoteinfra` |

#### 1.3.1.2 Using DNS Resolution for the Asymmetrical Topology

To set up the OracleAS Disaster Recovery solution to use DNS hostname resolution, you must set up site-specific DNS servers in the production and standby sites in addition to the overall corporate DNS servers (usually more than one DNS server exists in a corporate network for redundancy). Figure 1–11 provides an overview of this setup.

> **See Also:** Chapter 6, "Setting Up a DNS Server" for instructions on how to set up a DNS server in a UNIX environment.

***Figure 1–11 DNS Resolution for Asymmetrical Topology***



```
oracle.com DNS server entries:
prodmid1.oracle.com      IN   A   123.1.2.333
prodmid2.oracle.com      IN   A   123.1.2.334
prodinfra.oracle.com     IN   A   123.1.2.111
standbymid1.oracle.com   IN   A   213.2.2.443
standbyinfra.oracle.com  IN   A   213.2.2.210
```

```
Production Site DNS Zone Entries:
asmid1.oracleas      IN   A  123.1.2.333
asmid2.oracleas      IN   A  123.1.2.334
infra.oracleas       IN   A  123.1.2.111
remoteinfra.oracleas IN   A  213.2.2.210
```

```
Standby Site DNS Zone Entries:
asmid1.oracleas      IN   A  213.2.2.443
asmid2.oracleas      IN   A  213.2.2.443
infra.oracleas       IN   A  213.2.2.210
remoteinfra.oracleas IN   A  123.1.2.111
```

For the topology in Figure 1–11 to work, the following requirements and assumptions must be made:

- The DNS servers for the production and standby sites must not be aware of each other. They make non authoritative lookup requests to the overall corporate DNS servers if they fail to resolve any hostnames within their specific sites.
- The production site and standby site DNS servers must contain entries for middle-tier physical hostnames and OracleAS Infrastructure virtual hostnames. Each DNS server contains entries of only the hostnames within its own site. The sites have a common domain name that is different from that of the overall corporate domain name.
- The overall corporate DNS servers contain network hostname entries for the middle-tier hosts and OracleAS Infrastructure hosts of both production and standby sites.
- In UNIX, the `/etc/hosts` file in each host does not contain entries for the physical, network, or virtual hostnames of any host in either the production or standby site. In Windows, this applies to the file `C:\WINDOWS\system32\drivers\etc\hosts`.

To set up the OracleAS Disaster Recovery solution for DNS resolution, follow these steps:

1. Configure each of the overall corporate DNS servers with the network hostnames of all the hosts in the production and standby sites. Using the example presented in Figure 1–10, the following entries are made:

```
prodmid1.oracle.com      IN    A    123.1.2.333
prodmid2.oracle.com      IN    A    123.1.2.334
prodinfra.oracle.com     IN    A    123.1.2.111
standbymid1.oracle.com   IN    A    213.2.2.443
standbyinfra.oracle.com  IN    A    213.2.2.210
```

2. For each site, production and standby, create a unique DNS zone by configuring a DNS server as follows:

   **a.** Select a unique domain name to use for the two sites that is different from the corporate domain name. As an example, use the name "`oracleas`" for the domain name for the two sites in Figure 1–10. The high level corporate domain name is `oracle.com`.

   **b.** Configure the DNS server in each site to point to the overall corporate DNS servers for unresolved requests.

   **c.** Populate the DNS servers in each site with the physical hostnames of each middle-tier host and the virtual hostname of each OracleAS Infrastructure host. Include the domain name selected in the previous step.

   For the example in Figure 1–10, the entries are as follows:

   For the DNS server on the production site:

   ```
   asmid1.oracleas      IN    A    123.1.2.333
   asmid2.oracleas      IN    A    123.1.2.334
   infra.oracleas       IN    A    123.1.2.111
   ```

   For the DNS server on the standby site:

   ```
   asmid1.oracleas      IN    A    213.2.2.443
   asmid2.oracleas      IN    A    213.2.2.443
   infra.oracleas       IN    A    213.2.2.210
   ```

**Note:** If you are using a load balancer, you must alias the IP addresses inside the host file with DNS based virtual hostnames. This is essential for Oracle Application Server Guard on the local host to perform a local write of the topology file from a discover topology within farm command and to correctly perform an add instance command and update the topology file.

If you are using the OracleAS Cold Failover Cluster solution for the OracleAS Infrastructure in either site, enter the cluster's virtual hostname and virtual IP address. For example, in the previous step, infra is the virtual hostname and `123.1.2.111` is the virtual IP of the cluster in the production site. For more information on the OracleAS Cold Failover Cluster solution, see Section 6.2.2, "Active-Passive High Availability Topologies" on page 6-5 of the *Oracle Application Server High Availability Guide* in the Oracle Application Server Release 10.1.2.0.2 documentation set.

Also, for this asymmetrical topology, follow the instructions in Section 1.2.2.2.1, "Additional DNS Server Entries for Oracle Data Guard."

## 1.4 Overview of Installing Oracle Application Server

This section provides an overview of the steps for installing the OracleAS Disaster Recovery solution. These steps are applicable to the topologies described in Section 1.1.3, "Supported Topologies". After following the instructions in Section 1.2, "Preparing the OracleAS Disaster Recovery Environment" to set up the environment for the solution, read this section for an overview of the installation process. Then, follow the detailed instructions in the *Oracle Application Server Installation Guide* to install the solution.

> **Note:** To assign identical ports for use by symmetrical hosts in the production and standby sites, you can use static port definitions. These definitions are defined in a file, (for example, named `staticports.ini`). Then, specify the `staticports.ini` file in the **Specify Ports Configuration Options** screen in the installer. (Detailed information on the static ports file is found in the *Oracle Application Server Installation Guide*.)

The following steps represent the overall sequence for installing the OracleAS Disaster Recovery solution:

1. Install OracleAS Infrastructure in the production site (see *Oracle Application Server Installation Guide*).
2. Install OracleAS Infrastructure in the standby site (see *Oracle Application Server Installation Guide*).
3. Start the OracleAS Infrastructure in each site before installing the middle tiers for that site.
4. Install the middle tiers in the production site (see *Oracle Application Server Installation Guide*).
5. Install the middle tiers in the standby site (see *Oracle Application Server Installation Guide*).

The following points are important when you perform the installation:

- Ensure that the same ports are used by equivalent peer hosts in both sites. For example, the asmid1 host in the standby site must use the same ports as the asmid1 host in the production site. Use a static ports definition file. (see the previous note in this section and the following point).
- Specify the full path to the `staticports.ini` file in the installer's **Specify Ports Configuration Options** screen.
- During the Infrastructure installation, ensure that you select the High Availability and Replication option in the installer's **Select Configuration Options** screen.
- Specify the virtual address assigned to the OracleAS Infrastructure in the **Specify Virtual Hostname** screen during OracleAS Infrastructure installation.
- Install for the middle-tier hosts, any of the available middle-tier installation types. (Ensure that the OracleAS Infrastructure services have been started for a site before installing any middle tiers in that site.)
- Specify the OracleAS Infrastructure's virtual hostname as the OracleAS Infrastructure database during each middle-tier installation.
- Start the OracleAS services on the hosts in each site starting with the OracleAS Infrastructure.

## 1.5 Wide Area DNS Operations

To direct client requests to the entry point of a production site, use DNS resolution. When a site switchover or failover is performed, client requests must be redirected transparently to the new site that is playing the production role. To accomplish this redirection, the wide area DNS that resolves requests to the production site has to be switched over to the standby site. The DNS switchover can be accomplished by either using a wide area load balancer or manually changing DNS names.

**Note:** A hardware load balancer is assumed to be front-ending each site. Check for supported load balancers at:

`https://metalink.oracle.com`

The following subsections describe the DNS switchover operation.

### 1.5.1 Using a Wide Area Load Balancer

When a wide area load balancer (global traffic manager) is deployed in front of the production and standby sites, it provides fault detection services and performance-based routing redirection for the two sites. Additionally, the load balancer can provide authoritative DNS name server equivalent capabilities.

During normal operations, the wide area load balancer can be configured with the production site's load balancer name-to-IP mapping. When a DNS switchover is required, this mapping in the wide area load balancer is changed to map to the standby site's load balancer IP. This allows requests to be directed to the standby site, which now has the production role.

This method of DNS switchover works for both site switchover and failover. One advantage of using a wide area load balancer is that the time for a new name-to-IP mapping to take effect can be almost immediate. The downside is that an additional investment must be made for the wide area load balancer.

### 1.5.2 Manually Changing DNS Names

This method of DNS switchover involves the manual change of the name-to-IP mapping that is originally mapped to the IP address of the production site's load balancer. The mapping is changed to map to the IP address of the standby site's load balancer. Follow these instructions to perform the switchover:

1. Make a note the current time-to-live (TTL) value of the production site's load balancer mapping. This mapping is in the DNS cache and it will remain there until the TTL expires. As an example, let's assume that the TTL is 3600 seconds.
2. Modify the TTL value to a short interval (for example, 60 seconds).
3. Wait one interval of the original TTL. This is the original TTL of 3600 seconds from Step 1.
4. Ensure that the standby site is switched over to receive requests.
5. Modify the DNS mapping to resolve to the standby site's load balancer giving it the appropriate TTL value for normal operation (for example, 3600 seconds).

This method of DNS switchover works for planned site switchover operations only. The TTL value set in Step 2 should be a reasonable time period where client requests cannot be fulfilled. The modification of the TTL is effectively modifying the caching semantics of the address resolution from a long period of time to a short period. Due to the shortened caching period, an increase in DNS requests can be observed.

### 1.5.3 HTTP Server Configuration When Using a Server Load Balancer

If you are using a Server Load Balancer to direct HTTP requests to multiple Oracle HTTP Server instances, Web access to some applications (such as the Application Server Control console and Oracle Web Services Manager) may be redirected to the physical HTTP Server hosts.

To ensure that redirected requests are always sent to the load balancer, configure an Oracle HTTP Server virtual host for the load balancer.

For example, if Oracle HTTP Server is listening on port 7777 and a load balancer called `bigip.acme.com` is listening on port 80, then consider the following entry in the `httpd.conf` file:

```
NameVirtualHost *:7777
<VirtualHost *:7777>
ServerName bigip.us.oracle.com
Port 80
ServerAdmin youyour.address
RewriteEngine On
RewriteOptions inherit
</VirtualHost>
```

# 2

# Oracle Application Server Guard and asgctl

This chapter describes how to use Oracle Application Server Guard and the asgctl command-line utility to perform OracleAS Disaster Recovery tasks. It includes the following sections:

- Section 2.1, "Overview of Oracle Application Server Guard and asgctl"
- Section 2.2, "Authentication of Databases"
- Section 2.3, "Discovering, Dumping, and Verifying the Topology"
- Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands"
- Section 2.5, "Discovering Oracle Application Server 10.1.3 Instances in Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology"
- Section 2.6, "Adding or Removing Oracle Application Server 10.1.3 Instances to Redundant Single Oracle Application Server 10.1.3 Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology"
- Section 2.7, "Oracle Application Server Guard Operations -- Standby Site Cloning of One or More Production Instances to a Standby System"
- Section 2.8, "Oracle Application Server Guard Operations -- Standby Instantiation and Standby Synchronization"
- Section 2.9, "Runtime Operations -- Oracle Application Server Guard Switchover and Failover Operations"
- Section 2.10, "Monitoring Oracle Application Server Guard Operations and Troubleshooting"
- Section 2.11, "Using Oracle Application Server Guard Command-Line Utility (asgctl)"

## 2.1 Overview of Oracle Application Server Guard and asgctl

This section provides an overview of Oracle Application Server Guard and its command-line interface asgctl. If you are already familiar with this overview information, go to Section 2.2, "Authentication of Databases". This section contains the following subsections:

- Section 2.1.1, "Overview of asgctl"
- Section 2.1.2, "Oracle Application Server Guard Client"
- Section 2.1.3, "Oracle Application Server Guard Server"
- Section 2.1.4, "asgctl Operations"

- Section 2.1.5, "Oracle Application Server Guard Integration with OPMN"
- Section 2.1.6, "Supported OracleAS Disaster Recovery Configurations"
- Section 2.1.7, "Configuring Oracle Application Server Guard and Other Relevant Information"

### 2.1.1 Overview of asgctl

The asgctl command-line utility greatly simplifies the complexity and magnitude of the steps involved in setting up and managing OracleAS Disaster Recovery. This utility provides a distributed solution that consists of a client component and a server component. The client component (Oracle Application Server Guard client) can be installed on a system on the topology. The server component (Oracle Application Server Guard server) is installed by default on the systems hosting the primary and standby Oracle homes that comprise the OracleAS Disaster Recovery environment.

### 2.1.2 Oracle Application Server Guard Client

The Oracle Application Server Guard client is installed on every Oracle Application Server install type. The Oracle Application Server Guard client attempts to open and maintain a connection to the Oracle Application Server Guard server.

The Oracle Application Server Guard client provides an asgctl command-line interface (CLI) (see Chapter 5, "Oracle Application Server Guard asgctl Command-line Reference") consisting of a set of commands to perform administrative tasks described in Section 2.1.4, "asgctl Operations".

### 2.1.3 Oracle Application Server Guard Server

The Oracle Application Server Guard server is a distributed server (installed by default) that runs on all the systems in an OracleAS Disaster Recovery configuration. The Oracle Application Server Guard client maintains an active connection to the Oracle Application Server Guard server on one system that has network connectivity in the OracleAS Disaster Recovery configuration. This coordinating server communicates to the Oracle Application Server Guard servers on the other systems in the OracleAS Disaster Recovery configuration as necessary to complete processing during standby site cloning, instantiation, synchronization, verification, switchover, and failover operations. The Oracle Application Server Guard server carries out asgctl commands issued directly by the Oracle Application Server Guard client or issued on behalf of the Oracle Application Server Guard client by another Oracle Application Server Guard server in the network for the client session. The steps to complete an operation will execute throughout all systems in both the production and standby topologies. Most operational steps will be executed either in parallel or sequentially (as required) on these systems throughout the OracleAS Disaster Recovery configuration by the Oracle Application Server Guard server.

### 2.1.4 asgctl Operations

Major asgctl operations using the asgctl commands belong in the following categories of operations:

- Authentication -- Identifies the OracleAS Infrastructure database on the primary topology (set primary database command). If there are topologies with multiple Infrastructures, each must be identified using this command prior to performing an operation involving both production and standby topologies. Use the connect asg command first to connect to an ASG server, then issue the set primary database command multiple times until you identify each Infrastructure database.

Use the connect asg command first to connect to an ASG server and then identify the new OracleAS Infrastructure database on the standby topology using the set new primary database command before performing a failover operation.

Sets the credentials (set asg credentials command) used to authenticate the Oracle Application Server Guard client connections to Oracle Application Server Guard servers and the connections between Oracle Application Server Guard servers to a specific host. See the set asg credentials command for an example, and see Section 4.1.1.1, "Setting asgctl Credentials" for more information.

When Oracle Application Server Guard discovers the topology (discover topology command), it requires that you provide Oracle Internet Directory authentication credentials (Oracle Internet Directory password) in order to query Oracle Internet Directory to obtain instance information for the production site.

- Discover the topology -- Discovers (discover topology command) by querying Oracle Internet Directory all instances within the topology that share the same Oracle Internet Directory for a production site and generates a topology XML file (`topology.xml`) that describes the topology and replicates this file to all instances in the topology. See Section 2.3, "Discovering, Dumping, and Verifying the Topology" for more information.

  The command discover topology within farm discovers the topology using OPMN at a production site for special cases where Oracle Internet Directory is not available, such as in an Oracle Application Server 10.1.3 OPMN topology.

- Adding or removing an instance from the local topology file -- pulls an instance into or drops an instance from an existing local topology file. This is particularly useful in an Oracle Application Server 10.1.3 only topology, where an Oracle Application Server Guard client can connect to an existing Oracle Application Server 10.1.3 instance, and either perform an add instance command that adds the specified Oracle Application Server 10.1.3 instance to the topology file, or performs a remove instance command that removes the specified Oracle Application Server 10.1.3 instance from the local topology file, and in either case, if specified (by using the `to topology` parameter for the `add instance` command or the `from topology` parameter for the `remove instance` command), propagates the updated topology file to all instances in the Disaster Recovery production environment. Any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby environment.

  This command is also useful for adding an Oracle Application Server 10.1.3 J2EE instance to an Oracle Internet Directory (OID) based 10.1.2.0.2 local topology file to support a mixed version Disaster Recovery environment. For example, you can use the add instance command to add an Oracle Application Server 10.1.3 J2EE instance to your OID based 10.1.2.0.2 local topology file. See Section 2.6, "Adding or Removing Oracle Application Server 10.1.3 Instances to Redundant Single Oracle Application Server 10.1.3 Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology" for a use case.

- Standby site cloning -- Copies a single Application Server instance on a production site host to a standby site host (clone instance command) or copies two or more Application Server instances on production site hosts to standby site hosts (clone topology command). See Section 2.7, "Oracle Application Server Guard Operations -- Standby Site Cloning of One or More Production Instances to a Standby System" for more information. The standby site cloning commands provide an alternative to installing Application Server instances on standby site hosts using Oracle Universal Installer.

- Standby site instantiation -- Creates the disaster recovery environment. It establishes the relationship between standby and production instances, mirrors the configuration, then synchronizes the standby site with the primary site (instantiate topology command). See Section 2.8.1, "Standby Instantiation" for more information.
- Standby site synchronization -- Applies database redo logs for OracleAS Infrastructures to the standby site in conjunction with synchronizing external configuration files across the topology (sync topology command). See Section 2.8.2, "Standby Synchronization" for more information.
- Switchover -- Switches from the production site to the standby site after the standby site is synchronized with the production site with the application of the database redo logs (switchover topology command). See Section 2.9.1.1, "Scheduled Outages" for more information.
- Failover -- Makes the standby site the production site after restoring configuration files and restoring the Oracle Application Server server environment to the point of the last successful sync operation (failover command). See Section 2.9.1.2, "Unplanned Outages" for more information.
- Verification -- Validates that the primary topology is running and the configuration is valid (verify topology command) or if a standby topology is specified, compares the primary topology to which the local host system is a member with the standby topology to validate that they are consistent with one another and conform to the requirements for OracleAS Disaster Recovery. See Section 2.10.1, "Verifying the Topology" for more information.
- Using a policy file -- Used as a filter to filter out unnecessary instances for supporting asymmetric topologies. The dump policies command writes detailed, default policy information to respective XML formatted files for a select set of asgctl commands. You can then edit each respective XML policy file and use it in the `using policy <file>` parameter with any one of these select set of asgctl commands: dump topology, verify topology, clone topology, failover, instantiate topology, switchover topology, and sync topology to define by instance the domain of execution operations that are permitted for each of these asgctl commands. Each instance list entry in an XML policy file logically tags a production-standby peer combination with a particular attribute that defines the success requirement for its successful operation. For example, you may want to omit a node in a symmetric topology while performing one of the operations previously mentioned. Use the policy file to specify the node to be ignored. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.
- Instance management -- Enables you to shut down (shutdown topology command) and start up the topology (startup topology command).
- Troubleshooting -- Uses the dump topology command to write detailed information about the topology to the screen or to a file. Lets you determine the current operations that are running (show operation command) and stop any operations that must be halted (stop operation command). Uses the set trace command to log output about operations to the Oracle Application Server Guard log files.

Table 2–1 describes the OracleAS Disaster Recovery production and standby site environment before and after performing an asgctl clone, instantiate, sync, failover, and switchover operation.

*Table 2–1 Description of Disaster Recovery Production and Standby Environments Before and After Performing These Oracle Application Server Guard Operations*

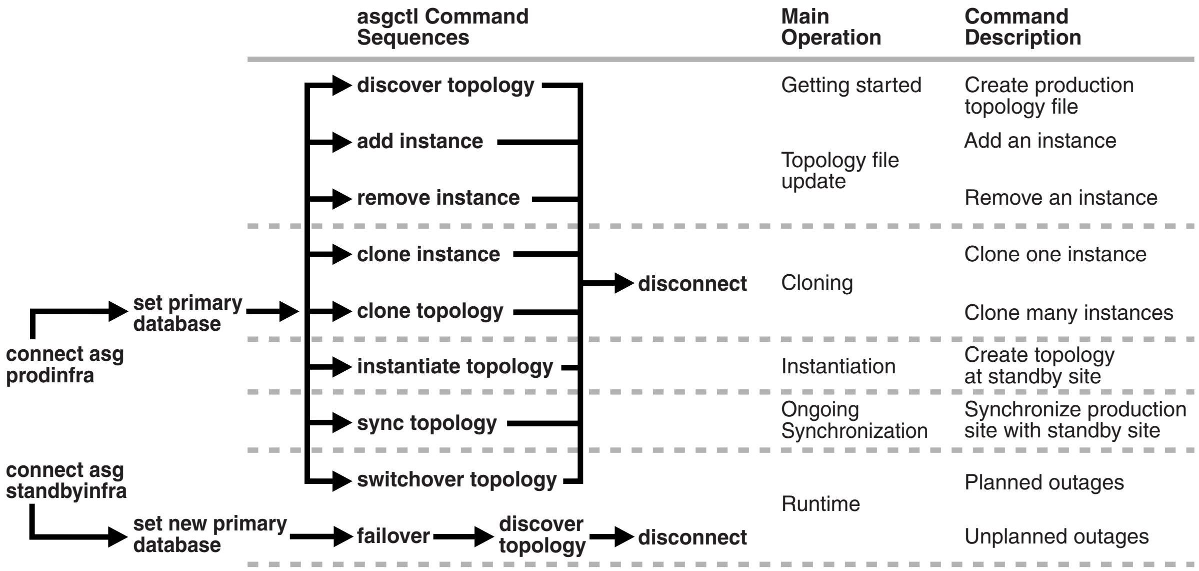
| OracleAS Guard Operation | Disaster Recovery Site Environment Before Operation | Disaster Recovery Site Environment After Operation |
|---|---|---|
| clone | The production site has one or more instances that must be installed on the standby site and instantiated. The cloning operations perform this task. | The standby site has one or more new standby instances that are a logical mirror of the production site instances. |
| instantiate | The standby site with its Oracle homes exists, but the OracleAS Disaster Recovery relationship across sites does not exist yet for an OracleAS Disaster Recovery operation to be performed. | A logical mirror of the production site is set up and maintained at the standby site. |
| sync | The standby site is not consistent with the production site. OracleAS Disaster Recovery is not able to restore the standby site to a consistent point in time without some manual intervention. | Database redo logs are applied to OracleAS Infrastructures in combination with synchronizing external configuration files across the topology. The sync operation is performed in the event that a failover or switchover operation is necessary, then the standby site can be restored to a consistent point in time. No manual intervention is necessary to synchronize the sites after the asgctl sync operation is performed. |
| switchover | A planned outage at the production site will make the standby site the production site for a period of time; that is, the roles of each site will be switched. | The standby site has become the production site. All OPMN services are started. The production site may become available again after the planned outage, at which time, another switchover operation could be performed to return activity back to the original production site from the standby site. |
| failover | An unscheduled outage at the production site has left the production site down or unavailable for an unknown period of time. The production site is lost due to some unforeseen circumstance. | The standby site has permanently become the production site. Configuration and Infrastructure data are restored to a consistent point in time on the standby site. Site services are brought up in a consistent fashion to the point of the last sync operation. All OPMN services are started. |

Figure 2–1 shows a summary of the main OracleAS Disaster Recovery operations and the command sequences used to perform these operations. For example, to get started with a new Disaster Recovery environment once it is set up and operational, you must always create a topology file on the production site. To do that, you perform a `connect asg` command to connect the Oracle Application Server Guard client to the Oracle Application Server Guard server followed by identifying the production Infrastructure database using a `set primary database` command, then perform the `discover topology` command, finally followed by a `disconnect` command to disconnect the Oracle Application Server client from the Oracle Application Server server. In essence, to perform any OracleAS Disaster Recovery operation using asgctl commands, you must always connect the Oracle Application Server Guard client to the Oracle Application Server Guard server, identify the location of the Infrastructure database, then perform the OracleAS Disaster Recovery operation or operations of interest, and finally disconnect the Oracle Application Server client from the Oracle Application Server server. The only exception to this general sequence is for a failover operation, in which the production site is permanently unavailable due to some unplanned problem. In this case, you connect the Oracle Application Server Guard

client to Oracle Application Server Guard server at the standby site, use the `set new primary database` command to identify the standby site Infrastructure as the new Infrastructure database, then perform the failover operation. Because there is no topology file created on the standby site following a failover operation, you must then perform a `discover topology` command to create it for the first time. Then you can disconnect the Oracle Application Server Guard client from Oracle Application Server Guard server.

The asgctl command sequences shown in Figure 2–1 assume the simplest topology configuration. For example, in a more complex case, if your production or standby site has multiple Metadata Repository instances installed, you must identify each instance with a `set primary database` command prior to performing the main Disaster Recovery operation, such as an instantiate, sync, or switchover operation. Similarly, Oracle Application Server Guard requires that you set the credentials for any Oracle Application Server Guard server system in the topology that has different credentials from the Oracle Application Server Guard server to which you are connected before you perform any of these main operations, such as instantiate, sync, or switchover. So for both of these cases, additional asgctl commands are required in the command sequence before you can perform the main Disaster Recovery operation. See the usage notes for the set asg credentials command for more information.

***Figure 2–1 Main Disaster Recovery Operations Performed Using the Following Oracle Application Server Guard asgctl Command Sequences***



## 2.1.5 Oracle Application Server Guard Integration with OPMN

A typical Oracle Application Server site has multiple farms. Oracle Application Server Guard server and its ias-component ASG process (or ias-component DSA process) is not started by default by OPMN because it is only necessary in the context of disaster recovery sites. You must start this ias-component ASG or ias-component DSA process in all Oracle homes as described later in this section. To check the status of this component and determine if the component is running, run the following opmnctl command on each system in your topology:

```
On UNIX systems
> <ORACLE HOME>/opmn/bin/opmnctl status
```

```
On Windows systems
> <ORACLE HOME>\opmn\bin\opmnctl status
```

Because there is no way an Oracle Application Server Guard client nor OPMN on the production site can start Oracle Application Server Guard services on the standby site, Oracle Application Server Guard must be started directly using opmnctl on the Infrastructure node in the standby topology. Connect to a node and run the appropriate OPMN command for your release on UNIX systems:

```
For 10.1.2.x and 10.1.4.x releases, this command starts OracleAS Guard on UNIX:
> <ORACLE_HOME>/opmn/bin/opmnctl startproc ias-component=DSA

For 10.1.3.x releases, this command starts OracleAS Guard on UNIX:
> <ORACLE_HOME>/opmn/bin/opmnctl startproc ias-component=ASG
```

On Windows systems, issue the appropriate OPMN command for your release to start Oracle Application Server Guard if your Oracle home is located on drive C:.

```
For 10.1.2.x and 10.1.4.x releases, this command starts OracleAS Guard on Windows:
C:\<ORACLE HOME>\opmn\bin\opmnctl startproc ias-component=DSA

For 10.1.3.x releases, this command starts OracleAS Guard on Windows:
C:\<ORACLE HOME>\opmn\bin\opmnctl startproc ias-component=ASG
```

After the Oracle Application Server Guard server is started it is non transient, while the remaining Oracle Application Server Guard servers in the standby topology are transient servers. This configuration allows cross-topology communication.

**Note:** When you perform an opmnctl status command on a system on which Oracle Application Server Guard is running, you will see an ias-component and process-type named DSA for 10.1.2.x and 10.1.4.x releases (for 10.1.3.x releases, the ias-component and process-type is named ASG). This is the Oracle Application Server component name and server process name for the Oracle Application Server Guard server.

### 2.1.6 Supported OracleAS Disaster Recovery Configurations

For Oracle Application Server 10g release (10.1.2), Oracle Application Server Guard supports not only the default OracleAS Infrastructure configuration supported on Oracle Application Server Cold Failover Clusters and single instance, but also the topologies described in Section 1.1.3, "Supported Topologies."

### 2.1.7 Configuring Oracle Application Server Guard and Other Relevant Information

By default, Oracle Application Server Guard and asgctl, the command-line utility for Oracle Application Server Guard, are installed for every Application Server installation type that installs a JDK or JRE environment. For any Application Server installation type that does not install a JDK or JRE environment, install the standalone Oracle Application Server Guard kit into the Application Server home. The Oracle Application Server Guard standalone kit is downloadable from Oracle Technology Network at:

http://www.oracle.com/technology/index.html

Oracle Application Server Guard and asgctl are installed with the following default configuration information:

- The following Oracle Application Server Guard parameters are configurable in the `dsa.conf` file (for 10.1.2.x and 10.1.4.x releases) or `asg.conf` file (for 10.1.3.x releases). The value is described and the default value is indicated. The Oracle Application Server Guard `readme.txt` file in the `<ORACLE_HOME>\dsa\doc` directory also lists these Oracle Application Server Guard parameters that are configurable.
  - `port` - the TCP/IP port for Oracle Application Server Guard server and client. Oracle Application Server Guard uses a default port (port) number of 7890; for example, port=7890. If there is a second Oracle home installed on a system, this second Oracle home must have a different Oracle Application Server Guard port number, usually incremented by one, for example, port=7891, and so on.

    Value: integer, any valid TCP/IP port number. Default is 7890.

    Use the same port numbers for ASG on primary site and standby site(s) Oracle homes in Disaster Recovery configurations to prevent cloning problems.
  - `exec_timeout_secs` - timeout value for executing operating system command.

    Value: integer, number of seconds. Default is 60 seconds.
  - `trace_flags` - trace flags to be turned on.

    Value: string list, separated by ",". Default is none.
  - `backup_mode` - indicates whether to perform a full or incremental backup.

    Value: string, "full" or "incremental". Default is "incremental".
  - `backup_path` - the backup directory path to be used by Oracle Application Server Guard server.

    Value: string, a directory path. Default is `<ORACLE_HOME>/dsa/backup`.
  - `ha_path` - the High Availability directory path where the backup scripts are located.

    Value: string, a directory path. Default is `<ORACLE_HOME>/backup_restore`.
  - `port.<host>` - the TCP/IP port for a given host.

    Value: integer, any valid TCP/IP port number.

---

**Note:** If the port number must be changed for some reason (it must be unique for each Oracle Application Server Guard server in each Oracle home on a system, which is automatically handled during installation), you can change its value in the `<ORACLE_HOME>/dsa/dsa.conf` file (for 10.1.2.x and 10.1.4.x releases) or `<ORACLE_HOME>/dsa/asg.conf` file (for 10.1.3.x releases). Then, stop the Oracle Application Server Guard server (using `<ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=DSA` for 10.1.2.x and 10.1.4.x releases and `<ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=ASG`) for 10.1.3.x releases) and start the Oracle Application Server Guard server (using `<ORACLE_HOME>/opmn/bin/opmnctl startproc ias-component=DSA` for 10.1.2.x and 10.1.4.x releases and `<ORACLE_HOME>/opmn/bin/opmnctl startproc ias-component=ASG` for 10.1.3.x releases) to activate the change. See Section 2.1.5, "Oracle Application Server Guard Integration with OPMN") for more information.

Within a topology, if the port number must be changed, use the asgctl `shutdown` command to stop the Oracle Application Server Guard server and then use the asgctl `start` command to start the Oracle Application Server Guard server. These shutdown and start operations must be performed in each Oracle home that is part of the Oracle Application Server Guard topology.

---

– `clone_unpack_cmd` - specifies the tar command for Application Server Guard to use when unpacking the jar file created during a clone instance or clone topology operation.

  Value: string, tar command to use when unpacking a jar file created during a cloning operation. For example:

  ```
  clone_unpack_cmd = tar -c /xpf
  ```

– `copyfile_buffersize` - the buffer size for copy file operation, in kilobytes.

  Value: integer, maximum buffer size is 500K.

– `server_inactive_timeout` - the number of seconds server will wait before shutting down due to inactivity.

  Value: integer, number of seconds. Default value is 600 seconds (10 minutes).

– `inventory_location` - the alternative Oracle Inventory location

  Value: string, the full path of the location of `oraInst.loc` file.

– `_realm_override` - overrides the dsa realm. This is an Oracle internal parameter that is valid only for Application Server Guard releases 10.1.2.2 and later. The `realm_override` parameter is set by default by the installer for the Application Server Guard 10.1.2.x releases. It is not set for Application Server Guard 10.1.3.x releases.

  Value: integer, 1=override, 0=no override

– `dufhosts_location` - specifies the location of a file to use for resolving hostnames instead of using the normal hostname resolution described in Section 1.2, "Preparing the OracleAS Disaster Recovery Environment." This parameter is available only in Application Server release 10.1.2.3.

Value: string, full path to the file to use for resolving hostnames. For example:

dufhosts_location = <ORACLE_HOME>/dsa/conf/dufhosts.txt

You can create a file named `dufhosts.txt` in the default location, which is the <ORACLE_HOME>/dsa/conf directory for Application Server Guard. The entries in this file use the same syntax as `/etc/hosts` files. If you create this file with the name dufhosts.txt in the default location, then it will be used for hostname resolution even if you do not use the dufhosts_location parameter in the `dsa.conf` file.

You can also use a different name or location for the file. In this case, you must use the `dufhosts_location` parameter in the `dsa.conf` file to specify the path to the file in order for the file to be used for hostname resolution.

- The Oracle Application Server Guard command-line utility asgctl is installed in the `<ORACLE_HOME>/dsa/bin` directory on UNIX systems and `<ORACLE_HOME>\dsa\bin` directory on Windows systems.
- See the Oracle Application Server Guard `readme.txt` file in the `<ORACLE_HOME>\dsa\doc` directory for more information about Oracle Application Server Guard parameters that are configurable.
- Oracle Application Server Guard starts up the Oracle Application Server component services across the production topology.
- The Oracle Application Server Guard operation status information for a topology (from either an asgctl `show operation full` or `show operation history` command) remains available for the life of the current Oracle Application Server Guard client asgctl connect session only. When the Oracle Application Server Guard client disconnects from the Oracle Application Server Guard server, this topology's operation history information becomes unavailable.
- After you start an asgctl operation, you cannot run another asgctl command on the same Oracle Application Server Guard server until the previous command that is running completes or is forced to stop (see the asgctl stop operation command for more information.) In addition, you cannot run an asgctl operation in the background and then quit or exit the asgctl utility.
- When you are running Oracle Application Server Guard in an Oracle RAC environment, be sure to have only one Oracle RAC instance running while performing Oracle Application Server Guard operations.

## 2.2 Authentication of Databases

Several levels of authentication are required when an Oracle Application Server Guard client connects to an Oracle Application Server Guard server and begins a session to perform administrative operations within the production topology or across both production and standby topologies:

- Infrastructure authentication
- Oracle Application Server Guard client authentication to Oracle Application Server Guard servers
- Oracle Internet Directory authentication

### Infrastructure Authentication

When initiating an Oracle Application Server Guard administrative session, after establishing the connection between the Oracle Application Server Guard client and Oracle Application Server Guard server, you must identify all the OracleAS

Infrastructure databases on the primary topology using the set primary database command. Infrastructure authentication must be performed before you initiate any operation that involves either the production topology or both the production and standby topologies. Use the connect asg command first to connect to an ASG server, then issue the set primary database command multiple times until you identify each Infrastructure database.

Another form of Infrastructure authentication occurs as part of a failover operation. In this scenario, the production site is down and you must failover to the standby site and make this site the new production site. Use the connect asg command first to connect to an ASG server and then identify the new OracleAS Infrastructure database on the standby topology using the set new primary database command before performing the failover operation. See Section 2.9.1.2, "Unplanned Outages" for more information.

### Oracle Application Server Guard Client Authentication to Oracle Application Server Guard Servers

By default, these are the same authentication credentials used for instance level authentication with the Oracle Application Server account (`ias_admin`/password) that was created during the Oracle Application Server installation and used in the connect asg command.

> **Note:** If this is an Oracle Application Server 10.1.3 installation, the user name must be `oc4jadmin` and the password for the `oc4jadmin` account created during the Oracle Application Server 10.1.3 installation.

These same credentials are used when the Oracle Application Server Guard client connects to any Oracle Application Server Guard server in the production and standby topology when executing administrative operations.

There may be cases where you want to use different credentials for a specific Oracle Application Server Guard server or set a common set of credentials in the standby topology that differs from the credentials used in the primary topology. To set credentials for an Oracle Application Server Guard server, use the set asg credentials command and one or more of its parameter options by either specifying the host name to which the credentials apply or the topology along with the new set of credentials (username/password).

If you set the credentials for a topology, these credentials are inherited for the entire topology. If you set the credentials for an individual host on the topology, the credentials for this host override the default credentials set for the topology. After you set the credentials so that they are different from the default connection credentials for a host system or an entire topology, whenever you initiate an Oracle Application Server Guard administrative session, you must specify all credentials that are different from the default connection credentials for any host system or topology before you perform an operation involving all the Oracle Application Server Guard servers within a production topology or across both production and standby topologies. Otherwise, the operation will fail with an authentication error. See the connect asg command for an example.

### Oracle Internet Directory Authentication

The `discover topology` command requires you provide Oracle Internet Directory authentication credentials (Oracle Internet Directory password) in order to query

Oracle Internet Directory to obtain instance information for the production site. See the section that follows for more information and the discover topology command.

## 2.3 Discovering, Dumping, and Verifying the Topology

The `discover topology` command discovers by querying Oracle Internet Directory all instances within the topology that share the same Oracle Internet Directory for a production site. A topology XML file is created and distributed to all Oracle homes within the topology that describes all instances for the topology. This topology file is used by all Oracle Application Server Guard operations.

You must perform a `discover topology` command when you first set up your OracleAS Disaster Recovery environment in order to initially create the topology XML file. Thereafter, you should perform a `discover topology` command whenever you procure another Oracle home in a production site or change roles from a production to a standby site through a switchover or failover operation. See the discover topology command for more information.

You should perform a `dump topology` command to inspect the information that describes your topology. See the dump topology command for more information. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

You should perform a `verify topology` command to validate that the primary topology is running and that the configuration is valid. In addition, if you specify the `with host` parameter, the verify operation compares the primary topology of which the local host system is a member with the standby topology to validate that they are consistent with one another and conform to the requirements for OracleAS Disaster Recovery. See Section 2.10.1, "Verifying the Topology" and the verify topology command for more information.

With both the `dump topology` and `verify topology` commands, if you want to use a policy file, edit and use the respective dump and verify policy files (`dump_policy.xml` and `verify_policy.xml`). Specify this file in the `using policy <file>` parameter of each command to dump or verify only those instances specified accordingly. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.

## 2.4 Dumping Policy Files and Using Policy Files With Some asgctl Commands

OracleAS Disaster Recovery provides support for a variety of Application Server topologies as described in Section 1.1.3, "Supported Topologies." As part of this support, a set of XML formatted policy files are maintained, local to the Oracle Application Server Guard client that performs the `dump policies` command, to record by instance the domain of execution operations that are permitted for each of the following asgctl commands: dump topology, verify topology, clone topology, failover, instantiate topology, switchover topology, and sync topology.

To understand the default policies in use for any these asgctl commands, enter the following command at the asgctl prompt:

```
ASGCTL> dump policies
Generating default policy for this operation
Creating policy files on local host in directory
"/private1/OraHome2/asr1012/dsa/conf/"
ASGCTL>
```

Each instance list entry in each of the XML policy files logically tags by default a production-standby peer combination with a particular attribute that defines the success requirement for the successful operation of each command. This approach provides greater flexibility in regulating how each of these Oracle Application Server Guard operations are to be successfully used among the supported topologies. See Section 1.1.3, "Supported Topologies" for more information.

After inspecting each of the XML formatted policy files, you can edit a policy file and then use the `using policy <file>` parameter with a particular asgctl command to use that policy fiile with the command. In this way, you can employ a particular Disaster Recovery policy that defines the success requirement attribute value by instance for each of these Oracle Application Server Guard operations mentioned earlier in this chapter.

**Note:** If you want to maintain a set of custom policy files, you must copy, edit, and maintain them in a location other than the default location; otherwise, your custom set of policy files will be overwritten whenever you perform a `discover topology` command followed subsequently by a `dump policies` command.

The success requirement attribute value can be one of the following: [`optional | mandatory | ignore | group <MinSucceeded=<number>>`], where:

- `Optional` -- means if there is a failure for that instance continue processing other instances.
- `Mandatory` -- means if an error occurs for this instance, the entire operation fails.
- `Ignore` -- means the instance is not part of the operation.
- `Group <MinSucceeded=<number>` -- means to combine groups of Oracle instances, and if the specified number of group members is successful, then the operation is successful; otherwise, if less than the number of group members that is specified is successful, the operation fails.

Each attribute value determines the success requirement for that peer group and will be referenced during failure cases of asgctl operations to determine whether or not to continue with the Oracle Application Server Guard operation. For example, when the success requirement is specified as mandatory, the particular Oracle Application Server Guard operation must be successful for the specified instance for that production-standby peer combination; otherwise, the Oracle Application Server Guard operation ceases, execution is rolled back to its starting point of execution, and an error message is returned.

For example, the following XML policy file in use for an asymmetric topology for the failover operation specifies that this asgctl operation is mandatory for the infra instance, optional for the portal_1 and portal_2 instances, can be ignored for the portal_3 instance, and must be successful for a minimum of any two of the group of three instances, BI_1, BI_2, and BI_3.

```
<policy>
  <instanceList successRequirement="Mandatory">
     <instance>infra</instance>
  </instanceList >
  <instanceList successRequirement="Optional">
     <instance>portal_1</instance>
     <instance>portal_2</instance>
  </instanceList >
  <instanceList successRequirement="Ignore">
```

```
      <instance>portal_3</instance>
   </instanceList >
   <instanceList successRequirement="Group" minSucceed="2">
      <instance>BI_1</instance>
      <instance>BI_2</instance>
      <instance>BI_3</instance>
   </instanceList >
</policy>
```

## 2.5 Discovering Oracle Application Server 10.1.3 Instances in Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology

As described in Section 1.1.3.5, "Redundant Multiple Oracle Application Server 10.1.3 Homes J2EE Topology," you can discover Oracle Application Server 10.1.3 instance changes in a redundant multiple Oracle Application Server 10.1.3 Homes J2EE topology. In a typical scenario, Oracle Application Server 10.1.3 J2EE is installed on a node with an OC4J instance and you want to add this node and instance to the existing redundant multiple Oracle Application Server 10.1.3 Homes J2EE topology. The following steps take you through a typical scenario:

1. Connect to the Oracle Application Server Guard server.

   ```
   For 10.1.2.x and 10.1.4.x releases:
   ASGCTL> connect asg prodinfra ias_admin/<adminpwd>
   Successfully connected to prodinfra:7890
   ASGCTL>

   For 10.1.3.x releases:
   ASGCTL> connect asg prodinfra oc4jadmin/<adminpwd>
   Successfully connected to prodinfra:7890
   ASGCTL>
   ```

2. Assume a Disaster Recovery Administrator has installed OracleAS 10.1.3 J2EE on a new node with an OC4J instance. Through dynamic node discovery, the cluster manages itself and automatically updates the information in each OPMN configuration file, `opmn.xml` on each node within the cluster.

3. Next, perform a discover topology within farm command. This operation determines all instances within the Oracle Application Server 10.1.3 cluster for this production site, generates the Disaster Recovery topology XML file that describes the production topology, and then propagates this topology XML file to all instances in the Disaster Recovery production environment. Note that any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby environment.

   ```
   ASGCTL> discover topology within farm
   ```

4. Now, having recently installed another oc4j instance named oc4j3 on the existing midtier host system named prodmid2, add to the local topology file an instance named oc4j3 specifying the host system named prodmid2. To propagate this updated local topology file to all instances in the Disaster Recovery production environment for this Oracle Application Server 10.1.3 cluster, specify the `to topology` keyword. Note that any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby topology.

   ```
   ASGCTL> add instance oc4j3 on prodmid2.oracle.com to topology
   ```

## 2.6 Adding or Removing Oracle Application Server 10.1.3 Instances to Redundant Single Oracle Application Server 10.1.3 Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology

As described in Section 1.1.3.6, "Redundant Single Oracle Application Server 10.1.3 Oracle Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology," you can add or remove Oracle Application Server 10.1.3 single home J2EE instances to or from an existing Oracle Interenet Directory (OID) 10.1.4.2 topology, thus creating or modifying a mixed version Disaster Recovery environment. The following steps take you through a typical scenario:

1. Connect to the Oracle Application Server Guard server.

```
ASGCTL > connect asg prodinfra ias_admin/<adminpwd>
Successfully connected to prodinfra:7890
ASGCTL>
```

2. Assume that a discover topology command was performed when you first set up the Disaster Recovery environment on the existing OID 10.1.4.2 topology, thus creating a topology XML file that was propagated to all instances in the Disaster Recovery environment. Also assume that no other changes have been made to the topology, so the topology XML files are current or up-to-date.

3. Assume that you have completed Oracle Application Server 10.1.3 installations using the Integrated Web server, J2EE Server, and OPMN installation option in a single Oracle home on four individual systems. This creates the redundant single Oracle home J2EE topology. Assume that the OracleAS Disaster Recovery solution is configured for each new node. Now you want to add the Oracle Application Server 10.1.3 instances named OC4J1 and OC4J2 to the existing OID 10.1.4.2 topology. To do this, perform the following asgctl commands:

```
ASGCTL> add instance oc4j1 on prodinfra.oracle.com to topology
ASGCTL> add instance oc4j2 on prodinfra.oracle.com to topology
```

   Performing each `add instance` command updates the local topology file on the Oracle Application Server Guard server to which the Oracle Application Server Guard client is connected. To propagate the updated topology file to all instances in the Disaster Recovery production environment, specify the `to topology` key word. This operation then results in a redundant single Oracle Application Server 10.1.3 Oracle home J2EE topology with instances oc4j1 and oc4j2 being integrated with an existing OID 10.1.4.2 production topology. Note that any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, switchover, and failover will automatically propagate the production topology file across the standby topology.

## 2.7 Oracle Application Server Guard Operations -- Standby Site Cloning of One or More Production Instances to a Standby System

Standby site cloning is the process of copying a single Oracle Application Server instance at a production site host to a standby site peer host (using the clone instance command) or copying two or more Oracle Application Server instances at production site hosts to standby site peer hosts (using the clone topology command).

One of the underlying technologies used by Oracle Application Server Guard to perform a cloning operation is the OracleAS Recovery Manager's backup and restore loss of host capability. See the section on recovering a loss of host automatically in *Oracle Application Server Administrator's Guide* for more information, including a list of prerequisites.

> **Note:** When you use a `clone instance` or `clone topology` command, the target host or hosts at the standby site should have *only* the Oracle Application Server Guard standalone kit installed in the Oracle Application Server Guard home. This software is required to copy the Oracle Application Server homes to the standby site hosts.
>
> No other Oracle software should be installed on the standby site target host or hosts for the cloning operation. The reason for this is that during the clone operation, the entire Oracle Universal Installer Central Inventory for each production site host involved in the clone operation is copied to the standby site peer host during the clone operation, which means that the Central Inventory from each production site host involved in the clone operation overwrites the Central Inventory on the standby site peer host.

Some of the underlying operations require elevated privileges, `root` for the UNIX environments and `Administrator` for Windows. On Windows, the user must ensure that the Oracle Application Server Guard client and server are started with `Administrator` privileges.

### Clone Instance

The `clone instance` command is used to create a new Oracle Application Server instance at a standby site host from an existing Oracle Application Server instance at a production site host.

There are two phases of clone. The first phase is to create the Oracle home and register it within the system environment. The second phase is to perform the Oracle Application Server Guard instantiate operation to link it into the OracleAS Disaster Recovery environment and logically match the Oracle home with its corresponding production home.

A series of `clone instance` commands for different instances is equivalent to a `clone topology` operation.

Read the clone instance reference information before using the `clone instance` command.

### Clone Topology

The `clone topology` command performs a `clone instance` operation across a group of hosts. By default, the `clone topology` command copies *all* of the Oracle Application Server homes on *all* of the production site hosts to the standby site peer hosts. However, you can use a policy file with a `clone topology` command, which allows the clone operation to copy only a subset of the Oracle Application Server homes at the production site hosts to the standby site peer hosts.

There are two methodologies that you must be aware of when planning for an OracleAS Disaster Recovery site setup:

- Creating a pure OracleAS Disaster Recovery site

- Adding Oracle Application Server homes to an existing site with OracleAS Disaster Recovery enabled

Each operation requires a different methodology to integrate the newly installed Oracle homes into the existing site or combine them into a standby site for a production site.

Read the clone topology reference information before using the `clone topology` command.

### Creating a Pure OracleAS Disaster Recovery Site

Prior to Oracle Application Server 10g release 10.1.2.0.2, this was the only type of site Oracle Application Server Guard could support. An OracleAS Disaster Recovery configuration was supported only for the default OracleAS Infrastructure and OracleAS middle-tier install types. With this type of configuration, all the Oracle homes on the production site hosts and standby site hosts were created using the Oracle Universal Installer. The Oracle Application Server Guard `instantiate topology` command created the Disaster Recovery relationships between the production and standby Oracle homes.

### Adding OracleAS Homes to an Existing Site with OracleAS Disaster Recovery Enabled

After an OracleAS site is OracleAS Disaster Recovery enabled, the relationship between the Oracle homes at the production and standby sites has been created. For releases previous to Oracle Application Server 10g release 10.1.2.0.2, the only way to add new instances to the site was to break the standby relationship, add the new instance at the production site using Oracle Installer, add the new instance to the standby site using Oracle Installer, and then re-create the standby site. With Oracle Application Server 10g 10.1.2.0.2 and later releases, you can use the `clone instance` command to add instances to a standby site.

For example, if you need a new middle tier to scale out the services in the middle tier, you can install the new instance at the production site. This operation creates the Oracle Application Server home for the instance and establishes the necessary relationships within the Oracle Application Server repositories at the production site.

With Oracle Application Server Guard asymmetrical topology support in Oracle Application Server 10g 10.1.2.0.2 and later releases, this Oracle home can optionally be ignored in regard to the site's OracleAS Disaster Recovery solution. If you want to add this instance to the standby site, the `clone topology` command will create the OracleAS Oracle home at the standby target host and establish the production-standby relationship for this instance. Before issuing this command, the standalone Oracle Application Server Guard kit must be installed and started at the target host (see the OracleAS Disaster Recovery installation information in *Oracle Application Server Installation Guide* for more information) and a `discover topology` command should be performed to discover the new instance in the production topology.

> **Warning:** Do not perform a clone operation to a standby site host that contains an existing Oracle home, other than the standalone Oracle Application Server Guard home, because it will get overwritten. Perform a clone operation only to a standby site host where no Oracle home is installed other than the standalone Oracle Application Server Guard home.

Some situations in which cloning operations are useful are:

- When you want to add two or more production instances to standby site hosts.
- When you want to add a single production instance to a standby site host.

More information about cloning is provided in the following section.

## 2.7.1 Cloning Single or Multiple Production Instances to a Standby System

Whether you are cloning a single production instance or multiple production instances to a standby site, the prerequisites and steps to follow are identical; the only difference is the actual asgctl command you would use for either cloning operation. For this reason, this section combines the cloning information and indicates where there are some minor differences.

### Cloning a Single Production Instance to a Standby Site

As an example, you want to add a production Oracle Application Server instance to a standby site host. You can use the `clone instance` command, which copies the Application Server instance at the production site host to the same directory on the standby site peer host and then performs an instantiate operation.

### Cloning Multiple Production Instances to a Standby Site

As an example, you want to add two or more production instances to a standby site host. You can use the `clone topology` command, which copies these Application Server instances to the same directories on the standby site peer hosts and then performs an instantiate operation.

If you want to use a policy file, edit and use the clone policy file (`clone_policy.xml`). Specify this file in the `using policy <file>` parameter of the clone topology command to clone a standby topology for only those instances specified accordingly. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information about using a policy file.

### Special Considerations

Before cloning multiple production instances to a standby site, consider the following:

- Each node should have its own virtual hostname mapped to the physical hostname's IP address in the hosts file. For example, edit the following:

  Windows: %SystemRoot%\system32\drivers\etc\hosts

  UNIX: /etc/hosts

  *ip_address* prodnode1.*domain*.com prodnode1 vhostdr.*domain*.com vhostdr

- The *only* type of database that is cloned by either a `clone instance` command or a `clone topology` command is a database with the Metadata Repository schemas that is created during an Application Server Infrastructure installation in the Application Server home. For more information on including databases in your OracleAS Disaster Recovery topology and configuring Oracle Data Guard for those databases, refer to Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology."

- For UNIX platforms only, the tar utility is used by the ASG `clone instance` and `clone topology` commands. The target host(s) for these operations must have a version of GNU tar in the default PATH of the system user account in which the standalone ASG install runs. GNU tar can be obtained at the following location:

  `http://www.gnu.org/software/tar/`

- For the Windows platform, add the directory that contains the jar utility to the PATH when installing a JDK on the standby site host. Otherwise, the ASG on the standalone site host cannot access the jar.exe utility.

### Prerequisites

The production instance or instances to be cloned cannot exist on the standby site host.

The following are prerequisites for performing the clone instance and clone topology operation to the standby site host:

- Install the Oracle Application Server Guard standalone kit in its own Oracle home on each standby site host. Do not install any other Oracle components on the standby site host.
- A Java development kit with its jar utility must be installed on the standby site host.
- For Windows hosts, release 5.0.2134.1 or higher of the services kit (`sc.exe`) must be installed under C:\WINDOWS\system32 on the production and standby site hosts.

  If you do not take this step prior to performing a cloning operation on Windows, you may see errors similar to these during a cloning operation:

  ```
  stajz09: -->ASG_DUF-4040: Error executing the external program or script.
  The error code is "255"
  "IM.asinfra.us.oracle.com"
  stajz09P: -->ASG_IAS-15689: Error running the backup script
  stajz09: -->ASG_IAS-15685: Failed to backup configuration data for instance
  "IM.asinfra.us.oracle.com"
  stajz09: -->ASG_DUF-3027: Error while executing Clone Instance at step -
  backup step.
  stajz09: -->ASG_DUF-3027: Error while executing Clone Topology at step -
  clone home step.
  ```

- For the dcm-daemon component in the %ORACLE_HOME%\opmn\conf\opmn.xml file on Windows hosts, increase the start timeout parameter's retry interval to 5 seconds. The following example shows the section of the opmn.xml file for the dcm-daemon component with the start timeout parameter's retry interval set to 5:

  ```
  <ias-component id="dcm-daemon" status="enabled" id-matching="true">
      <process-type id="dcm-daemon" module-id="DCMDaemon">
          <start timeout="600" retry="5"/>
          <stop timeout="120"/>
          <process-set id="dcm" numprocs="1">
  ```

  If you do not take this step prior to performing a cloning operation on Windows, you may see errors similar to these during a cloning operation:

  ```
  stajz09: -->ASG_SYSTEM-100: Command "C:\work\im/opmn/bin/opmnctl.exe shutdown"
  failed, check log file C:\work\im\dsa\bkup\log/2007-07-17_01-41-51_loha.log
  for detail.
  stajz09: -->ASG_SYSTEM-100: Failure : prepare failed.
  stajz09: -->ASG_SYSTEM-100:
  stajz09: -->ASG_SYSTEM-100: OPMN managed processes could not be stopped.
  stajz09: -->ASG_SYSTEM-100: Status code:
  stajz09: -->ASG_SYSTEM-100: opmnctl shutdown failed.
  ```

### Procedure

The basic procedure consists of the following pre-clone steps and clone steps.

### Pre-Clone Steps

Perform the following steps:

1. For the `clone instance` command, log in as su - root on UNIX or as Administrator on Windows to the production site host where the Application Server instance that you want to clone is installed. For the `clone topology` command, log in as su - root on UNIX or as Administrator on Windows to each production site host where an Application Server instance that you want to clone is installed.

2. For the `clone instance` command, CD to the instance home on the production site host that you want to clone. For the `clone topology` command, CD to the instance homes on the production site hosts for each instance that you want to clone.

3. On Windows, make sure that release 5.0.2134.1 or higher of the services kit (`sc.exe`) is installed in the `C:\WINDOWS\system32` directory on the production site hosts.

4. Shut down the Oracle Application Server Guard server.

   ```
   For 10.1.2.x and 10.1.4.x releases, this command stops OracleAS Guard on UNIX:
   > <ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=DSA

   For 10.1.3.x releases, this command stops OracleAS Guard on UNIX:
   > <ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=ASG

   For 10.1.2.x and 10.1.4.x releases, this command stops OracleAS Guard on
   Windows:
   > <ORACLE_HOME>\opmn\bin\opmnctl stopproc ias-component=DSA

   For 10.1.3.x releases, this command stops OracleAS Guard on Windows:
   > <ORACLE_HOME>\opmn\bin\opmnctl stopproc ias-component=ASG
   ```

5. On UNIX, log in as root on each standby site host that an Application Server instance home will be cloned to and make sure `dsaServer.sh` in *<ORACLE_HOME>*`/dsa/bin` is executable by everyone. If it is not, record the permission, then change the executable permission by issuing the following command:

   ```
   chmod +x dsaServer.sh
   chmod u+x asgexec
   ```

6. On Windows, on the standby site host or hosts that an Application Server instance home will be cloned to:

   1. Add the jdk\bin path to the system path on the standby site host.
   2. Create a new command window.
   3. In the new command window, run the `jar` command with no parameters to make sure the jar utility is found.
   4. Make sure that release 5.0.2134.1 or higher of the services kit (`sc.exe`) is installed in the `C:\WINDOWS\system32` directory on the standby site host or hosts.

7. On both the production site host or hosts and the standby site host or hosts that will be involved in the clone operation, invoke asgctl and issue the startup command.

```
From the <ORACLE_HOME>/dsa/bin directory on a UNIX host:
> asgctl.sh startup

From the <ORACLE_HOME>\dsa\bin directory in the new command window on a Windows
host:
> asgctl startup
```

8. On UNIX hosts, log out as root.

### Clone Steps

Perform the following steps:

1. Log in as user (non root user on UNIX hosts) to the production site host or hosts where the Application Server instance or instances that you want to clone are installed.
2. For the `clone instance` command, CD to the instance home on the production site host for the Application Server instance that you want to clone. For the `clone topology` command, CD to the instance homes on the production site hosts for each Application Server instance that you want to clone.
3. Invoke asgctl on the production site host and run the `clone instance` command to clone the Application Server instance and home to the same directory on the standby site peer host. Invoke asgctl on any of the production site hosts and run the `clone topology` command to clone the Application Server instances and homes for all of the production site hosts to the same directories on the standby site peer hosts. Remember that you can use a policy file to exclude specific instance homes from a clone topology operation.

> **Note:** In the command output, you will see a number of connect messages. This is normal as the Oracle Application Server Guard server is recycled during these operations.

```
> asgctl.sh

For 10.1.2.x and 10.1.4.x releases, use this command to connect to an ASG
server:
ASGCTL> connect asg prodoc4j ias_admin/adminpwd
Successfully connected to prodoc4j:7890

For 10.1.3.x releases, use this command to connect to an ASG server:
ASGCTL> connect asg prodoc4j oc4jadmin/adminpwd
Successfully connected to prodoc4j:7890

ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb

CLONE INSTANCE -- CLONE AN INSTANCE EXAMPLE

ASGCTL> clone instance portal_2 to asmid2.oracle.com
.
.
.

CLONE TOPOLOGY -- CLONE MULTIPLE INSTANCES EXAMPLE

# Command to use if you are cloning all the Application Server
# instances at all the production site hosts to the standby site
# peer hosts:
ASGCTL> clone topology to standbyinfra.oracle.com
.
.
.

# Command to use if you are using a policy file (where <file>
# is the full path and file specification of the clone policy file)
# to clone a subset of the Application Server instances at the production
# site hosts to the standby site peer hosts:
ASGCTL> clone topology to standbyinfra.oracle.com using policy <file>
.
.
.

ASGCTL> disconnect
ASGCTL> exit
>
```

4. Log out of the production site host or hosts.

> **Note:** If Oracle Application Server Guard does not run as root on UNIX hosts, the user will be prompted by the Oracle Application Server Guard client to run the underlying operations at each of the instance homes as root (manually) in order to continue with the operation.

The last step completes the cloning operation and brings the hosts back to where they were before you started the operation. At this point, you could invoke asgctl, connect to a production site host, discover the topology, and then perform a verify operation to determine if the production and standby topologies are valid and consistent with one another as you would expect them to be.

## 2.8 Oracle Application Server Guard Operations -- Standby Instantiation and Standby Synchronization

After adhering to the following conditions, you are ready to use the Oracle Application Server Guard for standby instantiation and standby synchronization.

- Meet the requirements for the implementation of the OracleAS Disaster Recovery solution as described in Section 1.1.1, "OracleAS Disaster Recovery Requirements," Section 1.1.3, "Supported Topologies," and Section 1.2, "Preparing the OracleAS Disaster Recovery Environment."
- Install the OracleAS Disaster Recovery solution as described in Section 1.4, "Overview of Installing Oracle Application Server."

The following sections describe standby instantiation and standby synchronization:

- Section 2.8.1, "Standby Instantiation"
- Section 2.8.2, "Standby Synchronization"

See Chapter 5, "Oracle Application Server Guard asgctl Command-line Reference" for Oracle Application Server Guard command-line asgctl utility reference information.

### 2.8.1 Standby Instantiation

The standby instantiation operation performs a number of operations to set up and maintain a logical mirror of the production site at the standby site. Oracle Application Server Guard is used to coordinate the distributed operations across the production and standby sites to ensure the disaster recovery functionality is enabled. The setup operations are:

- Uses a previous topology file created by performing a discovery topology operation.
- Verifies the topology definitions to ensure they comply with the rules of the OracleAS Disaster Recovery environment.
- Mirrors the configuration information of all the Oracle homes in the OracleAS topology to the corresponding Oracle home at the standby site.
- If you want to use a policy file, edit and use the instantiate policy file (`instantiate_policy.xml`). Specify this file in the `using policy <file>` parameter of the instantiate topology command to instantiate a standby topology for only those instances specified accordingly. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.
- Reports any errors found for correction.

The procedure to perform a standby instantiation operation uses the following example, which assumes that you have invoked the Oracle Application Server Guard client and performed a `discover topology` command to create a topology file.

See Section 5.2.1.1, "Special Considerations for Running Instantiate and Failover Operations in CFC Environments" if you have an OracleAS Disaster Recovery configuration in a CFC environment and are about to perform an instantiate operation.

1. Connect to the Oracle Application Server Guard server.

```
ASGCTL > connect asg prodinfra ias_admin/<adminpwd>
Successfully connected to prodinfra:7890
ASGCTL>
```

2. Specify the primary OracleAS Metadata Repository database. See Section 2.11.1.2, "Specifying the Primary Database" for more information. If you have multiple OracleAS Metadata Repositories in your topology, you must authenticate each one using the `set primary database` command.

```
ASGCTL> set primary database sys/testpwd@asdb
```

3. Dump the policies (dump policies command), then edit and use the verify policy file (`verify_policy.xml`) and the instantiate policy file (`instantiate_policy.xml`) to specify the success requirement attribute for each instance in the file. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.

```
ASGCTL> dump policies
Generating default policy for this operation
Creating policy files on local host in directory
"/private1/OraHome2/asr1012/dsa/conf/"
```

4. Verify the topology. The network hostname `standbyinfra` is used.

```
ASGCTL> verify topology with standbyinfra
```

5. Instantiate the topology at the secondary site. The network hostname `standbyinfra` is used. This command assumes that all the Oracle homes have

been installed using Oracle Universal Installer software or cloned using the `clone instance` command or `clone topology` command. Specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `instantiate_policy.xml` file.

```
ASGCTL> instantiate topology to standbyinfra using policy <file>
```

Whenever a standby instantiation is performed using the asgctl `instantiate topology` command, a synchronization operation is also performed. Thus, you do not need to perform another synchronization operation immediately following the instantiation operation. If a period of time had passed following an instantiate operation, ensure that both the primary and standby sites are consistent. Then, perform a sync topology operation to ensure any changes that occurred on the primary site are applied to the secondary site.

## 2.8.2 Standby Synchronization

The Oracle Application Server Guard synchronization operation synchronizes the standby site with the primary site to ensure that the two sites are logically consistent. This operation is necessary whenever any of the following circumstances exist:

- Deploy a new application or redeploy an existing application - Both the deployment of a new application and the redeployment of an existing application require changes to schema-based information in the metadata repository as well as component configuration information distributed among the Oracle homes in an Oracle Application Server topology. This information has to be uniformly deployed at the standby site.
- Configuration changes - Specific changes, small to large, to a configuration, must be reflected at the standby site.
- User Provisioning - The default Infrastructure installation maintains the database for Oracle Internet Directory. As new users are added to the database, they should be synchronized to the standby site on a schedule that fulfills the business availability requirements.
- Periodic full synchronization - By default, the synchronization operations synchronizes only the pieces of configuration that have changed since the last synchronization operation. During test cycles or occasional complex configuration changes, administrators may want to fully refresh of their configuration information to the standby site to ensure mirroring of these changes.

You can specify a full or incremental synchronization. By default, an incremental synchronization is performed, which offers the best performance. However, in the following three circumstances a full synchronization should be specified:

- When you want to force a full synchronization to happen for some reason, such as synchronizing the standby site completely with the primary site.
- When you know there are many transactional changes over a short period of time on the primary site that must be synchronized with the secondary site.
- When you know that there is a large accumulation of transactional changes over a long period of time on the primary site that must be synchronized with the secondary site.

As part of the synchronization operation, a verify operation is performed to ensure the required OracleAS Disaster Recovery environment is maintained. Additionally, if new Oracle Application Server instances are installed into the Oracle Application Server topology, Oracle Application Server Guard will discover these installations.

If you want to use a policy file, edit and use the synchronization policy file (`sync_policy.xml`). Specify this file in the `using policy <file>` parameter of the sync topology command for synchronizing a standby topology for only those instances specified accordingly. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.

The following example assumes that you have invoked the Oracle Application Server Guard client and performed a `discover topology` command to create a topology file.

The procedure to perform standby synchronization is as follows:

1. Connect to the Oracle Application Server Guard server.

   ```
   ASGCTL > connect asg prodinfra ias_admin/<adminpwd>
   Successfully connected to prodinfra:7890
   ASGCTL>
   ```

2. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information.

   ```
   ASGCTL> set primary database sys/testpwd@asdb
   ```

3. Synchronize the secondary site with the primary site.

   ```
   ASGCTL> sync topology to standbyinfra
   ```

## 2.9 Runtime Operations -- Oracle Application Server Guard Switchover and Failover Operations

Runtime operations include dealing with outages, whether they are scheduled or unscheduled (see Section 2.9.1, "Outages"), and monitoring ongoing Oracle Application Server Guard operations using the asgctl command-line utility and troubleshooting (see Section 2.10, "Monitoring Oracle Application Server Guard Operations and Troubleshooting").

### 2.9.1 Outages

Outages fall into two categories: scheduled and unplanned.

The following subsections describe these outages.

#### 2.9.1.1 Scheduled Outages

Scheduled outages are planned outages. They are required for regular maintenance of the technology infrastructure supporting the business applications and include tasks such as hardware maintenance, repair and upgrades, software upgrades and patching, application changes and patching, and changes to improve the performance and manageability of systems. Scheduled outages can occur either for the production or standby site. Descriptions of scheduled outages that impact the production or standby site are:

- Site-wide maintenance

  The entire site where the current production resides is unavailable. Examples of site-wide maintenance are scheduled power outages, site maintenance, and regularly planned switchover operations.

- OracleAS Cold Failover Cluster clusterwide maintenance

This is scheduled downtime of the OracleAS Cold Failover Cluster for hardware maintenance. The scope of this downtime is the whole hardware cluster. Examples of clusterwide maintenance are repair of the cluster interconnect and upgrade of the cluster management software.

- Testing and validating the standby site as a means to test OracleAS Disaster Recovery readiness.

For scheduled outages, a site switchover operation has to be performed, which is explained in the section that follows.

### Site Switchover Operations

A site switchover is performed for planned outages of the production site. At the production site, the ASG server must be started on each of the hosts, and the listener and database must be started on the Infrastructure host. At the standby site, the ASG server must be started on each of the hosts, and the listener must be started and the database must be in standby mode on the Infrastructure host. Before you perform the switchover, execute the asgctl `verify topology` command on the production site to confirm that the production topology is running and the configuration is valid. Then execute the asgctl `verify topology` command (specifying the standby infrastructure host) to confirm that the standby site is consistent with the production site and meets Disaster Recovery requirements. After you have confirmed that the topologies are valid, perform the switchover operation. During the site switchover operation, the application of the database redo logs is synchronized to match the backup and restoration of the configuration files for the middle tier and OracleAS Infrastructure installations.

> **Note:** During a switchover operation, the `opmn.xml` file is copied from the primary site to the standby site. For this reason, the value of the TMP variable must be defined the same in the `opmn.xml` file on both the primary and standby sites, otherwise this switchover operation will fail with a message that it could not find a directory. Therefore, make sure the TMP variable is defined identically and resolves to the same directory structure on both sites before attempting a switchover operation.

During site switchover, considerations must be made to avoid long periods of cached DNS information. Modifications to the site's DNS information, specifically time-to-live (TTL), must be performed. See Section 1.5, "Wide Area DNS Operations" for instructions.

If you want to use a policy file, edit and use the switchover policy file (`switchover_policy.xml`). Specify this file in the `using policy <file>` parameter of the switchover topology command for switching over to the standby topology only those instances specified accordingly. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information. This example does not show the use of a policy file.

See Section 5.2.1.3, "Special Considerations for Running a Switchover Operation in CFC Environments" if you have an OracleAS Disaster Recovery configuration in a CFC environment and are planning a switchover operation.

To switchover from the production site to the standby site, perform the following steps:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS

Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. Invoke the Oracle Application Server Guard client command-line utility asgctl (on UNIX systems, `asgctl.sh` is located in *<ORACLE_HOME>*`/dsa/bin` and on Windows systems, `asgctl.bat` is located in *<ORACLE_HOME>*`\dsa\bin`) and connect to the Oracle Application Server Guard server.

```
Use these commands for 10.1.2.x and 10.1.4.x releases:
> asgctl.sh
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg prodinfra ias_admin/<adminpwd>

Use these commands for 10.1.3.x releases:
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg prodinfra oc4jadmin/<adminpwd>
```

3. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information.

```
ASGCTL> set primary database sys/testpwd@asdb
```

4. Switchover the topology to the secondary site. If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file.

```
ASGCTL> switchover topology to standbyinfra
```

> **Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

5. Disconnect from the *old* primary site Oracle Application Server Guard server.

```
ASGCTL> disconnect
```

6. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations".

7. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

### Special Switchover Operation Considerations

This section describes the following special considerations relating to the switchover operation.

- When performing a switchover operation from a primary site with two Oracle Identity Management instances running to a standby site representing an asymmetric topology with only one Oracle Identity Management instance running, which means that the other node is to be ignored on the switchover site, the system administrator must not only edit the `switchover_policy.xml` policy file to indicate that this other node is to be set to Ignore, but must also shutdown all processes running on that node in order for the switchover operation to be successful. For example, if the two Oracle Identity Management instances running on the primary site are im.systemA.us.oracle.com and im.systemB.us.oracle.com, and the other node (im.systemB.us.oracle.com) is to be ignored on the switchover site, the system administrator must also shutdown all processes running on that node (im.systemB.us.oracle.com) in order for the switchover operation to succeed.

- When the `discover topology` command is issued following a switchover operation and the asymmetric standby site topology originally had one or more fewer middle tiers (for example, instA and instB) than there were in the original production site topology (instA, instB, and instC), a warning error message displays for each missing instance of a middle tier (instC, in this case). This warning error message is expected and can be ignored. When a `discover topology` command is issued following a switchover operation, OracleAS Server Guard reads the Oracle Internet Directory information, which is an exact copy of the original primary site Oracle Internet Directory information on this new primary site (former standby site). Because this Oracle Internet Directory information is identical to the original primary site Oracle Internet Directory information, when OracleAS Server Guard visits the host or home of each instance of these middle tiers to verify their existence, it discovers that some of the middle tiers do not exist, and issues warnings.
- Prior to performing an asgctl switchover operation in an asymmetric topology for instances that do not have a standby peer, you must perform an opmnctl stopall command to shutdown all components on each of these ignored instances on the primary site. When an XML policy file is in use for an asymmetric topology and has the <instanceList successRequirement ="Ignore" set for an instance, in a switchover operation Oracle Application Server Guard ignores that instance. Oracle Application Server Guard, on a switchover operation shuts down all components on the old primary site except for Oracle Application Server Guard and OPMN and ignores instance B because the policy file specifies to do so. The switchover operation fails because all components are not shut down on the primary site, in this case instance B, because the policy file specifies to ignore instance B on the primary site, which has no standby peer. To avoid this problem, perform an opmnctl stopall operation for all components on instance B prior to the switchover operation in order for the switchover operation to succeed in this asymmetric topology.
- After a switchback operation (`switchover topology to <primary site>`), the database will be started up on only one of the Oracle RAC nodes by Oracle Application Server Guard; however, the remaining Oracle RAC instances on the primary site must be started up manually.

### 2.9.1.2 Unplanned Outages

An unplanned outage that impacts a production site occurs when it becomes unavailable and there is no possibility of restoring the production site to service within a reasonable period of time. This includes site-wide outages at the production site such as fire, flood, earthquake, or power outages.

Unplanned outages warrant performing a failover operation of the production site to the standby site.

#### Site Failover Operations

A site failover operation is performed for unplanned outages for the production site. Failover operations require the restoration of the configuration and Infrastructure data to a consistent point in time. Oracle Application Server Guard ensures that the site services are brought up in a consistent fashion to the point of the last sync operation. A failover operation restores to the last synchronization point.

If you want to use a policy file, edit and use the failover policy file (`failover_policy.xml`). Specify this file in the `using policy <file>` parameter of the failover command for failing over to the standby topology only those instances specified accordingly. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.

See Section 5.2.1.1, "Special Considerations for Running Instantiate and Failover Operations in CFC Environments" if you have an OracleAS Disaster Recovery configuration in a CFC environment and are about to perform a failover operation.

To fail over the production site to the standby site, follow these steps:

1. Connect to the Oracle Application Server Guard server on the standby site. The network name is standbyinfra.

   ```
   For 10.1.2.x and 10.1.4.x releases:
   ASGCTL> connect asg standbyinfra ias_admin/<adminpwd>
   Successfully connected to standbyinfra:7890

   For 10.1.3.x releases:
   ASGCTL> connect asg standbyinfra oc4jadmin/<adminpwd>
   Successfully connected to standbyinfra:7890
   ```

2. Specify that the primary OracleAS Metadata Repository database on the standby site is now identified as the *new* primary database on this *new* production site. The keyword **new** is shown as bold text in the following example to indicate its importance as a key word. If you have multiple OracleAS Metadata Repositories in your topology, you must authenticate each one using the `set new primary database` command.

   ```
   ASGCTL> set new primary database sys/testpwd@asdb
   ```

3. Perform an asgctl failover operation.

   ```
   ASGCTL> set trace on all
   ASGCTL> failover
   ASGCTL> disconnect
   ```

4. Discover the topology. You must perform this operation to create a new topology file for this production site.

   For 10.1.2.x and 10.1.4.x environments, discover the topology as follows:

   ```
   ASGCTL> discover topology oidpassword=oidpwd
   ```

   For 10.1.3.x environments, discover the topology as follows:

   ```
   ASGCTL> discover topology within farm
   ```

   You can also use the `add instance` command to add the standalone database instances to the Disaster Recovery topology as follows:

   ```
   ASGCTL> add instance asdb on vhostdr.oracle.com
   ```

## 2.10 Monitoring Oracle Application Server Guard Operations and Troubleshooting

After setting up your OracleAS Disaster Recovery solution, and instantiating the standby topology, and synchronizing the standby topology, you can use the Oracle Application Server Guard client command-line utility asgctl to issue commands through the coordinating Oracle Application Server Guard server to monitor asgctl operations and perform troubleshooting tasks. A typical Oracle Application Server Guard monitoring or troubleshooting session may involve the following tasks:

1. Section 2.10.1, "Verifying the Topology"
2. Section 2.10.2, "Displaying the Current Operation"
3. Section 2.10.3, "Displaying a List of Completed Operations"
4. Section 2.10.4, "Stopping an Operation"
5. Section 2.10.5, "Tracing Tasks"
6. Section 2.10.6, "Writing Information About the Topology to a File"

As asgctl commands are issued through the Oracle Application Server Guard client and requests are then made to the coordinating Oracle Application Server Guard server, the coordinating Oracle Application Server Guard server communicates these requests to the other Oracle Application Server Guard servers in the production and standby topologies, and status messages are returned to the Oracle Application Server Guard client as well as any error messages should a particular task encounter a problem. Section 2.10.7, "Error Messages" describes where you can obtain more information about these error messages.

### 2.10.1 Verifying the Topology

To validate that the primary topology is running and the configuration is valid, enter the following asgctl command at the asgctl prompt.

```
For Application Server 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg ias_admin/<adminpwd>
Successfully connected to prodinfra:7890
ASGCTL> discover topology oidpassword=<oidpwd>
ASGCTL> verify topology
Generating default policy for this operation
prodinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
ASGCTL>

For Application Server 10.1.3.x releases:
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodinfra:7890
ASGCTL> discover topology within farm
ASGCTL> verify topology
Generating default policy for this operation
prodinfra:7890
     HA directory exists for instance asr1013.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
ASGCTL>
```

If you want to use a policy file, edit and use the verify policy file (`verify_policy.xml`) to specify the success requirement attribute for each instance in the file. Then specify the `using policy <file>` parameter in the `verify topology` command where `<file>` represents the path and file specification for the `verify_policy.xml` file. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information.

To compare a primary topology to which the local host is a member with a standby topology and ensure that they are consistent with one another and that both topologies conform to OracleAS Disaster Recovery requirements, enter the following asgctl command at the asgctl prompt and specify the name of the standby host system.

```
ASGCTL> dump policies
Generating default policy for this operation
Creating policy files on local host in directory
"/private1/OraHome2/asr1012/dsa/conf/"

ASGCTL> verify topology with standbyinfra
Generating default policy for this operation
prodinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
standbyinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
prodinfra:7890
    Verifying that the topology is symmetrical in both primary and standby configuration
ASGCTL>

# Command to use if you want to use a policy file
# verify topology with standbyinfra using policy <file>
```

## 2.10.2 Displaying the Current Operation

To display the status of all the current operations running on all nodes of the topology to which the Oracle Application Server Guard client is connected, enter the following asgctl command at the asgctl prompt:

```
ASGCTL> show operation
*************************************
OPERATION: 19
  Status: running
  Elapsed Time: 0 days, 0 hours, 0 minutes, 28 secs
  TASK: syncFarm
    TASK: backupFarm
      TASK: fileCopyRemote
      TASK: fileCopyRemote
    TASK: restoreFarm
      TASK: fileCopyLocal
```

### 2.10.3 Displaying a List of Completed Operations

To display only operations that have completed (are *not* running on any nodes of the topology to which the Oracle Application Server Guard client is connected for the current session), enter the following asgctl command at the asgctl prompt:

```
ASGCTL> show operation history
*************************************
OPERATION: 7
  Status: success
  Elapsed Time: 0 days, 0 hours, 0 minutes, 0 secs
  TASK: getTopology
    TASK: getInstance
*************************************
OPERATION: 16
  Status: success
  Elapsed Time: 0 days, 0 hours, 0 minutes, 0 secs
  TASK: getTopology
    TASK: getInstance
*************************************
OPERATION: 19
  Status: success
  Elapsed Time: 0 days, 0 hours, 1 minutes, 55 secs
  TASK: syncFarm
    TASK: backupFarm
      TASK: fileCopyRemote
      TASK: fileCopyRemote
    TASK: restoreFarm
      TASK: fileCopyLocall
```

### 2.10.4 Stopping an Operation

To stop a specific operation that is running on the server, enter the following asgctl command at the asgctl prompt and specify the operation number you want to stop. You can obtain the operation number you want to stop by entering a asgctl `show operation full` command.

```
ASGCTL> show operation full
*************************************
OPERATION: 19
  Status: running
  Elapsed Time: 0 days, 0 hours, 0 minutes, 28 secs
  Status: running
.
.
.
ASGCTL> stop operation 19
```

The operation is stopped at the most recently successfully completed sync point.

### 2.10.5 Tracing Tasks

To set a trace flag for a specific event and to log the output to the asgctl log files, enter the following asgctl command at the asgctl prompt and specify the **on** keyword and enter the trace flags to be enabled. In this case, the trace flag `DB` indicates that trace information regarding processing in the Oracle Database environment will be displayed. See the set trace command for more information about other trace flags that can be enabled. See the set trace command for a complete list of the trace flags that can be set.

```
ASGCTL> set trace on db
```

### 2.10.6 Writing Information About the Topology to a File

To write detailed information about the topology to which the local host is connected, enter the following asgctl command at the asgctl prompt and specify the path name and file name where the detailed output is to be written. The output is the same as the display shown in the dump topology command, except it is written to a file that you can save for future reference.

```
ASGCTL> dump topology to c:\dump_mid_1.txt
```

### 2.10.7 Error Messages

Appendix B, "Oracle Application Server Guard Error Messages" categorizes and describes the error messages that may appear while using the OracleAS Disaster Recovery solution.

## 2.11 Using Oracle Application Server Guard Command-Line Utility (asgctl)

This section includes the following subsections:

- Section 2.11.1, "Typical Oracle Application Server Guard Session Using asgctl"
- Section 2.11.2, "Periodic Scheduling of Oracle Application Server Guard asgctl Scripts"
- Section 2.11.3, "Submitting Oracle Application Server Guard Jobs to the Enterprise Manager Job System"
- Section 4.1.1, "Special Considerations for Multiple OracleAS Metadata Repository Configurations"
- Chapter 5, "Oracle Application Server Guard asgctl Command-line Reference"

### 2.11.1 Typical Oracle Application Server Guard Session Using asgctl

A typical Oracle Application Server Guard session using asgctl involves the following tasks, which are described in the following subsections:

- Section 2.11.1.1, "Getting Help"
- Section 2.11.1.2, "Specifying the Primary Database"
- Section 2.11.1.3, "Discovering the Topology"

One of the advantages of supporting an asgctl command-line interface is that you can place these asgctl commands in a proper sequence in a script as described in Section 2.11.1.4, "Creating and Executing an asgctl Script" and then execute the script as described in Section 2.11.2, "Periodic Scheduling of Oracle Application Server Guard asgctl Scripts" and Section 2.11.3, "Submitting Oracle Application Server Guard Jobs to the Enterprise Manager Job System".

#### 2.11.1.1 Getting Help

To get help on a particular command, enter the asgctl command at the asgctl prompt and specify the command name you for which you want help information. Otherwise, to get help on all commands, enter the following asgctl command at the asgctl prompt:

```
ASGCTL> help
    connect asg [<host>] [<ias_administrator_account>/<password>]
    disconnect
    exit
    quit
    add instance <instance_name> on <instance_host> [to topology]
    clone topology to <standby_topology_host> [using policy <file>] [no standby]
    clone instance <instance> to <standby_topology_host> [no standby
    discover topology [oidhost=<host>] [oidsslport=<sslport>] [oiduser=<user>]
oidpassword=<pass>
    discover topology within farm
    dump farm [to <file>]  (Deprecated)
    dump topology  [to <file>] [using policy <file>]
    dump policies
    failover [using policy <file>]
    help [<command>]
    instantiate farm to <standby_farm_host> (Deprecated)
    instantiate topology to <standby_topology_host> [using policy <file>]
    remove instance <instance_name> [from topology]
    set asg credentials <host> <ias_administrator_account>/<password> [for topology]
    set asg credentials <host> ias_admin/<password> [for farm] (Deprecated)
    set primary database <username>/<password>@<servicename> [pfile <filename> | spfile
<filename>]
    set new primary database <username>/<password>@<servicename> [pfile <filename> | spfile
<filename>]
    set noprompt
    set trace on|off <traceflags>
    sync farm to <standby_farm_host> [full | incr[emental]] (Deprecated)
    sync topology to <standby_topology_host> [full | incr[emental]] [using policy <file>]
    startup [asg]
    startup farm (Deprecated)
    startup topology
    shutdown [local]
    shutdown farm (Deprecated)
    shutdown topology
    show op[eration] [full] [[his]tory]
    show env
    stop op[eration] <op#>
    switchover farm to <standby_farm_host> (Deprecated)
    switchover topology to <standby_topology_host> [using policy <file>]
    verify farm [with <host>](Deprecated)
    verify topology [with <host>] [using policy <file>]
ASGCTL>
```

#### 2.11.1.2 Specifying the Primary Database

To identify the OracleAS Infrastructure database on the primary topology, enter the following asgctl command at the asgctl prompt and specify the user name and password for the database account with sysdba privileges to access the OracleAS Infrastructure database and the TNS service name of the OracleAS Infrastructure database:

```
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL>
```

The standby site uses the same values as specified for the primary database because the service name and password for both the primary and standby OracleAS

Infrastructure Databases must be the same. You must always set the primary database before performing an instantiate, sync, or switchover operation.

If you have multiple OracleAS Metadata Repositories in your topology, you must authenticate each one using the `set primary database` command.

#### 2.11.1.3 Discovering the Topology

You must perform a `discover topology` command when you first set up your OracleAS Disaster Recovery environment in order to initially create the `topology.xml` file. There after, you should perform a discover topology operation whenever you procure another Oracle home in a production site or change roles from a production to a standby site through a switchover or failover operation. The `discover topology` command queries Oracle Internet Directory for all instances within the topology that share the same Oracle Internet Directory for the production site. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files. Enter the following asgctl command at the asgctl prompt to discover the topology:

```
For Application Server 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg prodinfra ias_admin/<adminpwd>
Successfully connected to prodinfra:7890
ASGCTL> discover topology oidpassword=<oidpwd>
Discovering topology on host "infra" with IP address "123.1.2.111" prodinfra:7890
    Connecting to the OID server on host "infra.us.oracle.com" using SSL port
"636" and username "orcladmin"
    Getting the list of databases from OID
    Gathering database information for SID "asdb" from host "infra.us.oracle.com"
    Getting the list of instances from OID
    Gathering instance information for "asr1012.infra.us.oracle.com" from host
"infra.us.oracle.com"
    Gathering instance information for "asmid1.asmid1.us.oracle.com" from host
"asmid1.us.oracle.com"
    Gathering instance information for "asmid2.asmid2.us.oracle.com" from host
"asmid2.us.oracle.com"
The topology has been discovered. A topology.xml file has been written to each
home in the topology.
ASGCTL>

For Application Server 10.1.3.x releases:
ASGCTL> connect asg prodinfra oc4jadmin/<adminpwd>
Successfully connected to prodinfra:7890
ASGCTL> discover topology within farm
Warning: If OID is part of your environment, you should use it for discovery
Discovering topology on host "infra" with IP address "a.b.c.d" asmid:7890
Discovering instances within the topology using OPMN
Gathering instance information for "1013inst.asmid.oracle.com" from host
"asmid.us.oracle.com"
The topology has been discovered. A topology.xml file has been written to each
home in the topology.
ASGCTL>
```

You can also use the `add instance` command to add the standalone database instances to the Disaster Recovery topology as follows:

```
ASGCTL> add instance asdb on vhostdr.oracle.com
```

After the production topology is known by Oracle Application Server Guard for a production site, you can execute any one of the subsequent commands to perform a subsequent `asgctl` operation that involves the standby site. See discover topology for more information.

#### 2.11.1.4 Creating and Executing an asgctl Script

To create a script containing a sequence of asgctl command names and their arguments, open an edit session with your favorite editor, enter the asgctl commands in the proper sequence according to the operations you want to perform, save the script file, then execute the script when you invoke asgctl as shown in the following command:

```
> ASGCTL @myasgctlscript.txt
```

See the set echo command for an example of a script containing a series of asgctl commands.

You can also set the noprompt state for use in executing commands in an asgctl script in which all interactive prompts are later ignored. See the asgctl set noprompt command for more information.

### 2.11.2 Periodic Scheduling of Oracle Application Server Guard asgctl Scripts

For Oracle Application Server Guard operations that you want to run periodically, such as a periodic sync topology operation to keep the standby topology synchronized with the primary topology, you can automate the periodic running of an Oracle Application Server Guard asgctl script.

On UNIX systems, you can set up a cron job to run the asgctl script. Copy your asgctl script into the appropriate `/etc` subdirectory `cron.hourly`, `cron.daily`, `cron.weekly`, or `cron.monthly`. It will run either hourly, daily, weekly, or monthly, depending on the name of the subdirectory in which you choose to place your script. Or you can edit a crontab and create an entry that will be specific for the time on which you want to run the asgctl script. See the one or two manpages on cron and crontab for more information.

On Windows systems, you can use the task scheduler or scheduled tasks from the **Control Panel** to choose the time to run the asgctl script, daily, weekly, monthly, or at specific times. You can also purchase additional scheduler software with more options from a third party and then set the time and frequency to run the asgctl script. See the Windows operating system help for more information.

### 2.11.3 Submitting Oracle Application Server Guard Jobs to the Enterprise Manager Job System

You can use the Oracle Enterprise Manager Job System to automate the execution of any asgctl script to be run at a specified time interval or at a specified time and date, or both, in addition to setting other custom settings. To do this, access the Oracle Enterprise Manager **Job Activity** page and create your own host command job to execute your asgctl script, which is called a job task. Your job task (script) will invoke asgctl to run the asgctl commands in the order in which they are listed. After you create your Oracle Application Server Guard job, save it to the Oracle Enterprise Manager Job Library, which is a repository for frequently used jobs, where it can be executed based on the custom settings and time specifications you selected. See the Oracle Enterprise Manager online help and *Oracle Enterprise Manager Concepts* for more information.

# 3

# Configuring OracleAS Disaster Recovery

This chapter provides the following following sections about how to configure OracleAS Disaster Recovery with or without using RAC databases in your topology:

- Section 3.1, "Configuring OracleAS Disaster Recovery Without Real Application Clusters Databases"
- Section 3.2, "Using Oracle Real Application Clusters Database with OracleAS Disaster Recovery"

## 3.1 Configuring OracleAS Disaster Recovery Without Real Application Clusters Databases

This section describes how to set up OracleAS Disaster Recovery where neither the primary site nor the standby site uses an Oracle Real Application Clusters database to store the Metadata Repository schemas. This section shows how to use the asgctl `create standby database` command, which creates the database at the standby site and configures Oracle Data Guard to ship archive logs from the database at the production site to the database at the standby site.

The next section, Section 3.2, "Using Oracle Real Application Clusters Database with OracleAS Disaster Recovery", describes OracleAS Disaster Recovery topologies with Real Application Clusters databases.

This section contains the following subsections:

- Section 3.1.1, "Assumptions"
- Section 3.1.2, "Configuration Procedure"
- Section 3.1.3, "Switchover Procedure"
- Section 3.1.4, "Switchback Procedure"
- Section 3.1.5, "Failover Procedure"

### 3.1.1 Assumptions

The following sections include steps that demonstrate how to perform certain tasks:

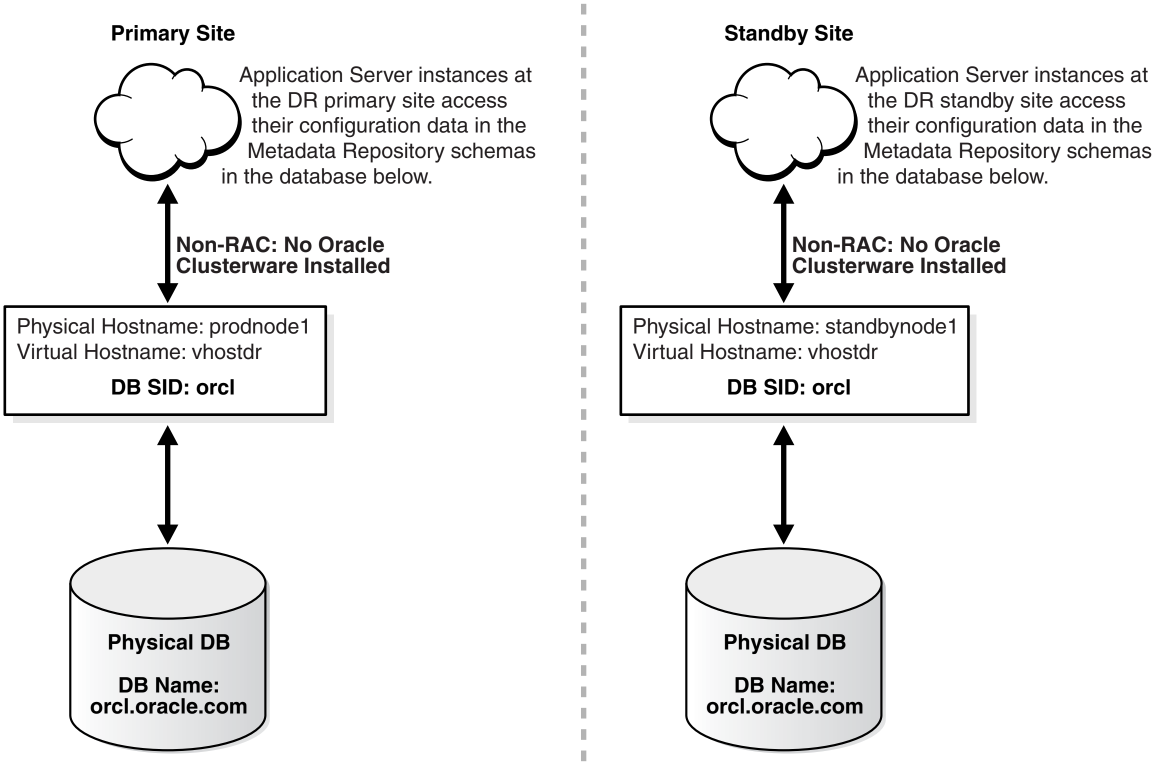
- Section 3.1.2, "Configuration Procedure" shows how to configure an OracleAS Disaster Recovery topology where neither the primary or standby site uses a Real Application Clusters database
- Section 3.1.3, "Switchover Procedure" shows how to perform a switchover procedure for this topology

- Section 3.1.4, "Switchback Procedure" shows how to perform a switchback procedure for this topology
- Section 3.1.5, "Failover Procedure" shows how to perform a failover procedure for this topology

Note the following assumptions for this topology:

- Figure 3–1 shows the host and database names used in the Disaster Recovery topology described in this section. Note that neither the production site or the standby site uses a Real Application Clusters database to store the Metadata Repository schemas for the Oracle Application Server instances in the topology. Because one to many Oracle Application Server instances could exist in this topology, the diagram does not depict any Oracle Application Server instances at the production site or standby site but instead states that they access their configuration data in the Metadata Repository schemas in the database at the production site and standby site, respectively.

***Figure 3–1 OracleAS Disaster Recovery Topology - non-Real Application Clusters Database at the Production Site and the Standby Site***


Primary Site
Application Server instances at the DR primary site access their configuration data in the Metadata Repository schemas in the database below.
Non-RAC: No Oracle Clusterware Installed
Physical Hostname: prodnode1
Virtual Hostname: vhostdr
DB SID: orcl
Physical DB
DB Name: orcl.oracle.com
Standby Site
Application Server instances at the DR standby site access their configuration data in the Metadata Repository schemas in the database below.
Non-RAC: No Oracle Clusterware Installed
Physical Hostname: standbynode1
Virtual Hostname: vhostdr
DB SID: orcl
Physical DB
DB Name: orcl.oracle.com

- Table 3–1 shows the host and database names that will be used in the steps in this section.

***Table 3–1 Host and Database Names on the Primary and Standby Sites***

| | Primary Site | Standby Site |
|---|---|---|
| Physical hostnames | prodnode1 | standbynode1 |
| Virtual hostnames | vhostdr | vhostdr |

*Table 3–1 (Cont.) Host and Database Names on the Primary and Standby Sites*

| | Primary Site | Standby Site |
|---|---|---|
| Database name | orcl.oracle.com | orcl.oracle.com |
| Database SID | orcl | orcl |

- The vhostdr virtual hostname should be mapped to the system's IP address. You set up this mapping in each system's `hosts` file. For example:

  On the primary site, edit `%SystemRoot%\system32\drivers\etc\hosts` (Windows) or `/etc/hosts` (UNIX) to add an entry similar to the following:

  ```
  ip_address  prodnode1.domain.com  prodnode1  vhostdr.domain.com  vhostdr
  ```

  On the standby site, edit the `hosts` file to add an entry similar to the following:

  ```
  ip_address  standbynode1.domain.com  standbynode1  vhostdr.domain.com  vhostdr
  ```

- Before proceeding to Section 3.1.2, "Configuration Procedure," do the following:

  1. On host prodnode1 at the primary site, install an OracleAS Infrastructure in its own Oracle home.

  2. On host standbynode1 at the standby site, install the same OracleAS Infrastructure in the equivalent Oracle home directory.

  3. On host prodnode1 at the primary site, create a database in a new Oracle home (*not* the Oracle AS Infrastructure home). Then use the Oracle Application Server Metadata Repository Creation Assistant to store the Metadata Repository schemas required by the OracleAS Infrastructure in that database. Refer to *Oracle Application Server Metadata Repository Creation Assistant User's Guide* for more information on using the OracleAS Metadata Repository Creation Assistant to load the Metadata Repository schemas into a database.

  4. On host standbynode1 at the standby site, install *only* the Oracle Database software in the equivalent database home.

  > **Note:** During the Oracle Database software installation on standbynode1, do *not* create a database in the home. In Section 3.1.2, "Configuration Procedure," you will use the `create standby database` command to create the standby database in the standby site home and to configure Oracle Data Guard so that it will ship archive logs from the primary database at the production site to the standby database at the standby site.
  >
  > The `create standby database` command does not work properly if a database exists in the standby site home when the command is executed. Refer to Appendix A.1.22, "Database Already Exists Errors During Create Standby" if a database already exists in the home on the standby site peer host.

- You must install the standalone version of Oracle Application Server Guard 10.1.2.3 into its own Oracle home on the database host. The standalone version of Oracle Application Server Guard 10.1.2.3 can be found on Oracle Technology Network at:

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

For instructions on how to run the standalone Oracle Application Server Guard installer, see the "Installing in High Availability Environments" chapter in the *Oracle Application Server Installation Guide* for your platform.

- The asgctl `set primary database` command must be issued for both the primary and standby site within asgctl to define the service name mapping within Oracle Application Server Guard before attempting an asgctl `create standby database` command
- The `create standby database` command is designed to automate the creation of simple standby databases and to set up the Oracle Data Guard configuration between the primary and standby databases. It does not support some database options, such as the OMF (Oracle Managed Files) or ASM (Automatic Storage Management) storage options. If you plan to use the `create standby database` command to create a database at the standby site, create the database instance on the primary site without specifying the OMF or ASM storage options.
- Databases that use the OMF or ASM storage options can be included in a Disaster Recovery topology, but you cannot use the `create standby database` command to create a standby database that uses these storage options, and you must use Oracle Data Guard to set up the Oracle Data Guard configuration for the primary and standby databases. See Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology" for more information on configuring Oracle Data Guard for databases that use the OMF or ASM storage options.
- The `create standby database` command must be initiated by ASG clients from the source primary node where the database for the primary site resides.
- If you update the AS Recovery Manager's config_misc_files.inp to include other configurations files as part of the backup procedure, special consideration has to be made for corresponding changes at a Disaster Recovery standby site. Changes to this file could be made manually or indirectly through the MDS repository LifeCycle Tool backup option. With each change, files specified in config_misc_files.inp and on a corresponding restore, are restored. However, the config_misc_files.inp is versioned and restored in the form config_misc_files.inp config_misc_files.inp_<backup_time_stamp>. For a Disaster Recovery standby site, the latest version of this file has to be used after a failover or switchover operation. To ensure this configuration change is made, if a timestamped version exists, the latest version of ORACLE_HOME/backup_restore/config/config_misc_files.inp_<backup_time_stamp> in each Oracle home has to be copied to replace the current version of ORACLE_HOME/backup_restore/config/config_misc_files.inp. This must be done after any Oracle Application Server Guard `instantiate`, `failover`, or `switchover` operation.
- If you have multiple nodes in the topology on the primary site, then the name of each Oracle Application Server instance must be unique across all the homes on all the nodes in the primary site.

#### 3.1.1.1 Special Considerations for Multiple Databases in a Topology

If including multiple databases in a single Disaster Recovery topology, consider the following:

- The `set primary database` command must be invoked individually for each database, before invoking the `create standby database` command. The `create standby database` must be run from the system where the primary database resides.

Example:

```
ASGCTL> set primary database sys/<pass>@orcl1
ASGCTL> create standby database orcl1 on standbynode1
ASGCTL> set primary database sys/<pass>@orcl2
ASGCTL> create standby database orcl2 on standbynode1
```

- The `set primary database` command must be invoked for each database included in the Disaster Recovery topology for all other Disaster Recovery operations, such as `instantiate`, `switchover`, `or sync` operations. For example:

  ```
  ASGCTL> set primary database sys/<pass>@orcl1
  ASGCTL> set primary database sys/<pass>@orcl2
  ASGCTL> sync topology to standbynode1
  ```

- If databases from 2 different Oracle homes, for example, `Oracle 10gR1 db` in `OHOME1` and `Oracle 10gR2 db` in `OHOME2`, on the same system are included in a Disaster Recovery topology, Oracle recommends running the database listener from the Oracle home having the higher database version, for this example, `OHOME2`.

### 3.1.2 Configuration Procedure

Perform the following steps to set up a OracleAS Disaster Recovery topology where neither the primary nor the standby site uses a Real Application Clusters database.

1. Start up the database on prodnode1.

   ```
   > DBHOME/bin/sqlplus / as sysdba
   SQL> startup
   ```

2. Create the standby database on the standby node by running the following ASGCTL commands on prodnode1.

   > **Note:** The `create standby database` command does not work properly if a database exists in the database home on the standby site peer host when the command is executed. If a database already exists in the home on the standby site peer host, follow the instructions in Appendix A.1.22, "Database Already Exists Errors During Create Standby" before executing the `create standby database` command in the following steps.

   ```
   For 10.1.2.x and 10.1.4.x releases:
   ASGCTL> connect asg prodnode1 ias_admin/<adminpwd>

   For 10.1.3.x releases:
   ASGCTL> connect asg prodnode1 oc4jadmin/<adminpwd>

   ASGCTL> set trace on all

   ASGCTL> add instance orcl on vhostdr.oracle.com

   The output from this dump topology command should list the database instance
   added in the previous command:
   ASGCTL> dump topology

   The output from this verify topology command should indicate that an HA
   ```

```
directory exists for the database instance added with the add instance command:
ASGCTL> verify topology

ASGCTL> set noprompt
ASGCTL> set primary database sys/<passwd>@orcl

ASGCTL> create standby database orcl on standbynode1

The output from this verify topology command should indicate that an HA
directory exists on both the primary site and standby site for the database
instance added with the add instance command:
ASGCTL> verify topology with standbynode1
ASGCTL> instantiate topology to standbynode1
ASGCTL> sync topology to standbynode1
```

### 3.1.3 Switchover Procedure

For scheduled outages of the primary site, you run the ASGCTL `switchover topology` command to switch over to the standby site. If you have an unscheduled outage, you must perform failover steps as described in Section 3.1.5, "Failover Procedure."

To switchover from the production site to the standby site, perform the following steps:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.
2. On prodnode1, invoke the Oracle Application Server Guard client command-line utility asgctl (asgctl.sh is located in `<ORACLE_HOME>/dsa/bin` on UNIX systems and asgctl.bat is located in `<ORACLE_HOME>\dsa\bin` on Windows systems) and then connect to the Oracle Application Server Guard server:

```
For 10.1.2.x and 10.1.4.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg ias_admin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.3.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.2.x and 10.1.4.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg ias_admin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.3.x releases on Windows:
> asgctl.bat
```

```
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodnode1:7890
```

3. Use the `verify topology` command to confirm that the production topology is running and the configuration is valid:

   ```
   ASGCTL> verify topology
   ```

4. Use the `verify topology` and specify the standby Infrastructure host (the standby Infrastructure host's network hostname is standbynode1) to confirm that the standby topology is consistent with the production site and meets Disaster Recovery requirements:

   ```
   ASGCTL> verify topology with standbynode1
   ```

5. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information:

   ```
   ASGCTL> set primary database sys/<passwd>@orcl
   ```

6. Switchover the topology to the standby site. If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file:

   ```
   ASGCTL> switchover topology to standbynode1
   ```

   **Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

7. Disconnect from the *old* production site Oracle Application Server Guard server:

   ```
   ASGCTL> disconnect
   ```

8. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

9. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

### 3.1.4 Switchback Procedure

When the scheduled outage is over, you switch back from the standby site to the primary site.

Run the following commands to switch back to the primary site:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. On standbynode1 (the current primary site), invoke the Oracle Application Server Guard client command-line utility asgctl (asgctl.sh is located in `<ORACLE_`

HOME>/dsa/bin on UNIX systems and asgctl.bat is located in <ORACLE_HOME>\dsa\bin on Windows systems) and then connect to the Oracle Application Server Guard server:

```
For 10.1.2.x and 10.1.4.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
Successfully connected to standbynode1:7890

For 10.1.3.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
Successfully connected to standbynode1:7890

For 10.1.2.x and 10.1.4.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
Successfully connected to standbynode1:7890

For 10.1.3.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
Successfully connected to standbynode1:7890
```

3. Use the `verify topology` command to confirm that the production topology is running and the configuration is valid:

   ```
   ASGCTL> verify topology
   ```

4. Use the `verify topology` and specify the standby Infrastructure host (the standby Infrastructure host's network hostname is prodnode1, the original primary host) to confirm that the standby topology is consistent with the production site and meets Disaster Recovery requirements:

   ```
   ASGCTL> verify topology with prodnode1
   ```

5. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information:

   ```
   ASGCTL> set primary database sys/<passwd>@orcl
   ```

6. Switchover the topology to the standby site (the original primary site). If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file:

   ```
   ASGCTL> switchover topology to prodnode1
   ```

> **Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

7. Disconnect from the *old* production site Oracle Application Server Guard server:

   ```
   ASGCTL> disconnect
   ```

8. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

9. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

### 3.1.5 Failover Procedure

You perform the failover steps to fail over to the standby site when the primary site fails unexpectedly. For scheduled outages, you should perform the steps in Section 3.1.3, "Switchover Procedure" instead.

To fail over to the standby site, run these commands on the standby site and activate it as the new primary site.

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. Connect to the Oracle Application Server Guard server on the standby site. The network name is standbynode1.

   ```
   For 10.1.2.x and 10.1.4.x releases:
   ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
   Successfully connected to standbynode1:7890

   For 10.1.3.x releases:
   ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
   Successfully connected to standbynode1:7890
   ```

3. Specify that the primary OracleAS Metadata Repository database on the standby site is now identified as the new primary database on this new production site. The keyword **new** is in bold text in the following example to indicate its importance as a keyword. If you have multiple OracleAS Metadata Repositories in your topology, you must authenticate each one using the `set new primary database` command.

   ```
   ASGCTL> set new primary database sys/<passwd>@orcl
   ```

4. Execute the asgctl `failover` command.

   ```
   ASGCTL> set trace on all
   ASGCTL> failover
   ASGCTL> disconnect
   ```

5. Discover the topology. You must perform this operation to create a new topology file for this production site.

   For 10.1.2.x and 10.1.4.x environments, discover the topology as follows:

   ```
   ASGCTL> discover topology oidpassword=oidpwd
   ```

   For 10.1.3.x environments, discover the topology as follows:

   ```
   ASGCTL> discover topology within farm
   ```

You can also use the `add instance` command to add the standalone database instances to the Disaster Recovery topology as follows:

```
ASGCTL> add instance asdb on vhostdr.oracle.com
```

6. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."
7. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

When you must perform a failover, you should also decide whether to set up a new standby site or not. If you decide to set up a new standby site, then after the failover operation has completed, perform an asgctl `create standby database` command to create the standby database on the remote host and then perform an asgctl `instantiate topology` operation.

## 3.2 Using Oracle Real Application Clusters Database with OracleAS Disaster Recovery

This section describes how to configure your OracleAS Disaster Recovery topology if you are using a Real Application Clusters database for your OracleAS Metadata Repository. You can use Real Application Clusters database on both your primary and standby sites, or just on the primary site (the standby site uses a non-Real Application Clusters database). The following subsections cover these cases:

- Section 3.2.1, "Configuring OracleAS Disaster Recovery Where Both the Primary and Standby Sites Use Oracle Real Application Clusters Databases"
- Section 3.2.2, "Configuring OracleAS Disaster Recovery Where Only the Primary Site Uses Oracle Real Application Clusters Database (Standby Site Uses a Non-Real Application Clusters Database)"

**Note:** The Disaster Recovery RAC support described in Section 3.2.1, "Configuring OracleAS Disaster Recovery Where Both the Primary and Standby Sites Use Oracle Real Application Clusters Databases" and Section 3.2.2, "Configuring OracleAS Disaster Recovery Where Only the Primary Site Uses Oracle Real Application Clusters Database (Standby Site Uses a Non-Real Application Clusters Database)" is available in Application Server releases 10.1.2.3 and 10.1.3.3.

### 3.2.1 Configuring OracleAS Disaster Recovery Where Both the Primary and Standby Sites Use Oracle Real Application Clusters Databases

This section describes how to set up OracleAS Disaster Recovery in a topology where both the primary and standby sites use an Oracle Real Application Clusters database to store the OracleAS Metadata Repository schemas. This section shows how to use the asgctl `create standby database` command, which creates the database at the standby site and configures Oracle Data Guard to ship archive logs from the database at the production site to the database at the standby site.

This section contains the following subsections:

- Section 3.2.1.1, "Assumptions"

- Section 3.2.1.2, "Configuration Procedure"
- Section 3.2.1.3, "Switchover Procedure"
- Section 3.2.1.4, "Switchback Procedure (for Switching Back to the Primary Site)"
- Section 3.2.1.5, "Failover Procedure"

#### 3.2.1.1 Assumptions

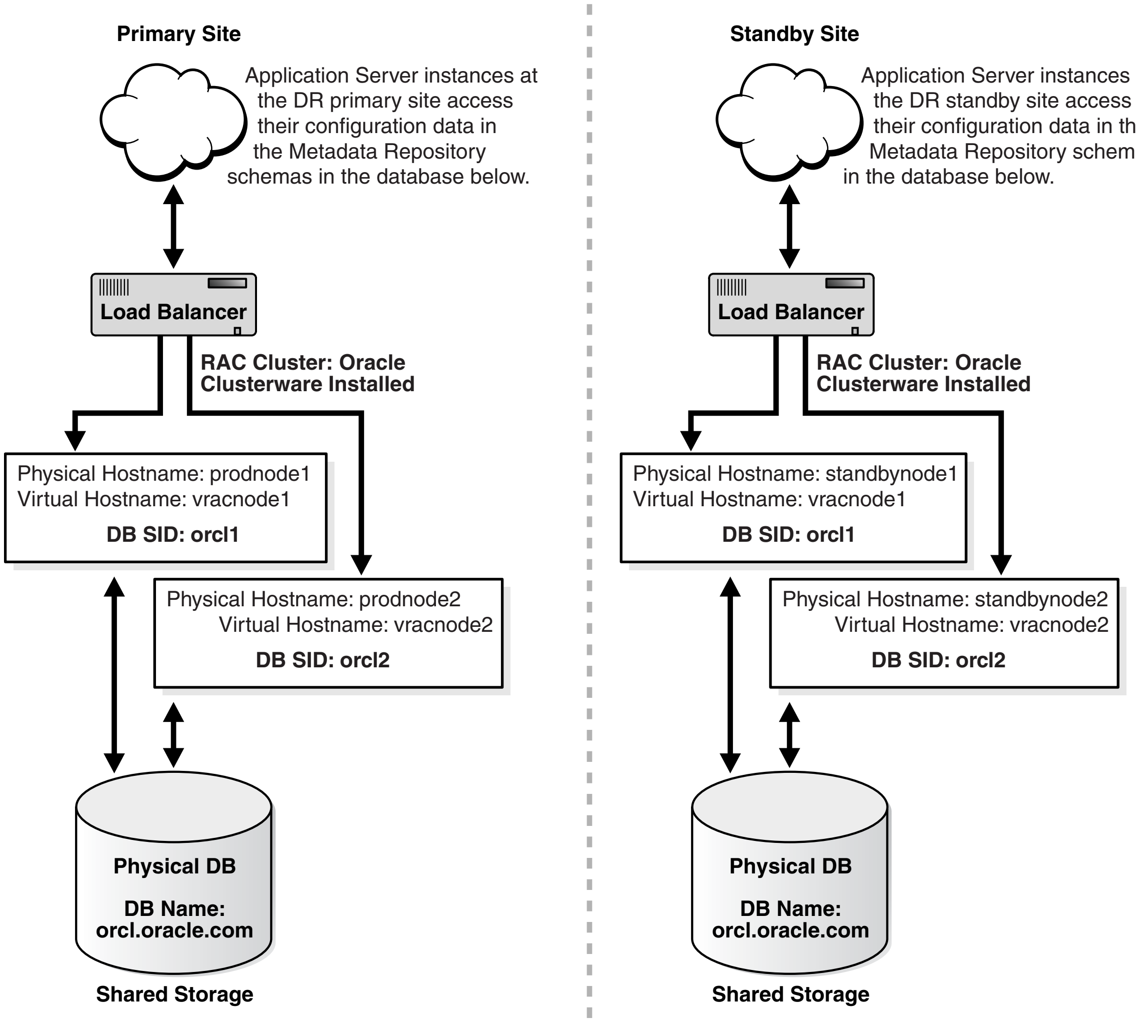
The following sections include steps that demonstrate how to perform certain tasks:

- Section 3.2.1.2, "Configuration Procedure" shows how to configure an OracleAS Disaster Recovery topology where both the primary site and standby site uses a Real Application Clusters database
- Section 3.2.1.3, "Switchover Procedure" shows how to perform a switchover procedure for this topology
- Section 3.2.1.4, "Switchback Procedure (for Switching Back to the Primary Site)" shows how to perform a switchback procedure for this topology
- Section 3.2.1.5, "Failover Procedure" shows how to perform a failover procedure for this topology

Note the following assumptions for this topology:

- Oracle Real Application Clusters (RAC) software and Oracle Clusterware software has been installed on the primary and standby sites.
- Figure 3–2 shows the host and database names used in the Disaster Recovery topology described in this section. Note that the production site and the standby site use a Real Application Clusters database to store the Metadata Repository schemas for the Oracle Application Server instances in the topology. Because one to many Oracle Application Server instances could exist in this topology, the diagram does not depict any Oracle Application Server instances at the production site or standby site but instead states that they access their configuration data in the Metadata Repository schemas in the database at the production site and standby site, respectively.

***Figure 3–2 OracleAS Disaster Recovery Topology - Real Application Clusters Database at Production Site and Standby Site***


Primary Site
Application Server instances at the DR primary site access their configuration data in the Metadata Repository schemas in the database below.
Load Balancer
RAC Cluster: Oracle Clusterware Installed
Physical Hostname: prodnode1
Virtual Hostname: vracnode1
DB SID: orcl1
Physical Hostname: prodnode2
Virtual Hostname: vracnode2
DB SID: orcl2
Physical DB
DB Name: orcl.oracle.com
Shared Storage
Standby Site
Application Server instances at the DR standby site access their configuration data in the Metadata Repository schemas in the database below.
Load Balancer
RAC Cluster: Oracle Clusterware Installed
Physical Hostname: standbynode1
Virtual Hostname: vracnode1
DB SID: orcl1
Physical Hostname: standbynode2
Virtual Hostname: vracnode2
DB SID: orcl2
Physical DB
DB Name: orcl.oracle.com
Shared Storage

- Table 3–2 shows the host and database names that will be used in the steps in the following section. The procedure assumes a two-node Real Application Clusters on each site.

***Table 3–2 Host and Database Names on the Primary and Standby Sites***

| | Primary Site | Standby Site |
|---|---|---|
| Physical hostnames | prodnode1, prodnode2 | standbynode1, standbynode2 |
| Virtual hostnames | vracnode1, vracnode2 | vracnode1, vracnode2 |
| Database name | orcl.oracle.com | orcl.oracle.com |
| Database SID | orcl1 on prodnode1<br>orcl2 on prodnode2 | orcl1 on standbynode1<br>orcl2 on standbynode2 |

- The vracnode1 virtual hostname should be mapped to the system's IP address. You set up this mapping in each system's `hosts` file. For example:

On the primary site, edit `%SystemRoot%\system32\drivers\etc\hosts` (Windows) or `/etc/hosts` (UNIX) to add an entry similar to the following:

```
ip_address  prodnode1.domain.com  prodnode1  vracnode1.domain.com  vracnode1
```

On the standby site, edit the `hosts` file to add an entry similar to the following:

```
ip_address  standbynode1.domain.com  standbynode1  vracnode1.domain.com
 vracnode1
```

- Before proceeding to Section 3.2.1.2, "Configuration Procedure" do the following:

  1. On host prodnode1 at the primary site, install an OracleAS Infrastructure in its own Oracle home.
  2. On host standbynode1 at the standby site, install the same OracleAS Infrastructure in the equivalent Oracle home directory.
  3. On host prodnode1 at the primary site, create a database in a new Oracle home (*not* the Oracle AS Infrastructure home). Then use the Oracle Application Server Metadata Repository Creation Assistant to store the Metadata Repository schemas required by the OracleAS Infrastructure in that database. Refer to *Oracle Application Server Metadata Repository Creation Assistant User's Guide* for more information on using the OracleAS Metadata Repository Creation Assistant to load the Metadata Repository schemas into a database.
  4. On host standbynode1 at the standby site, install *only* the Oracle Database software in the equivalent database home.

  > **Note:** During the Oracle Database software installation on standbynode1, do *not* create a database in the home. In Section 3.2.1.2, "Configuration Procedure" you will use the `create standby database` command to create the standby database in the standby site home and to configure Oracle Data Guard so that it will ship archive logs from the primary database at the production site to the standby database at the standby site.
  >
  > The `create standby database` command does not work properly if a database exists in the standby site home when the command is executed. Refer to Appendix A.1.22, "Database Already Exists Errors During Create Standby" if a database already exists in the home on the standby site peer host.

- You must install the standalone version of Oracle Application Server Guard 10.1.2.3 into its own Oracle home on the database host. The standalone version of Oracle Application Server Guard 10.1.2.3 can be found on Oracle Technology Network at:

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

  For instructions on how to run the standalone Oracle Application Server Guard installer, see the "Installing in High Availability Environments" chapter in the *Oracle Application Server Installation Guide* for your platform.

- The `create standby database` command is designed to automate the creation of simple standby databases and to set up the Oracle Data Guard configuration between the primary and standby databases. It does not support some database options, such as the OMF (Oracle Managed Files) or ASM

(Automatic Storage Management) storage options. If you plan to use the create standby database command to create a database at the standby site, create the database instance on the primary site without specifying the OMF or ASM storage options.

- Databases that use the OMF or ASM storage options can be included in a Disaster Recovery topology, but you cannot use the create standby database command to create a standby database that uses these storage options, and you must use Oracle Data Guard to set up the Oracle Data Guard configuration for the primary and standby databases. See Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology" for more information on configuring Oracle Data Guard for databases that use the OMF or ASM storage options.
- The create standby database command must be initiated by ASG clients from the source primary node where the database for the primary site resides.
- The DBName (without domain) and DBSID must be the same when creating the database for a Disaster Recovery setup on primary or standby sites. When you create a new database instance using the DBCA, the SID defaults to the database name. If you enter a name in the SID field other than the database name, and then later add this database to the Disaster Recovery topology, the instance name in the topology added will be empty.
- To create a orapwd based password file in the shared location for use by other RAC instances, you must copy over the orapw file to the shared location from the instance on the standby where the database was created after executing the create standby database command. Also after copying the file to the shared location, it should be symlinked to this file from the local disk where it was created.
- If a sync topology command in a RAC-RAC Linux environment fails and you receive missing archive logs errors, ping the standby node using tnsping. If you are unable to ping the standby node, stop and restart the listener for that node and retry the tnsping.
- If you update the AS Recovery Manager's config_misc_files.inp to include other configurations files as part of the backup procedure, special consideration has to be made for corresponding changes at a Disaster Recovery standby site. Changes to this file could be made manually or indirectly through the MDS repository LifeCycle Tool backup option. With each change, files specified in config_misc_files.inp and on a corresponding restore, are restored. However, the config_misc_files.inp is versioned and restored in the form config_misc_files.inp config_misc_files.inp_<backup_time_stamp>. For a Disaster Recovery standby site, the latest version of this file has to be used after a failover or switchover operation. To ensure this configuration change is made, if a timestamped version exists, the latest version of ORACLE_HOME/backup_restore/config/config_misc_files.inp_<backup_time_stamp> in each Oracle home has to be copied to replace the current version of ORACLE_HOME/backup_restore/config/config_misc_files.inp. This must be done after any Oracle Application Server Guard instantiate, failover, or switchover operation.
- If you have multiple nodes in the topology on the primary site, then the name of each Oracle Application Server instance must be unique across all the homes on all the nodes in the primary site.

#### 3.2.1.2 Configuration Procedure

Perform the following steps to configure OracleAS Disaster Recovery topologies where the primary and standby sites use RAC databases.

1. Make sure that the database on prodnode1 is running.
2. Create the standby database on the standby node by running the following ASGCTL commands on prodnode1.

   > **Note:** The `create standby database` command does not work properly if a database exists in the database home on the standby site peer host when the command is executed. If a database already exists in the home on the standby site peer host, follow the instructions in Appendix A.1.22, "Database Already Exists Errors During Create Standby" before executing the `create standby database` command in the following steps.

   Here are some notes on the commands in the following section:

   - On UNIX, the `add instance` command uses `orcl` (the database name) to locate the oratab entry.

     On Windows, it uses `orcl1` (the database SID) to locate the registry entry.
   - The `set primary database` and `create standby database` commands use `orcl` (the database name) on UNIX, but on Windows they use `orcl1` (the database SID).

   ```
   For 10.1.2.x and 10.1.4.x releases:
   ASGCTL> connect asg prodnode1 ias_admin/<adminpwd>

   For 10.1.3.x releases:
   ASGCTL> connect asg prodnode1 oc4jadmin/<adminpwd>

   ASGCTL> set trace on all

   UNIX only: ASGCTL> add instance orcl on vracnode1.oracle.com
   Windows only: ASGCTL> add instance orcl1 on vracnode1.oracle.com

   The output from this dump topology command should list the database instance
   added in the previous command:
   ASGCTL> dump topology

   The output from this verify topology command should indicate that an HA
   directory exists for the database instance added with the add instance command:
   ASGCTL> verify topology

   ASGCTL> set noprompt

   UNIX only: ASGCTL> set primary database sys/<passwd>@orcl
   Windows only: ASGCTL> set primary database sys/<passwd>@orcl1

   UNIX only: ASGCTL> create standby database orcl on standbynode1
   Windows only: ASGCTL> create standby database orcl1 on standbynode1

   The output from this verify topology command should indicate that an HA
   directory exists on both the primary site and standby site for the database
   instance added with the add instance command:
   ASGCTL> verify topology with standbynode1
   ```

```
ASGCTL> instantiate topology to standbynode1
```

3. On standbynode1, modify parameters in the `initorcl1.ora` file.

   a. On standbynode1, make a backup copy of the file `DBHOME/dbs/initorcl1.ora` (UNIX) or `DBHOME\database\initorcl1.ora` (Windows). You will be editing the file in the next step.

   b. Verify that these parameters in `DBHOME/dbs/initorcl1.ora` (UNIX) or `DBHOME\database\initorcl1.ora` (Windows) are set to valid values:

   ```
   *.cluster_database_instances=2
   *.cluster_database=TRUE
   *.remote_listener='LISTENERS_ORCL'
   ```

   c. Copy `initorcl1.ora` from standbynode1 to the corresponding directory on standbynode2 (`DBHOME/dbs` on UNIX, `DBHOME\database` on Windows).

   d. On standbynode2, rename the file to `initorcl2.ora`.

   e. On standbynode2, update the `initorcl2.ora` file to replace any instance-specific parameters. For example, you would change these lines:

   ```
   *.service_names=orcl1
   *.instance_name=orcl1
   ```

   to:

   ```
   *.service_names=orcl2
   *.instance_name=orcl2
   ```

   f. Only on UNIX platforms, create an admin directory under the default ADMIN directory  of the database ($ADMINDIR/<dbname>/admin) on standbynode2. If the directory already exists, please skip this step:

   ```
   mkdir $ohome/admin/<dbname>/admin
   ```

4. Propagate orcl_remote1 or orcl1_remote1 entries from standbynode1 to other RAC nodes on the standby site.

   a. Copy the orcl_remote1 (UNIX) or orcl1_remote1 (Windows) entries in tnsnames.ora on standbynode1 to all the other RAC nodes on the standby site.

   On UNIX, the entry uses the database name (`orcl`), but on Windows it uses the database SID (`orcl1`). A "`_remote`*<n>*" is appended to the name of the entry, where *<n>* is a number.

   In some cases, the *<n>* number will advance, and the `_remote`*<n>* entry specified in the `SERVICE` attribute of the `LOG_ARCHIVE_DEST_`*<n>* parameter must be propagated as well.

   b. On standbynode2, restart the listener using Oracle Clusterware:

   ```
   > CRSHOME/bin/crs_stop  ora.standbynode2.LISTENER_STANDBYNODE2.lsnr
   > CRSHOME/bin/crs_start ora.standbynode2.LISTENER_STANDBYNODE2.lsnr
   ```

   c. Make sure that the standby database mentioned in the remote entry is pingable using TNS.

   ```
   UNIX only: > tnsping orcl_remote1
   Windows only: > tnsping orcl1_remote1
   ```

5. On standbynode2, start up the database, create an spfile, and shut down the database.

```
SQL> startup;

UNIX only: SQL> create spfile='<ORADATASHAREDLOCATION>/orcl/spfileorcl.ora'
                              from pfile='<DBHOME>/dbs/initORCL2.ora';
Windows only: SQL> create spfile='<ORADATASHAREDLOCATION>\orcl\spfileorcl.ora'
                              from pfile='<DBHOME>/database/initORCL2.ora';

SQL> shutdown immediate;
```

6. Backup initORCL1.ora and initORCL2.ora on standbynode1 and standbynode2.

   On standbynode1, modify initORCL1.ora to include only the following line:

```
cat initORCL1.ora
SPFILE='<ORADATASHAREDLOCATION>/ORCL/spfileorcl.ora'
```

   On standbynode2, modify initORCL2.ora to include only the following line:

```
cat initORCL2.ora
SPFILE='<ORADATASHAREDLOCATION>/ORCL/spfileorcl.ora'
```

7. On prodnode1, run the asgctl `sync topology` command.

```
For 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg ias_admin/<adminpwd>

For 10.1.3.x releases:
ASGCTL> connect asg oc4jadmin/<adminpwd>

UNIX only: ASGCTL> set primary database sys/<passwd>@orcl
Windows only: ASGCTL> set primary database sys/<passwd>@orcl1

ASGCTL> sync topology to standbynode1
```

#### 3.2.1.3 Switchover Procedure

This section describes how to run the ASGCTL `switchover topology` command to switch from the primary site to the standby site to prepare for a scheduled outage of the primary site.

After the scheduled outage is over, you can switch back to the primary site. See Section 3.2.1.4, "Switchback Procedure (for Switching Back to the Primary Site)" for details.

For unscheduled outages, you should perform the steps in Section 3.2.1.5, "Failover Procedure" instead.

Procedure for switching over to the standby site for scheduled outages:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.
2. On prodnode1, invoke the Oracle Application Server Guard client command-line utility asgctl (asgctl.sh is located in `<ORACLE_HOME>/dsa/bin` on UNIX systems

and asgctl.bat is located in `<ORACLE_HOME>\dsa\bin` on Windows systems) and then connect to the Oracle Application Server Guard server:

```
For 10.1.2.x and 10.1.4.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg ias_admin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.3.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.2.x and 10.1.4.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg ias_admin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.3.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodnode1:7890
```

3. Use the `verify topology` command to confirm that the production topology is running and the configuration is valid:

```
ASGCTL> verify topology
```

4. Use the `verify topology` and specify the standby Infrastructure host (the standby Infrastructure host's network hostname is standbynode1) to confirm that the standby topology is consistent with the production site and meets Disaster Recovery requirements:

```
ASGCTL> verify topology with standbynode1
```

5. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information:

```
UNIX only: ASGCTL> set primary database sys/<paswd>@orcl
Windows only: ASGCTL> set primary database sys/<passwd>@orcl1
```

6. Switchover the topology to the standby site. If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file:

```
ASGCTL> switchover topology to standbynode1
```

**Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

7. Disconnect from the *old* production site Oracle Application Server Guard server:

   ```
   ASGCTL> disconnect
   ```

8. If the database uses temp files (for example, temp01.dbf), delete these files from <ORADATASHAREDLOCATION> on standbynode1.

9. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

10. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

#### 3.2.1.4 Switchback Procedure (for Switching Back to the Primary Site)

When the scheduled outage of the primary site is over, perform these steps to switch back to the primary site.

Run the following commands to switch back to the primary site:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. On standbynode1 (the current primary site), invoke the Oracle Application Server Guard client command-line utility asgctl (asgctl.sh is located in `<ORACLE_HOME>/dsa/bin` on UNIX systems and asgctl.bat is located in `<ORACLE_HOME>\dsa\bin` on Windows systems) and then connect to the Oracle Application Server Guard server:

   ```
   For 10.1.2.x and 10.1.4.x releases on UNIX:
   > asgctl.sh
   Application Server Guard: Release 10.1.2.0.2
   (c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
   ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
   Successfully connected to standbynode1:7890

   For 10.1.3.x releases on UNIX:
   > asgctl.sh
   Application Server Guard: Release 10.1.3.2.0
   (c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
   ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
   Successfully connected to standbynode1:7890

   For 10.1.2.x and 10.1.4.x releases on Windows:
   > asgctl.bat
   Application Server Guard: Release 10.1.2.0.2
   (c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
   ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
   Successfully connected to standbynode1:7890

   For 10.1.3.x releases on Windows:
   > asgctl.bat
   Application Server Guard: Release 10.1.3.2.0
   (c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
   ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
   Successfully connected to standbynode1:7890
   ```

3. Use the `verify topology` command to confirm that the production topology is running and the configuration is valid:

   ```
   ASGCTL> verify topology
   ```

4. Use the `verify topology` and specify the standby Infrastructure host (the standby Infrastructure host's network hostname is prodnode1, the original primary host) to confirm that the standby topology is consistent with the production site and meets Disaster Recovery requirements:

   ```
   ASGCTL> verify topology with prodnode1
   ```

5. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information:

   ```
   UNIX only: ASGCTL> set primary database sys/<passwd>@orcl
   Windows only: ASGCTL> set primary database sys/<passwd>@orcl1
   ```

6. Switchover the topology to the standby site (the original primary site). If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file:

   ```
   ASGCTL> switchover topology to prodnode1
   ```

   **Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

7. Disconnect from the *old* production site Oracle Application Server Guard server:

   ```
   ASGCTL> disconnect
   ```

8. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

9. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

#### 3.2.1.5 Failover Procedure

This section describes the steps for failing over to the standby site. Use these steps for unscheduled outages of the primary site. For scheduled outages, see the steps in Section 3.2.1.3, "Switchover Procedure".

To fail over to the standby site, run these commands on the standby site and activate it as the new primary site.

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. Connect to the Oracle Application Server Guard server on the standby site. The network name is standbynode1.

```
For 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>

For 10.1.3.x releases:
ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
```

3. Specify that the primary OracleAS Metadata Repository database on the standby site is now identified as the new primary database on this new production site. The keyword **new** is in bold text in the following example to indicate its importance as a keyword. If you have multiple OracleAS Metadata Repositories in your topology, you must authenticate each one using the `set new primary database` command.

   ```
   UNIX only: ASGCTL> set new primary database sys/<passwd>@orcl
   Windows only: ASGCTL> set new primary database sys/<passwd>@orcl1
   ```

4. Execute the asgctl `failover` command.

   ```
   ASGCTL> set trace on all
   ASGCTL> failover
   ASGCTL> disconnect
   ```

5. Discover the topology. You must perform this operation to create a new topology file for this production site.

   For 10.1.2.x and 10.1.4.x environments, discover the topology as follows:

   ```
   ASGCTL> discover topology oidpassword=oidpwd
   ```

   For 10.1.3.x environments, discover the topology as follows:

   ```
   ASGCTL> discover topology within farm
   ```

   You can also use the `add instance` command to add the standalone database instances to the Disaster Recovery topology as follows:

   ```
   ASGCTL> add instance asdb on vracnode1.oracle.com
   ```

6. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

7. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

When you must perform a failover, you should also decide whether to set up a new standby site or not. If you decide to set up a new standby site, then after the failover operation has completed, perform an asgctl `create standby database` command to create the standby database on the remote host and then perform an asgctl `instantiate topology` operation.

### 3.2.2 Configuring OracleAS Disaster Recovery Where Only the Primary Site Uses Oracle Real Application Clusters Database (Standby Site Uses a Non-Real Application Clusters Database)

This section describes how to set up OracleAS Disaster Recovery in a topology where the primary site uses an Oracle Real Application Clusters database to store the OracleAS Metadata Repository schemas, and the standby site uses a non-Real Application Clusters database to store the OracleAS Metadata Repository schemas. This section shows how to use the asgctl `create standby database` command, which creates the database at the standby site and configures Oracle Data Guard to ship archive logs from the database at the production site to the database at the standby site.

This section contains the following subsections:

- Section 3.2.2.1, "Assumptions"
- Section 3.2.2.2, "Configuration Procedure"
- Section 3.2.2.3, "Switchover Procedure"
- Section 3.2.2.4, "Switchback Procedure"
- Section 3.2.2.5, "Failover Procedure"

### 3.2.2.1 Assumptions

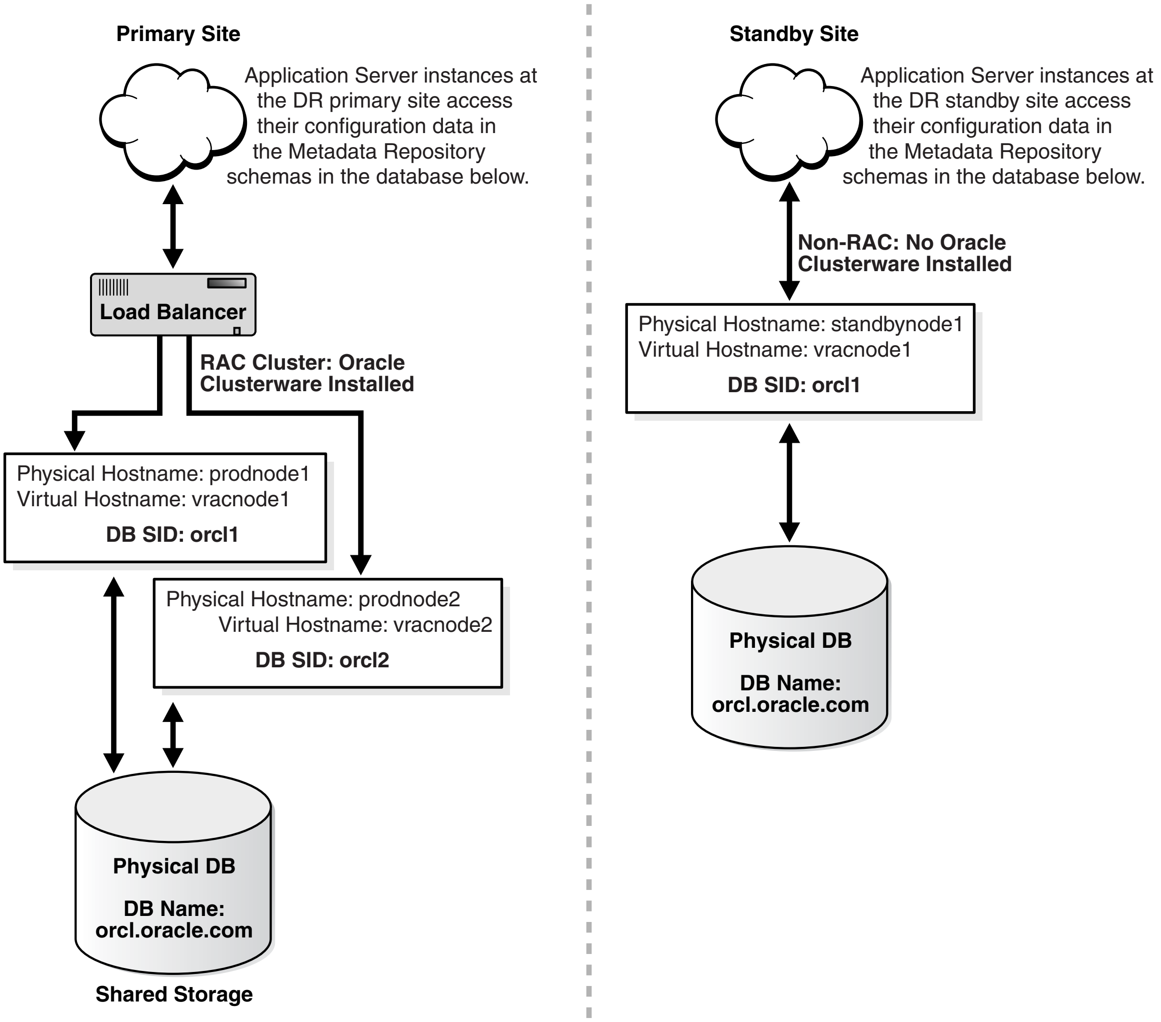
The following sections include steps that demonstrate how to perform certain tasks:

- Section 3.2.2.2, "Configuration Procedure" shows how to configure an OracleAS Disaster Recovery topology where the primary site uses an Oracle Real Application Clusters database and the standby site uses a non-Real Application Clusters database
- Section 3.2.2.3, "Switchover Procedure" shows how to perform a switchover procedure for this topology
- Section 3.2.2.4, "Switchback Procedure" shows how to perform a switchback procedure for this topology
- Section 3.2.2.5, "Failover Procedure" shows how to perform a failover procedure for this topology

Note the following assumptions for this topology:

- Oracle Real Application Clusters (RAC) software and Oracle Clusterware software has been installed on the primary site.
- Figure 3–3 shows the host and database names used in the Disaster Recovery topology described in this section. Note that the production site uses a RealApplication Clusters database and the standby site use a non-Real Application Clusters database to store the Metadata Repository schemas for the Oracle Application Server instances in the topology. Because one to many Oracle Application Server instances could exist in this topology, the diagram does not depict any Oracle Application Server instances at the production site or standby site but instead states that they access their configuration data in the Metadata Repository schemas in the database at the production site and standby site, respectively.

*Figure 3–3 OracleAS Disaster Recovery Topology - Real Application Clusters Database at Production Site and non-Real Application Clusters Database at Standby Site*


Primary Site
Application Server instances at the DR primary site access their configuration data in the Metadata Repository schemas in the database below.
Load Balancer
RAC Cluster: Oracle Clusterware Installed
Physical Hostname: prodnode1
Virtual Hostname: vracnode1
DB SID: orcl1
Physical Hostname: prodnode2
Virtual Hostname: vracnode2
DB SID: orcl2
Physical DB
DB Name: orcl.oracle.com
Shared Storage
Standby Site
Application Server instances at the DR standby site access their configuration data in the Metadata Repository schemas in the database below.
Non-RAC: No Oracle Clusterware Installed
Physical Hostname: standbynode1
Virtual Hostname: vracnode1
DB SID: orcl1
Physical DB
DB Name: orcl.oracle.com

- Table 3–3 shows the host and database names that will be used in the steps in the following section. The procedure assumes a two-node Real Application Clusters on the primary site.

*Table 3–3 Host and Database Names on the Primary and Standby Sites*

| | Primary Site | Standby Site |
|---|---|---|
| Physical hostnames | prodnode1, prodnode2 | standbynode1 |
| Virtual hostnames | vracnode1, vracnode2 | vracnode1 |
| Database name | orcl.oracle.com | orcl.oracle.com |
| Database SID | orcl1 on prodnode1<br>orcl2 on prodnode2 | orcl1 on standbynode1 |

- The vracnode1 virtual hostname should be mapped to the system's IP address. You set up this mapping in each system's `hosts` file. For example:

On the primary site, edit `%SystemRoot%\system32\drivers\etc\hosts` (Windows) or `/etc/hosts` (UNIX) to add an entry similar to the following:

```
ip_address  prodnode1.domain.com  prodnode1  vracnode1.domain.com  vracnode1
```

On the standby site, edit the `hosts` file to add an entry similar to the following:

```
ip_address  standbynode1.domain.com  standbynode1  vracnode1.domain.com
 vracnode1
```

- Before proceeding to Section 3.2.2.2, "Configuration Procedure," do the following:

  1. On host prodnode1 at the primary site, install an OracleAS Infrastructure in its own Oracle home.
  2. On host standbynode1 at the standby site, install the same OracleAS Infrastructure in the equivalent Oracle home directory.
  3. On host prodnode1 at the primary site, create a database in a new Oracle home (*not* the Oracle AS Infrastructure home). Then use the Oracle Application Server Metadata Repository Creation Assistant to store the Metadata Repository schemas required by the OracleAS Infrastructure in that database. Refer to *Oracle Application Server Metadata Repository Creation Assistant User's Guide* for more information on using the OracleAS Metadata Repository Creation Assistant to load the Metadata Repository schemas into a database.
  4. On host standbynode1 at the standby site, install *only* the Oracle Database software in the equivalent database home.

  **Note:** During the Oracle Database software installation on standbynode1, do *not* create a database in the home. In Section 3.2.2.2, "Configuration Procedure," you will use the `create standby database` command to create the standby database in the standby site home and to configure Oracle Data Guard so that it will ship archive logs from the primary database at the production site to the standby database at the standby site.

  The `create standby database` command does not work properly if a database exists in the standby site home when the command is executed. Refer to Appendix A.1.22, "Database Already Exists Errors During Create Standby" if a database already exists in the home on the standby site peer host.

- You must install the standalone version of Oracle Application Server Guard 10.1.2.3 into its own Oracle home on the database host. The standalone version of Oracle Application Server Guard 10.1.2.3 can be found on Oracle Technology Network at:

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

  For instructions on how to run the standalone Oracle Application Server Guard installer, see the "Installing in High Availability Environments" chapter in the *Oracle Application Server Installation Guide* for your platform.

- The `create standby database` command is designed to automate the creation of simple standby databases and to set up the Oracle Data Guard configuration between the primary and standby databases. It does not support some database options, such as the OMF (Oracle Managed Files) or ASM

(Automatic Storage Management) storage options. If you plan to use the `create standby database` command to create a database at the standby site, create the database instance on the primary site without specifying the OMF or ASM storage options.

- Databases that use the OMF or ASM storage options can be included in a Disaster Recovery topology, but you cannot use the `create standby database` command to create a standby database that uses these storage options, and you must use Oracle Data Guard to set up the Oracle Data Guard configuration for the primary and standby databases. See Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology" for more information on configuring Oracle Data Guard for databases that use the OMF or ASM storage options.
- The DBName (without domain) and DBSID must be the same when creating the database for a Disaster Recovery setup on primary or standby sites. When you create a new database instance using the DBCA, the SID defaults to the database name. If you enter a name in the SID field other than the database name, and then later add this database to the Disaster Recovery topology, the instance name in the topology added will be empty.
- If a `sync topology` command in a RAC-Non RAC Linux environment fails and you receive missing archive logs errors, ping the standby node using tnsping. If you are unable to ping the standby node, stop and restart the listener for that node and retry the tnsping.
- If you update the AS Recovery Manager's  config_misc_files.inp to include other configuration files as part of the backup procedure, special consideration has to be made for corresponding changes at a Disaster Recovery standby site. Changes to this file could be made manually or indirectly through the MDS repository LifeCycle Tool backup option. With each change, files specified in config_misc_files.inp and on a corresponding restore, are restored. However, the config_misc_files.inp is versioned and restored in the form config_misc_files.inp config_misc_files.inp_<backup_time_stamp>. For a Disaster Recovery standby site, the latest version of this file has to be used after a failover or switchover operation. To ensure this configuration change is made, if a timestamped version exists, the latest version of ORACLE_HOME/backup_restore/config/config_misc_files.inp_<backup_time_stamp> in each Oracle home has to be copied to replace the current version of ORACLE_HOME/backup_restore/config/config_misc_files.inp. This must be done after any Oracle Application Server Guard `instantiate`, `failover`, or `switchover` operation.
- If you have multiple nodes in the topology on the primary site, then the name of each Oracle Application Server instance must be unique across all the homes on all the nodes in the primary site.

#### 3.2.2.2 Configuration Procedure

Perform the following steps to configure your OracleAS Disaster Recovery topology where the primary site uses a RAC database, but the standby site uses a non-RAC database.

1. Make sure that the database on prodnode1 is running.
2. Create the standby database on the standby node by running the following ASGCTL commands on prodnode1.

> **Note:** The `create standby database` command does not work properly if a database exists in the database home on the standby site peer host when the command is executed. If a database already exists in the home on the standby site peer host, follow the instructions in Appendix A.1.22, "Database Already Exists Errors During Create Standby" before executing the `create standby database` command in the following steps.

Here are some notes on the commands in the following section:

- On UNIX, the `add instance` command uses `orcl` (the database name) to locate the oratab entry.

  On Windows, it uses `orcl1` (the database SID) to locate the registry entry.

- The `set primary database` and `create standby database` commands use `orcl` (the database name) on UNIX, but on Windows they use `orcl1` (the database SID).

```
For 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg prodnode1 ias_admin/<adminpwd>

For 10.1.3.x releases:
ASGCTL> connect asg prodnode1 oc4jadmin/<adminpwd>

ASGCTL> set trace on all

UNIX only: ASGCTL> add instance orcl on vracnode1.oracle.com
Windows only: ASGCTL> add instance orcl1 on vracnode1.oracle.com

The output from this dump topology command should list the database instance
added in the previous command:
ASGCTL> dump topology

The output from this verify topology command should indicate that an HA
directory exists for the database instance added with the add instance command:
ASGCTL> verify topology

ASGCTL> set noprompt

UNIX only: ASGCTL> set primary database sys/<passwd>@orcl
Windows only: ASGCTL> set primary database sys/<passwd>@orcl1

UNIX only: ASGCTL> create standby database orcl on standbynode1
Windows only: ASGCTL> create standby database orcl1 on standbynode1

The output from this verify topology command should indicate that an HA
directory exists on both the primary site and standby site for the database
instance added with the add instance command:
ASGCTL> verify topology with standbynode1
ASGCTL> instantiate topology to standbynode1
```

3. Propagate orcl_remote1 or orcl1_remote1 entries from prodnode1 to other nodes on the primary site.

   a. Copy the orcl_remote1 (UNIX) or orcl1_remote1 (Windows) entries in tnsnames.ora on prodnode1 to all the other RAC nodes on the primary site.

On UNIX, the entry uses the database name (`orcl`), but on Windows it uses the database SID (`orcl1`). A "`_remote<n>`" is appended to the name of the entry, where *<n>* is a number.

In some cases, the *<n>* number will advance, and the `_remote<n>` entry specified in the `SERVICE` attribute of the `LOG_ARCHIVE_DEST_<n>` parameter must be propagated as well.

**b.** On prodnode2, restart the listener using Oracle Clusterware:

```
> CRSHOME/bin/crs_stop  ora.prodnode2.LISTENER_PRODNODE2.lsnr
> CRSHOME/bin/crs_start ora.prodnode2.LISTENER_PRODNODE2.lsnr
```

**c.** Make sure that the standby database mentioned in the remote entry is pingable using TNS.

```
UNIX only: > tnsping orcl_remote1
Windows only: > tnsping orcl1_remote1
```

**4.** On prodnode1, run the ASGCTL `sync topology` command.

```
For 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg ias_admin/<adminpwd>

For 10.1.3.x releases:
ASGCTL> connect asg oc4jadmin/<adminpwd>

UNIX only: ASGCTL> set primary database sys/<passwd>@orcl
Windows only: ASGCTL> set primary database sys/<passwd>@orcl1

ASGCTL> sync topology to standbynode1
```

#### 3.2.2.3 Switchover Procedure

This section describes how to run the ASGCTL `switchover topology` command to switch from the primary site to the standby site to prepare for a scheduled outage of the primary site.

After the scheduled outage is over, you can switch back to the primary site. See Section 3.2.2.4, "Switchback Procedure" for details.

For unscheduled outages, you should perform the steps in Section 3.2.2.5, "Failover Procedure" instead.

Procedure for switching over to the standby site for scheduled outages:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.
2. On prodnode1, invoke the Oracle Application Server Guard client command-line utility asgctl (asgctl.sh is located in `<ORACLE_HOME>/dsa/bin` on UNIX systems and asgctl.bat is located in `<ORACLE_HOME>\dsa\bin` on Windows systems) and then connect to the Oracle Application Server Guard server:

```
For 10.1.2.x and 10.1.4.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg ias_admin/<adminpwd>
```

```
Successfully connected to prodnode1:7890

For 10.1.3.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.2.x and 10.1.4.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg ias_admin/<adminpwd>
Successfully connected to prodnode1:7890

For 10.1.3.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg oc4jadmin/<adminpwd>
Successfully connected to prodnode1:7890
```

3. Use the `verify topology` command to confirm that the production topology is running and the configuration is valid:

   ```
   ASGCTL> verify topology
   ```

4. Use the `verify topology` and specify the standby Infrastructure host (the standby Infrastructure host's network hostname is standbynode1) to confirm that the standby topology is consistent with the production site and meets Disaster Recovery requirements:

   ```
   ASGCTL> verify topology with standbynode1
   ```

5. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information:

   ```
   UNIX only: ASGCTL> set primary database sys/<paswd>@orcl
   Windows only: ASGCTL> set primary database sys/<passwd>@orcl1
   ```

6. Switchover the topology to the standby site. If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file:

   ```
   ASGCTL> switchover topology to standbynode1
   ```

   > **Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

7. Disconnect from the *old* production site Oracle Application Server Guard server:

   ```
   ASGCTL> disconnect
   ```

8. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

9. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

#### 3.2.2.4 Switchback Procedure

This section describes the steps for switching back to the primary site when the scheduled outage is over.

Run the following commands to switch back to the primary site:

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. On standbynode1 (the current primary site), invoke the Oracle Application Server Guard client command-line utility asgctl (asgctl.sh is located in `<ORACLE_HOME>/dsa/bin` on UNIX systems and asgctl.bat is located in `<ORACLE_HOME>\dsa\bin` on Windows systems) and then connect to the Oracle Application Server Guard server:

```
For 10.1.2.x and 10.1.4.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
Successfully connected to standbynode1:7890

For 10.1.3.x releases on UNIX:
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
Successfully connected to standbynode1:7890

For 10.1.2.x and 10.1.4.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.2.0.2
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>
Successfully connected to standbynode1:7890

For 10.1.3.x releases on Windows:
> asgctl.bat
Application Server Guard: Release 10.1.3.2.0
(c) Copyright 2004, 2005 Oracle Corporation. All rights reserved
ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
Successfully connected to standbynode1:7890
```

3. Use the `verify topology` command to confirm that the production topology is running and the configuration is valid:

```
ASGCTL> verify topology
```

4. Use the `verify topology` and specify the standby Infrastructure host (the standby Infrastructure host's network hostname is prodnode1, the original

primary host) to confirm that the standby topology is consistent with the production site and meets Disaster Recovery requirements:

```
ASGCTL> verify topology with prodnode1
```

5. Specify the primary database. See Section 2.11.1.2, "Specifying the Primary Database" for more information:

```
UNIX only: ASGCTL> set primary database sys/<passwd>@orcl
Windows only: ASGCTL> set primary database sys/<passwd>@orcl1
```

6. Switchover the topology to the standby site (the original primary site). If you want to use a policy file, specify the `using policy <file>` parameter where `<file>` represents the path and file specification for the `switchover_policy.xml` file:

```
ASGCTL> switchover topology to prodnode1
```

> **Note:** As part of the Oracle Application Server Guard switchover operation, an implicit `sync topology` command is performed to make sure the topologies are identical. In addition, all OPMN services are stopped and then restarted on the production site.

7. Disconnect from the *old* production site Oracle Application Server Guard server:

```
ASGCTL> disconnect
```

8. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

9. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

#### 3.2.2.5 Failover Procedure

This section describes the steps for failing over to the standby site. Use these steps for unscheduled outages of the primary site. For scheduled outages, see the steps in Section 3.2.2.3, "Switchover Procedure."

To fail over to the standby site, run these commands on the standby site and activate it as the new primary site.

1. If you are manually changing DNS names to manage DNS switchover, reduce the wide area DNS TTL value for the site. See Section 1.5, "Wide Area DNS Operations" for a description of the two methods of performing a DNS switchover. See Section 1.5.2, "Manually Changing DNS Names" for more information about manually changing DNS names.

2. Connect to the Oracle Application Server Guard server on the standby site. The network name is standbynode1.

```
For 10.1.2.x and 10.1.4.x releases:
ASGCTL> connect asg standbynode1 ias_admin/<adminpwd>

For 10.1.3.x releases:
ASGCTL> connect asg standbynode1 oc4jadmin/<adminpwd>
```

3. Specify that the primary OracleAS Metadata Repository database on the standby site is now identified as the new primary database on this new production site.

The keyword **new** is in bold text in the following example to indicate its importance as a keyword. If you have multiple OracleAS Metadata Repositories in your topology, you must authenticate each one using the `set new primary database` command.

```
UNIX only: ASGCTL> set new primary database sys/<passwd>@orcl
Windows only: ASGCTL> set new primary database sys/<passwd>@orcl1
```

4. Execute the asgctl `failover` command.

```
ASGCTL> set trace on all
ASGCTL> failover
ASGCTL> disconnect
```

5. Discover the topology. You must perform this operation to create a new topology file for this production site.

   For 10.1.2.x and 10.1.4.x environments, discover the topology as follows:

```
ASGCTL> discover topology oidpassword=oidpwd
```

   For 10.1.3.x environments, discover the topology as follows:

```
ASGCTL> discover topology within farm
```

   You can also use the `add instance` command to add the standalone database instances to the Disaster Recovery topology as follows:

```
ASGCTL> add instance asdb on vracnode1.oracle.com
```

6. Perform a wide area DNS switchover to direct requests to the new production site based on one of the methods presented in Section 1.5, "Wide Area DNS Operations."

7. If you are manually changing DNS names to manage DNS switchover, adjust the wide area DNS TTL to an appropriate value.

When you must perform a failover, you should also decide whether to set up a new standby site or not. If you decide to set up a new standby sitie, then after the failover operation has completed, perform an asgctl `create standby database` command to create the standby database on the remote host and then perform an asgctl `instantiate topology` operation.

# 4

# Disaster Recovery Special Considerations

This chapter includes the following information about special considerations for OracleAS Disaster Recovery:

- Section 4.1, "Special Considerations for Some OracleAS Metadata Repository Configurations"
- Section 4.2, "Special Considerations for OracleAS Disaster Recovery Environments"

## 4.1 Special Considerations for Some OracleAS Metadata Repository Configurations

This section describes special considerations for multiple OracleAS Metadata Repositories and OracleAS Metadata Repositories created using the OracleAS Metadata Repository Creation Assistant.

### 4.1.1 Special Considerations for Multiple OracleAS Metadata Repository Configurations

By default, the credentials you specified in the asgctl `connect asg` command are used whenever one Oracle Application Server Guard server connects to another Oracle Application Server Guard server. However, there may be cases where you want to do either of the following:

- Use different credentials for each system on a given site, see Section 4.1.1.1, "Setting asgctl Credentials."
- Use a common set of credentials in the standby topology that are the same as the credentials used in the primary topology, see Section 4.1.1.2, "Specifying the Primary Database."

If the credentials for any host system are not the same as those used in the asgctl connect command, you must set the Oracle Application Server Guard credentials so that the Oracle Application Server Guard server can connect to each host system in the configuration.

#### 4.1.1.1 Setting asgctl Credentials

To set different credentials for all the host systems belonging to the same topology, enter the following asgctl command at the asgctl prompt. Specify the node name of the host system to which the credentials apply and the `ias_admin` account name and password for the `ias_admin` account created during the Oracle Application Server installation, and the key words **for topology**.

> **Note:** For Oracle AS 10.1.3.x releases, the user name must be oc4jadmin and the password for the oc4jadmin account created during the Oracle Application Server 10.1.3.x installation.

These settings are good for the current session.

```
ASGCTL> set asg credentials standbyinfra ias_admin/<iasadminpwd> for topology
```

When you specify the key words, **for topology**, you set the credentials for all the host systems that belong to the same topology as the specified system; otherwise, the credentials will apply only for the specified host system.

The `set asg credentials` command is also useful when you want to use different credentials for a specific server on the topology. In the previous example, the same credentials were set for all nodes on the standby topology, so that these credentials differ from the credentials used in the primary topology. The following command sets the credentials for a specific node, the standbyinfra node, on the standby topology.

```
ASGCTL> set asg credentials standbyinfra ias_admin/<iasadminpwd>
```

To summarize, if you set the credentials for a topology, these credentials are inherited for the entire topology. If you set the credentials for an individual host on the topology, the credentials (for this host) override the default credentials set for the topology.

In addition, for topologies that have more than one Infrastructure, such as a collocated Oracle Internet Directory+OracleAS Metadata Repository and a separate Portal OracleAS Metadata Repository, Oracle Application Server Guard requires that you set the credentials for each system on which an Infrastructure resides before performing any important Oracle Application Server Guard operations, such as instantiate, sync, switchover, and failover. See set asg credentials for an example.

#### 4.1.1.2 Specifying the Primary Database

To identify the OracleAS Infrastructure database on the primary topology, enter the following asgctl command at the asgctl prompt. Specify the user name and password for the database account with sysdba privileges to access the OracleAS Infrastructure Database on the primary topology and the TNS service name of the OracleAS Infrastructure database:

```
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL>
```

The standby site uses the same values as specified for the primary database because the service name and password for both the primary and standby OracleAS Infrastructure databases must be the same.

If a production or standby site has multiple OracleAS Metadata Repository instances installed and you are performing an instantiate, sync, or switchover operation, you must identify all of the OracleAS Metadata Repository instances by performing a `set primary database` command for each OracleAS Metadata Repository instance before performing either an instantiate, sync, or switchover operation. See set asg credentials for an example.

#### 4.1.1.3 Setting Oracle Application Server Guard Port Numbers

Oracle Application Server Guard uses a default port (port) number of 7890; for example, `port=7890`. If there are any additional Oracle homes installed on a system, each additional Oracle home must have a unique Oracle Application Server Guard port number, that is usually incremented by the value one, for example, `port=7891`, and so forth. See Section 2.1.6, "Supported OracleAS Disaster Recovery Configurations" for more information.

### 4.1.2 Special Considerations for OracleAS Metadata Repository Configurations Created Using OracleAS Metadata Repository Creation Assistant

The following items are special considerations for an OracleAS Metadata Repository configuration created using OracleAS Metadata Repository Creation Assistant. These Metadata Repository databases are installed in Oracle homes with schemas containing user data. For this reason, there are some special considerations regarding OracleAS Disaster Recovery.

- On the standby site, no Metadata Repository is created by OracleAS Disaster Recovery. The System Administrator must use the OracleAS Metadata Repository Creation Assistant on the standby site and create this Metadata Repository.
- During a clone topology operation to the standby site no instantiate operation is performed on the Metadata Repository.
- **Warning:** Do not perform a clone operation to a standby system containing an existing Oracle home, other than the standalone Oracle Application Server Guard home, because it will get overwritten. Perform a clone operation only to a standby system where no Oracle home is installed other than the standalone Oracle Application Server Guard home.
- The OracleAS Disaster Recovery solution assumes that user schemas are already configured for Oracle Data Guard.
- The OracleAS Disaster Recovery solution assumes that when using Oracle Data Guard, that the Metadata Repository is not in managed recovery mode.
- OracleAS Disaster Recovery will not change the recovery mode of Oracle Data Guard for the Metadata Repository if it is found to be in user-managed recovery mode; instead, Oracle Application Server Guard will issue a warning indicating that the database is in user-managed recovery mode and this feature must be set differently. For more information about user-managed recovery mode, refer to the "Performing User-Managed Database Flashback and Recovery" section of *Oracle Database Backup and Recovery User's Guide* in the Oracle Database documentation set.
- Oracle Application Server Guard must be installed in every Oracle Application Server home on every system that is part of your production and standby topology configured for the OracleAS Disaster Recovery solution. The standalone version of Oracle Application Server Guard 10.1.2.3 must also be installed in its own Oracle home on any database host that includes a database that you want to include in your Disaster Recovery topology. After you install the standalone Oracle Application Server Guard kit on the database host, use the asgctl add instance command to add the database instance to your Disaster Recovery topology. The standalone version of Oracle Application Server Guard 10.1.2.3 can be downloaded from Oracle Technology Network at:

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

  See the OracleAS Disaster Recovery installation information in in *Oracle Application Server Installation Guide* for more information.

## 4.2 Special Considerations for OracleAS Disaster Recovery Environments

The following sections describe some additional special considerations for OracleAS Disaster Recovery environments.

### 4.2.1 Some Special Considerations That Must Be Taken When Setting Up Some OracleAS Disaster Recovery Sites

Some special considerations must be taken when setting up OracleAS Disaster Recovery for sites that include middle-tier CFC configurations.

In CFC configurations, the instance name stored in Oracle Internet Directory is comprised of the original host name on which the production site installation was performed. In the case of an OracleAS Disaster Recovery site having a symmetric topology, the instance on the standby site peer host must be installed identically to the production site or you must use the clone instance or clone topology command to create the instance at the standby site peer host.

### 4.2.2 Handling dsa.conf or asg.conf Configuration Files for Asymmetric Topologies

The Oracle Application Server Guard operation synchronizes the configuration files of the standby site with those of the production site through a backup operation on the primary site and restores them to the standby site.

Additionally, for asymmetric topologies the `dsa.conf` configuration file (for 10.1.2.x and 10.1.4.x releases) or `asg.conf` configuration file (for 10.1.3.x releases) for an Oracle home may contain special settings on the production site that are different from the standby site. For example, the `inventory_location` parameter setting may be different on the standby site than it is on the primary site. In this case, you should also exclude the `dsa.conf` or `asg.conf` configuration file from the backup list of files so it is not restored on the standby site. Otherwise, in this example, the location of the OraInventory will not be correct on the standby site following a switchover or failover operation.

In both these cases, you should modify the OracleAS Recovery Manager backup and restore exclusion file, `config_exclude_files.inp`, as follows to exclude both of these configuration files from the list of files being backed up, which ensures that neither of these files is then restored to the standby site:

```
For 10.1.2.x and 10.1.4.x releases:
# Exclude Files
# - Add additional files to this list that you want to be ignored
# - during the configuration file backup/restore
c:\oracle\ias1012\opmn\conf\ons.conf
c:\oracle\ias1012\dsa\dsa.conf

For 10.1.3.x releases:
# Exclude Files
# - Add additional files to this list that you want to be ignored
# - during the configuration file backup/restore
c:\oracle\ias1013\opmn\conf\ons.conf
c:\oracle\ias1013\dsa\asg.conf
```

The exclusion file is found at:

```
ORACLE_HOME/backup_restore/config/config_exclude_files.inp
```

See the backup and recovery part of the *Oracle Application Server Administrator's Guide* for more information about performing backup and restore operations using OracleAS Recovery Manager.

If the directives set in the `dsa.conf` file or `asg.conf` file are necessary at the site that currently functions as the production site, it may be desirable to include the `dsa.conf` file or `asg.conf` file for synchronization and add a post switchover or failover step to edit physical site specific directives.

### 4.2.3 Customized Preference Store Location for Portlets Not Preserved After Switchover Operation

If you change the preference stores of the OmniPortlet and Web Clipping portlets to a location other than the default out-of-box setting (by changing the value in the `provider.xml` file for OmniPortlet and `mds-config.xml` file for Web Clipping), you must edit the XML files on the standby site after the switchover operation. You must do this because the switchover operation does not copy over the updated XML files. The location you specify for the preference stores must exist at the standby site.

Table 4–1 shows the location of the XML files for the OmniPortlet and Web Clipping portlets.

***Table 4–1 Location of XML Files for OmniPortlet and Web Clipping***

| Portlet | Location of XML File |
|---|---|
| OmniPortlet | `ORACLE_HOME/j2ee/OC4J_WebCenter/applications/portalTools/omniPortlet/WEB-INF/providers/omniPortlet/provider.xml` |
| Web Clipping | `ORACLE_HOME/j2ee/OC4J_WebCenter/applications/portalTools/webClipping/WEB-INF/mds-config.xml` |

If you do not customize the location of the preference store, then you do not have to edit the XML files after a switchover operation.

### 4.2.4 Other Special Considerations for OracleAS Disaster Recovery Environments

See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for information describing some additional special considerations.

# 5

# Oracle Application Server Guard asgctl Command-line Reference

This chapter contains reference information describing the asgctl commands. Table 5–1 summarizes all the asgctl commands. Table 5–2 summarizes all the asgctl commands that were deprecated beginning with OracleAS release 10.1.2.0.2. Subsequent sections provide detailed reference information common to many commands and about each command.

*Table 5–1 Summary of asgctl Commands*

| Command | Description |
|---|---|
| add instance | Adds to the local topology file, the specified instance name and name of the host system on which this instance is installed, and if specified, propagates this updated topology file to all instances in the Disaster Recovery production environment. |
| asgctl | Invokes the Oracle Application Server Guard client command-line utility asgctl. On UNIX systems, `asgctl.sh` is located in *<ORACLE_HOME>*`/dsa/bin` and on Windows systems, `asgctl.bat` is located in *<ORACLE_HOME>*`\dsa\bin`. |
| clone instance | Copies a single Application Server instance at a production site host to a standby site host. |
| clone topology | Copies two or more Application Server instances at a production site to the standby site. |
| connect asg | Connects the Oracle Application Server Guard client to the Oracle Application Server Guard server. |
| create standby database | Creates the standby database on the remote host system. |
| disconnect | Disconnects the Oracle Application Server Guard client from the Oracle Application Server Guard server. |
| discover topology | Discovers by querying Oracle Internet Directory all instances within the topology that share the same Oracle Internet Directory for a production site and generates a topology XML file that describes the topology. |
| discover topology within farm | Discovers the topology within the farm for a site when Oracle Internet Directory is not available; in this case, Oracle Application Server Guard server uses OPMN to discover the topology within the farm. |
| dump policies | Directs Oracle Application Server Guard server to write detailed, default policy information to respective XML formatted files for a set of asgctl commands. Each policy file can then be edited and later specified to define the topology's disaster recovery policy to be used with the respective administrative command. |

*Table 5–1 (Cont.) Summary of asgctl Commands*

| Command | Description |
|---|---|
| dump topology | Directs the Oracle Application Server Guard server to write detailed information about the topology to the screen or if specified, to a file. |
| exit | Disconnects the Oracle Application Server Guard client from any existing connections and exits the Oracle Application Server Guard client. This has the same effect as the quit command. |
| failover | During an unscheduled outage of the production site, the standby site becomes the production site. |
| help | Displays help information at the command line. |
| instantiate topology | Creates a topology at the standby site (after verifying that the primary and standby sites are valid for OracleAS Disaster Recovery); also synchronizes the standby site with the primary site so that the primary and standby sites are consistent. |
| quit | Disconnects the Oracle Application Server Guard client from any existing connections and exits the Oracle Application Server Guard client. This has the same effect as the exit command. |
| remove instance | Removes from the local topology file, the specified instance name, and if specified, propagates this updated topology file to all instances in the Disaster Recovery production environment. |
| run | Remotely executes a script or program that resides in any home where Oracle Application Server Guard is installed. |
| set asg credentials | Sets the credentials used to authenticate the Oracle Application Server Guard client connections to Oracle Application Server Guard servers and connections between Oracle Application Server Guard servers to a specific host. |
| set echo | Sets command-echoing on or off in an asgctl script. |
| set new primary database | Identifies the OracleAS Infrastructure database on the standby topology as the new primary OracleAS Infrastructure database. |
| set noprompt | Sets the noprompt state in an asgctl script in which all interactive prompts are thereafter ignored. |
| set primary database | Identifies the OracleAS Infrastructure database on the primary topology. |
| set trace | Enables or disables tracing for the specified trace flag. When tracing for a flag is set to on, the output of the trace is written to the Oracle Application Server Guard log files. |
| show env | Shows the current environment of the Oracle Application Server Guard server to which the Oracle Application Server Guard clients is connected. |
| show operation | Shows the current operation. |
| shutdown | Shuts down the Oracle Application Server Guard server at the operating system command-line prompt on a system on which OPMN is not running. This command is only used with cloning an instance or cloning a topology. |
| shutdown topology | Shuts down a running topology. |
| startup | Starts up the Oracle Application Server Guard server at the operating system command-line prompt on a system on which OPMN is not running. This command is only used with cloning an instance or cloning a topology. |
| startup topology | Starts up a shutdown topology. |

*Table 5–1 (Cont.) Summary of asgctl Commands*

| Command | Description |
|---|---|
| stop operation | Stops the specified operation. |
| switchover topology | During a scheduled outage of the production site, switches the roles of the production site with the standby site so that the standby site now becomes the production site. |
| sync topology | Synchronizes the standby site with the primary site so that the primary and standby sites are consistent. |
| verify topology | Verifies that the topology is running and the configuration is valid. If a standby topology is specified, this command compares the primary and standby topologies to verify that they conform to the requirements for OracleAS Disaster Recovery. |

*Table 5–2 Summary of Deprecated asgctl Commands*

| Command | Description |
|---|---|
| dump farm (Deprecated) | Directs the Oracle Application Server Guard server to write detailed information about the farm to the screen or if specified, to a file. |
| instantiate farm (Deprecated) | Creates a farm at the standby site (after verifying that the primary and standby sites are valid for OracleAS Disaster Recovery; also synchronizes the standby site with the primary site so that the primary and standby sites are consistent. |
| shutdown farm (Deprecated) | Shuts down a running farm. |
| startup farm (Deprecated) | Starts up a shutdown farm. |
| switchover farm (Deprecated) | During a scheduled outage of the production site, switches the roles of the production site with the standby site so that the standby site now becomes the production site. |
| sync farm (Deprecated) | Synchronizes the standby site with the primary site so that the primary and standby sites are consistent. |
| verify farm (Deprecated) | Verifies that the farm is running and the configuration is valid. If a standby farm is specified, this command compares the primary and standby farms to verify that they conform to the requirements for OracleAS Disaster Recovery. |

## 5.1 Information Common to Oracle Application Server Guard asgctl Commands

This section describes information that is common to Oracle Application Server Guard asgctl commands.

### General Information

All asgctl commands should be run using the physical host names of systems where there is a parameter in which a system host name can be specified.

The Oracle Application Server Guard client must be connected to an Oracle Application Server Guard server when you issue any asgctl command with the exception of startup and shutdown commands.

The Oracle Application Server Guard server will act as the coordinating server for all operations performed on the systems being configured. By default, this is the local system where the `connect asg` command is being executed. This system must be a member of the production site topology.

### Oracle Application Server Guard Server Information

The Oracle Application Server Guard server must be started on the standby host system (`<standby_topology_host>`. The Oracle Application Server Guard server can be stopped and started using the opmnctl command-line Utility as follows:

```
For 10.1.2.x and 10.1.4.x releases, these commands start and stop OracleAS Guard
on UNIX:
<ORACLE_HOME>/opmn/bin/opmnctl  startproc  ias-component=DSA
<ORACLE_HOME>/opmn/bin/opmnctl  stopproc  ias-component=DSA

For 10.1.3.x releases, these commands start and stop OracleAS Guard on UNIX:
<ORACLE_HOME>/opmn/bin/opmnctl  startproc  ias-component=ASG
<ORACLE_HOME>/opmn/bin/opmnctl  stopproc  ias-component=ASG

For 10.1.2.x and 10.1.4.x releases, these commands start and stop OracleAS Guard
on Windows:
<ORACLE_HOME>\opmn\bin\opmnctl  startproc  ias-component=DSA
<ORACLE_HOME>\opmn\bin\opmnctl  stopproc  ias-component=DSA

For 10.1.3.x releases, these commands start and stop OracleAS Guard on Windows:
<ORACLE_HOME>\opmn\bin\opmnctl  startproc  ias-component=ASG
<ORACLE_HOME>\opmn\bin\opmnctl  stopproc  ias-component=ASG
```

## 5.2 Information Specific to a Small Set of Oracle Application Server Guard Commands

This section describes information that is specific to a small set of Oracle Application Server Guard operations, such as instantiate, sync, failover, switchover, dump topology, discover topology, clone topology, verify topology, setting the primary database, and setting asg credentials.

If a production or standby site has multiple OracleAS Metadata Repository instances installed and you are performing an instantiate, sync, switchover, or failover operation, you must identify all of the OracleAS Metadata Repository instances by performing a `set primary database` command for each and every OracleAS Metadata Repository instance prior to performing either an instantiate, sync, switchover, or failover operation.

Before performing any Oracle Application Server Guard operations, you must shut down the emagents. This operation is required for Oracle Application Server Guard commands that recycle OracleAS services, such as failover and switchover operations. You can issue the asgctl run command in a script to perform this operation from within Oracle Application Server Guard. Otherwise, for example you may get an "ORA-01093: ALTER DATABASE CLOSE only permitted with no sessions connected" error message. Shutting down emagents is only described for performing a switchover operation. However, it applies to all Oracle Application Server Guard operations as previously described.

Oracle Application Server Guard requires that you set the credentials for any Oracle Application Server Guard server system in the topology that has different credentials from the Oracle Application Server Guard server to which you are connected before performing any important Oracle Application Server Guard operations, such as instantiate, sync, switchover, and failover. See set asg credentials for an example.

You must perform a discover topology command when you first set up your OracleAS Disaster Recovery environment in order to initially create the topology XML file; there after, you should perform a discover topology operation whenever you procure another Oracle home in a production site or change roles from a production to a standby site through a switchover or failover operation.

Oracle Application Server Guard Administrators must give a delay of about a minute before performing the next asgctl sync topology, switchover topology, or instantiate topology operation. This is especially important when performing incremental sync topology operations. This is a known limitation intended to keep sequences of underlying events as orderly as possible.

The following information and scenarios will help to clarify the usage of topology files.

- For OracleAS release 10.1.2 and earlier, use the discover topology command. If Oracle Internet Directory is not in the topology, then use the discover topology within farm command.
- For OracleAS releases 10.1.2 and 10.1.3, connect to the Oracle Application Server Guard 10.1.2 server (from either an OracleAS 10.1.2 or 10.1.3 home) and perform a discover topology command. Oracle Application Server Guard will write a `topology.xml` file to each OracleAS home in the discovered topology. Next, if you want to add an instance (add instance command), then you must perform this operation from the OracleAS 10.1.3 home and connect to any OracleAS 10.1.2 system in the existing topology and perform the add instance command.
- For OracleAS release 10.1.3, connect to an Oracle Application Server Guard 10.1.3 server and add that instance to the topology (which does not yet exist). Then, add any additional instances using the same Oracle Application Server Guard connection.

An important point to emphasize in these last two scenarios is that if the `topology.xml` file does not exist in the OracleAS home of the instance you are adding to the topology, Oracle Application Server Guard creates a new `topology.xml` file, which essentially defines a new topology.

If you want to use a policy file, edit the contents of the XML policy file to define by instance the domain of execution operations that are permitted for any one of these asgctl commands (clone topology, dump topology, failover, instantiate topology, switchover topology, sync topology, and verify topology). Each instance list entry in this XML policy file (`clone_policy.xml`, `dump_policy.xml`, `failover_policy.xml`, `instantiate_policy.xml`, `switchover_policy.xml`, `sync_policy.xml`, and `verify_policy.xml`) logically tags a production-standby peer combination with a particular attribute that defines the success requirement for the commands successful operation. See Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information and an example of an XML policy file.

### 5.2.1 Special Considerations for OracleAS Disaster Recovery Configurations in CFC Environments

In an OracleAS Disaster Recovery configuration that uses CFC on the primary topology or standby topology, or both, the following information must be considered before performing an asgctl clone, instantiate topology, switchover topology, or failover command. Before taking a cold backup or restoring the metadata repository database, the OracleAS Recovery Manager shuts down the database first.

For example, in the Windows CFC environment, Oracle Fail Safe performs database polling and restarts the database if it is down. Hence, every time before the administrator performs a clone, instantiate, switchover, or failover operation, the administrator must disable database polling in Oracle Fail Safe and re-enable it after the backup/restore operation (after the clone, instantiate, switchover, or failover operation completes). The steps to perform this sequence of operations are described in a note in Section 5.2.1.1, "Special Considerations for Running Instantiate and Failover Operations in CFC Environments" and Section 5.2.1.3, "Special Considerations for Running a Switchover Operation in CFC Environments".

#### 5.2.1.1 Special Considerations for Running Instantiate and Failover Operations in CFC Environments

In an OracleAS Disaster Recovery configuration that uses CFC on the primary topology or standby topology, or both, the following information must be considered before performing an asgctl clone, instantiate, switchover, or failover operation.

Before taking a cold backup or restoring the metadata repository database, the OracleAS Recovery Manager shuts down the database first.

For example, in the Windows CFC environment, Oracle Fail Safe performs database polling and restarts the database if it is down. Hence, every time before the administrator performs an instantiate, switchover, or failover operation, the administrator must disable database polling in Oracle Fail Safe and re-enable it after the backup/restore operation (after the clone, instantiate, switchover, or failover operation completes).

The steps to perform this sequence of operations are as follows:

1. Using Microsoft Cluster Administrator, open the cluster group that contains the Application Server resources. Take the following resources offline in this order: Oracle Process Manager, then Oracle Database, then Oracle Listener.
2. Using Windows Service Control Manager, start the following services in this order: Fail Safe Listener, then the Oracle Database service.
3. From a Windows command prompt, use the sqlplus command-line Utility to startup the database.
4. Using Windows Service Control Manager, start the Oracle Process Manager.
5. Perform the asgctl commands, including the clone, instantiate, switchover, or failover operation.
6. Using Microsoft Cluster Administrator, open up the cluster group that contains the Application Server resources and bring up the following resources online in this order: Oracle Listener, then Oracle Database, then Oracle Process Manager.

#### 5.2.1.2 A Special Consideration and Workaround for Performing an Instantiate Operation in CFC Environments

When performing an instantiate operation, Oracle Application Server Guard puts an entry for the remote database in the `tnsnames.ora` file on both the production and standby site. The service name of this entry is constructed by concatenating _ REMOTE1 to the database service name (for example, ORCL_REMOTE1). The entry contains the IP address of the target host where the database is running. On the production site, the IP will refer to the standby system and on the standby site, the IP refers to the production system.

In a CFC environment, the database is accessed using a virtual IP rather than a physical IP. When Oracle Application Server Guard creates the `tnsnames.ora` entry it should use the virtual IP, but it uses the physical IP instead. This problem will be fixed in a future release of Oracle Application Server Guard. As a workaround, when performing an instantiate operation in this environment, edit the `tnsnames.ora` file after an instantiation operation and replace the physical IP in the entry with the virtual IP used to access the database.

#### 5.2.1.3 Special Considerations for Running a Switchover Operation in CFC Environments

In an OracleAS Disaster Recovery configuration that uses CFC on the primary topology or standby topology or both, the following information must be considered before performing an asgctl instantiate topology, switchover topology, or failover command.

Before taking a cold backup or restoring the metadata repository database, the OracleAS Recovery Manager shuts down the database first.

For example, in the Windows CFC environment, Oracle Fail Safe performs database polling and restarts the database if it is down. Hence, every time before the administrator performs an instantiate, switchover, or failover operation, the administrator must disable database polling in Oracle Fail Safe and re-enable it after the backup/restore operation (after the instantiate, switchover, or failover operation completes).

The steps to perform this sequence of operations are as follows:

1. Using Microsoft Cluster Administrator, open the cluster group that contains the Application Server resources. Take the following resources offline in this order: Oracle Process Manager, then Oracle Database, then Oracle Listener.
2. Using Windows Service Control Manager, start the following services in this order: Fail Safe Listener, then the Oracle Database service.
3. From a Windows command prompt, use sqlplus to start up the database.
4. Perform the asgctl commands, including the instantiate topology, switchover topology, or failover command.
5. Using Microsoft Cluster Administrator, open up the cluster group that contains the Application Server resources and bring up the following resources online in this order: Oracle Listener, then Oracle Database, then Oracle Process Manager.

### 5.2.2 Other Special Considerations for OracleAS Disaster Recovery Environments

See Section 4.1, "Special Considerations for Some OracleAS Metadata Repository Configurations" and Section 4.2, "Special Considerations for OracleAS Disaster Recovery Environments" for information describing some additional special considerations for OracleAS Disaster Recovery environments.

# add instance

Adds to the local topology file, the specified instance name and name of the host system name on which this instance is installed, and if specified, propagates the updated topology file to all instances in the Disaster Recovery production environment.

You can use this command to add either an Application Server instance or an Oracle database instance to the local topology file.

## Format

```
add instance <instance_name> on <instance_host> [to topology]
```

## Parameters

**instance_name**
The name of the instance to be added to the topology file.

If you are adding a database instance to the topology file, instead of specifying the database instance name, specify the database identifier. If the database is a RAC database, the database identifier is the database SID. If the database is a single instance Linux database, then the database identifier is the database unique name.

**instance_host**
The name of the host on which this instance is installed. In the Disaster Recovery environment, if this host has a virtual hostname or a virtual hostname alias, then the virtual hostname must be used because it is this value that is placed in the topology file. For hosts with a virtual hostname or a virtual hostname aliases, use the fully qualified virtual hostname or fully qualified virtual hostname alias (the fully qualified name includes the domain name) for the instance host.

If the host does not have a virtual hostname or virtual hostname alias, use its fully qualified physical hostname.

**Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**to topology**
A keyword, that if present in the command line, directs Oracle Application Server Guard to propagate the updated topology file to all instances in the Disaster Recovery production environment. Any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby environment.

## Usage Notes

This command is useful for managing an instance on an Oracle Application Server Guard server to which the Oracle Application Server Guard client is connected. For example, you may have used the remove instance command to remove from the local topology file or from all topology files within the topology, an instance on this local host system because of a problem with it. Now you want to add to the local topology file or to all topology files in the topology, the good instance. In this case, you may not have wanted to manage the bad instance through the policy file where you could have set the success requirement attribute to `Ignore` for this instance when invoking asgctl commands to run across the entire topology.

This command is particularly useful for managing Disaster Recovery farms in which Oracle Internet Directory (OID) is not available, in other words, an OracleAS Release 10.1.3 only topology. You must use the discover topology within farm command to initially create the topology file for each instance within this farm. Then you can manage instances by adding or removing individual instances from the local topology file using the add instance and remove instance commands. If you specify the `to topology` or `from topology` keywords, the updated local topology file changes are propagated to all instances in the Disaster Recovery environment. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

This command is useful for adding an OracleAS 10.1.3 J2EE instance to an OID based 10.1.2.0.2 topology to support a mixed version Disaster Recovery environment. For example, you can use the add instance command to add an OracleAS 10.1.3 J2EE instance to your OID based 10.1.2.0.2 topology. See Section 2.6, "Adding or Removing Oracle Application Server 10.1.3 Instances to Redundant Single Oracle Application Server 10.1.3 Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology" for a use case. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

## Example

The following command in the example adds to the local topology file only, an instance named oc4j1 that is installed on the local host system named prodinfra.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> discover topology within farm
ASGCTL> add instance oc4j1 on prodinfra.oracle.com
```

## asgctl

Invokes the Oracle Application Server Guard client from the operating system command-line prompt or runs a script, if the path name to the script is provided.

### Format

asgctl@[filename]

### Parameters

**filename = <file-path>**
The path to a file that contains asgctl commands that you want to run as a script.

### Usage Notes

On UNIX systems, `asgctl.sh` is located in *<ORACLE_HOME>*`/dsa/bin` and on Windows systems, `asgctl.bat` is located in *<ORACLE_HOME>*`\dsa\bin`.

### Example

```
> asgctl.sh
Application Server Guard: Release 10.1.3.2.0

(c) Copyright 2004, 2007 Oracle Corporation. All rights reserved
ASGCTL>
```

## clone instance

Copies a single Oracle Application Server instance at a production site host to the same directory on a standby site host. This command also establishes the Disaster Recovery production-standby relationship for the instances.

### Format

```
clone instance <instance> to <standby_topology_host> [no standby]
```

### Parameters

**instance**
The name of the instance.

**standby_topology_host**
The name of the standby site host to which the instance is to be cloned. In the Disaster Recovery environment, if this host has a physical hostname, use the fully qualified physical hostname (the fully qualified name includes the domain name) for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its fully qualified network hostname (the fully qualified name includes the domain name).

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**no standby**
This keyword directs Oracle Application Server Guard to copy the Application Server instance home from the production site host to the standby site host, but *not* establish the Disaster Recovery relationship between the production instance and standby instance. When you use this keyword, the Oracle Application Server instance home at the production site host is only copied to the standby site host, which is equivalent to using Oracle Universal Installer to install the instance at the standby site host.

When this keyword is used for a `clone instance` command that copies an Application Server home installed by an Infrastructure installation and a database with the Metadata Repository schemas was created in the Application Server home during the installation, Oracle Application Server Guard will copy the database to the standby site host, but will not configure Oracle Data Guard for the database.

### Usage Notes

The default command copies an Oracle Application Server instance at a primary site host to the same directory on the standby site peer host and then sets up the Disaster Recovery relationship between the production and standby instance.

If you clone an Oracle Application Server instance installed by an Infrastructure installation and a database with the Metadata Repository schemas was created in the Application Server home during the installation, then Oracle Application Server Guard will copy the database to the standby site peer host and will configure Oracle Data Guard for the primary and standby databases.

**Note:** The *only* type of database that the `clone instance` command copies and configures Oracle Data Guard for is described in the previous paragraph.

For more information about the Oracle Data Guard configuration set up for this type of database by this command, refer to Section 1.1.1.2, "Understanding the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard."

For more information on including databases in your OracleAS Disaster Recovery topology and configuring Oracle Data Guard for those databases, refer to Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology."

The production instance must be included in the Disaster Recovery topology at the primary site. Refer to the add instance command description for information on adding an instance to the Disaster Recovery topology.

The production instance to be cloned cannot exist on the standby site host.

The following are prerequisites for the `clone instance` command:

- Install and start the Oracle Application Server Guard standalone kit in its own Oracle home on the standby site host. Do not install any other Oracle components in the Oracle home for the Oracle Application Server Guard standalone kit on the standby site host.
- A Java development kit with its jar utility must be installed on the standby site host.
- For Windows hosts, release 5.0.2134.1 or higher of the services kit (`sc.exe`) must be installed under C:\WINDOWS\system32 on the production and standby site hosts.

  If you do not take this step prior to performing a cloning operation on Windows, you may see errors similar to these during a cloning operation:

  ```
  stajz09: -->ASG_DUF-4040: Error executing the external program or script.
  The error code is "255"
  "IM.asinfra.us.oracle.com"
  stajz09P: -->ASG_IAS-15689: Error running the backup script
  stajz09: -->ASG_IAS-15685: Failed to backup configuration data for instance
  "IM.asinfra.us.oracle.com"
  stajz09: -->ASG_DUF-3027: Error while executing Clone Instance at step -
  backup step.
  stajz09: -->ASG_DUF-3027: Error while executing Clone Topology at step -
  clone home step.
  ```

- For the dcm-daemon component in the %ORACLE_HOME%\opmn\conf\opmn.xml file on Windows hosts, increase the start timeout parameter's retry interval to 5 seconds. The following example shows the section of the opmn.xml file for the dcm-daemon component with the start timeout parameter's retry interval set to 5:

  ```
  <ias-component id="dcm-daemon" status="enabled" id-matching="true">
      <process-type id="dcm-daemon" module-id="DCMDaemon">
          <start timeout="600" retry="5"/>
          <stop timeout="120"/>
          <process-set id="dcm" numprocs="1">
  ```

If you do not take this step prior to performing a cloning operation on Windows, you may see errors similar to these during a cloning operation:

```
stajz09: -->ASG_SYSTEM-100: Command "C:\work\im/opmn/bin/opmnctl.exe shutdown"
failed, check log file C:\work\im\dsa\bkup\log/2007-07-17_01-41-51_loha.log
for detail.
stajz09: -->ASG_SYSTEM-100: Failure : prepare failed.
stajz09: -->ASG_SYSTEM-100:
stajz09: -->ASG_SYSTEM-100: OPMN managed processes could not be stopped.
stajz09: -->ASG_SYSTEM-100: Status code:
stajz09: -->ASG_SYSTEM-100: opmnctl shutdown failed.
```

Use the `no standby` parameter with the `clone instance` command to clone the current configuration, but not establish the target as a standby site. The ability to create Application Server cluster topologies was introduced in Application Server release 10.1.3.1, but the clustering attributes cannot be propagated to the target without interfering with the original cluster if the target is on the same network. To alleviate this problem, ASG removes the topology entry `[<topology>...</topology>]` from the `opmn.xml` configuration file and leaves the resulting target unclustered. After the `clone instance` command with the `no standby` parameter has completed, you can cluster the resulting target configuration by editing and renaming the `opmn.xml_asg` file in each Oracle home. When the target is configured and maintained as a standby site, the standby cluster is automatically reconfigured during failover and switchover operations.

It is important to understand that during every asgctl clone operation the following are copied from the production site host (or hosts) to the standby site peer host (or hosts):

- the Oracle home directory and files for each of the Oracle Application Server instances involved in the clone operation
- the entire Oracle Universal Installer Central Inventory (Central Inventory) for each production site host for which an Oracle Application Server home is copied

Be aware of the following implications of the cloning behavior described in the previous list:

- You cannot successfully clone one Application Server instance from one production site host and then a second Application Server instance from a different production site host to a single standby site host. Each `clone instance` command *will* successfully copy the Oracle home directory and files for the specified Application Server instance from the production site host to the standby site host. Because the entire Central Inventory for the production site host is also copied to the standby site host during the clone operation, the Central Inventory from the production site host overwrites the Central Inventory on the standby site host.

  Therefore, in this cloning scenario, during the second clone operation the Central Inventory from the second production site host is copied to the standby site host, and it overwrites the Central Inventory that was copied to the standby site host during the first clone operation. After the second clone operation, there will no longer be a Central Inventory entry at the standby site host for the first Application Server instance that was cloned, so you will not be able to use the software for the first Application Server instance that was cloned to the standby site host.

  However, after the second clone operation in this scenario, you can use Oracle Universal Installer to register the Application Server home that was copied to the standby site host during the first clone operation with the Central Inventory on that host. Registering the Application Server home that was copied during the first clone operation will allow you to use that software when the standby site becomes the production site (after a switchover or failover operation). Use the `-attachHome` flag of Oracle Universal Installer to register the home with the Central Inventory on the standby site host. See *Oracle Universal Installer and OPatch User's Guide* for information on using the `-attachHome` flag.

- If a production site host has several Application Server instances installed and you clone one of those instances to a standby site host, the clone operation will copy the Oracle home directory and files for the specified Application Server instance to the standby site host and the entire Central Inventory for the production site host. After the clone operation completes, the standby site host's Central Inventory includes references to *all* of the Oracle homes that were installed on the production site host. However, the home for the Application Server instance that was specified with the `clone instance` command is the *only* Oracle home that was actually copied to the standby site host. The other Oracle homes referenced in the Central Inventory on the standby site were *not* copied from the production site host to the standby site host.

  In this scenario, after the clone operation completes, you can use Oracle Universal Installer to remove references to the Oracle homes that were not copied to the standby site host from the Central Inventory on that host. Removing a reference to an Oracle home from the Central Inventory is called detaching an Oracle home. After detaching the Oracle homes that do not exist at the standby site host, the Central Inventory for that host will have the correct Oracle home information. Use the `-detachHome` flag of Oracle Universal Installer to detach Oracle homes from the Central Inventory on the standby site host. See *Oracle Universal Installer and OPatch User's Guide* for information on using the `-detachHome` flag.

See Section 2.7, "Oracle Application Server Guard Operations -- Standby Site Cloning of One or More Production Instances to a Standby System" for more information.

The basic procedure consists of the following pre-clone steps and clone steps.

### Pre-Clone Steps

Perform the following steps:

1. Log in as su - root on UNIX or as Administrator on Windows to the production site host where the Application Server instance that you want to clone is installed.
2. CD to the instance home on the production site host for the instance that you want to clone.
3. On Windows, make sure that release 5.0.2134.1 or higher of the services kit (`sc.exe`) is installed in the `C:\WINDOWS\system32` directory on the production site host.
4. Shut down the Oracle Application Server Guard server.

```
For 10.1.2.x and 10.1.4.x releases, this command stops OracleAS Guard on UNIX:
> <ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=DSA

For 10.1.3.x releases, this command stops OracleAS Guard on UNIX:
> <ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=ASG

For 10.1.2.x and 10.1.4.x releases, this command stops OracleAS Guard on
Windows:
> <ORACLE_HOME>\opmn\bin\opmnctl stopproc ias-component=DSA

For 10.1.3.x releases, this command stops OracleAS Guard on Windows:
```

```
> <ORACLE_HOME>\opmn\bin\opmnctl stopproc ias-component=ASG
```

5. On UNIX, log in as root on the standby site host that the Application Server instance home will be cloned to and make sure `dsaServer.sh` in *<ORACLE_HOME>*`/dsa/bin` is executable by everyone. If it is not, record the permission, then change the executable permission by issuing the following command:

   ```
   chmod +x dsaServer.sh
   chmod u+x asgexec
   ```

6. On Windows, on the standby site host that the Application Server instance home will be cloned to:

   1. Add the jdk\bin path to the system path on the standby site host.
   2. Create a new command window.
   3. In the new command window, run the jar command with no parameters to make sure the jar utility is found.
   4. Make sure that release 5.0.2134.1 or higher of the services kit (`sc.exe`) is installed in the `C:\WINDOWS\system32` directory on the standby site host.

7. On both the production site host and standby site host that will be involved in the clone operation, invoke asgctl and issue the startup command.

   ```
   From the <ORACLE_HOME>/dsa/bin directory on a UNIX host:
   > asgctl.sh startup

   From the <ORACLE_HOME>\asg\bin directory in the new command window on a Windows
   host:
   > asgctl startup
   ```

8. On UNIX hosts, log out as root.

### Clone Steps

Perform the following steps:

1. Log in as user (non root user on UNIX hosts) to the production site host where the Application Server instance that you want to clone is installed.
2. CD to the instance home on the production site host for the Application Server instance that you want to clone.
3. Invoke asgctl on the production site host and run the `clone instance` command to clone the Application Server instance and home to the same directory on the standby site peer host.

   **Note:** In the command output, you will see a number of connect messages. This is normal as the Oracle Application Server Guard server is recycled during these operations.

4. Log out of the production site host.

   **Note:** If Oracle Application Server Guard does not run as root on UNIX hosts, the user will be prompted by the Oracle Application Server Guard client to run the underlying operations at each of the instance homes as root (manually) in order to continue with the operation.

This last step completes the cloning instance operation and brings the hosts back to where they were before you started the clone instance operation. At this point you could invoke asgctl, connect to a production site host, discover the topology, and then perform a verify operation to determine whether the production and standby topologies are valid and consistent with one another as you would expect them to be.

## Example

The following command in the example clones an instance named portal_2 to the standby site host named asmid2.

```
1. Check the prerequisites as described in the Usage Notes.
2. Perform the Pre-Clone steps as described in the Usage Notes.
3. Perform the Clone steps as described in the Usage Notes.
   a. Log in as user to the production site host for the instance you want to
clone.
   b. CD to the Oracle home for the Application Server instance to be cloned.
   c. Invoke asgctl and run the clone instance command.
> asgctl.sh

For 10.1.2.x and 10.1.4.x releases, use this command to connect to an ASG server:
ASGCTL> connect asg prodoc4j ias_admin/adminpwd
Successfully connected to prodoc4j:7890

For 10.1.3.x releases, use this command to connect to an ASG server:
ASGCTL> connect asg prodoc4j oc4jadmin/adminpwd
Successfully connected to prodoc4j:7890

ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> clone instance portal_2 to asmid2.oracle.com
.
.
.
ASGCTL> disconnect
ASGCTL> exit
>
   d. Log off the production site host.
```

## clone topology

Copies all the Oracle Application Server instances from all of the production site hosts to the same directories on the standby site peer hosts. This command also establishes the Disaster Recovery production-standby relationship for these instances.

### Format

```
clone topology to <standby_topology_host> [using policy <file>] [no standby]
```

### Parameters

**standby_topology_host**
The name of a standby site host to which one or more instances will be cloned. In the Disaster Recovery environment, if this host has a physical hostname, use the fully qualified physical hostname for the host (the fully qualified name includes the domain name).

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its fully qualified network hostname (the fully qualified name includes the domain name).

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**using policy <file>**
Full path and file specification for the XML policy file.

You can use a policy file with the `clone topology` command to copy only a subset of the Oracle Application Server homes at the production site hosts to the standby site peer hosts. Refer to Section 2.4, "Dumping Policy Files and Using Policy Files With Some asgctl Commands" for more information about using policy files.

**no standby**
This keyword directs Oracle Application Server Guard to copy the Application Server instance homes from the production site hosts to the standby site hosts, but *not* establish the Disaster Recovery relationship between the production instances and standby instances. When you use this keyword, the Oracle Application Server instance homes at the production site hosts are only copied to the standby site hosts, which is equivalent to using Oracle Universal Installer to install the instances at the standby site hosts.

When this keyword is used for a `clone topology` command that copies any Application Server home installed by an Infrastructure installation and a database with the Metadata Repository schemas was created in the Application Server home during the installation, Oracle Application Server Guard will copy the database to the standby site host, but will not configure Oracle Data Guard for the database.

### Usage Notes

The default command copies all the Oracle Application Server instances from all of the production site hosts to the same directories on the standby site peer hosts and then sets up the Disaster Recovery relationship between the production site instances and the standby site peer instances. You can use a policy file to exclude specific Application Server instance homes on production site hosts from the `clone topology` operation.

If you clone any Oracle Application Server instance installed by an Infrastructure installation and a database with the Metadata Repository schemas was created in the Application Server home during the installation, then Oracle Application Server Guard will copy the database to the standby site peer host and will configure Oracle Data Guard for the primary and standby databases.

**Note:** The *only* type of database that the `clone topology` command copies and configures Oracle Data Guard for is described in the previous paragraph.

For more information about the Oracle Data Guard configuration set up for this type of database by this command, refer to Section 1.1.1.2, "Understanding the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard."

For more information on including databases in your OracleAS Disaster Recovery topology and configuring Oracle Data Guard for those databases, refer to Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology."

This command is not supported by Disaster Recovery configurations where production and standby peer hosts have a different number of Oracle homes.

The production instances to be cloned cannot exist on the standby site hosts.

The following are prerequisites for the `clone topology` command:

- Install and start the Oracle Application Server Guard standalone kit in its own Oracle home on each standby site host. Do not install any other Oracle components in the Oracle home for the Oracle Application Server Guard standalone kit on the standby site host.
- A Java development kit with its jar utility must be installed on each standby site host.
- For Windows hosts, release 5.0.2134.1 or higher of the services kit (`sc.exe`) must be installed under C:\WINDOWS\system32 on the production and standby site hosts.

  If you do not take this step prior to performing a cloning operation on Windows, you may see errors similar to these during a cloning operation:

  ```
  stajz09: -->ASG_DUF-4040: Error executing the external program or script.
  The error code is "255"
  "IM.asinfra.us.oracle.com"
  stajz09P: -->ASG_IAS-15689: Error running the backup script
  stajz09: -->ASG_IAS-15685: Failed to backup configuration data for instance
  "IM.asinfra.us.oracle.com"
  stajz09: -->ASG_DUF-3027: Error while executing Clone Instance at step -
  backup step.
  stajz09: -->ASG_DUF-3027: Error while executing Clone Topology at step -
  clone home step.
  ```

- For the dcm-daemon component in the %ORACLE_HOME%\opmn\conf\opmn.xml file on Windows hosts, increase the start timeout parameter's retry interval to 5 seconds. The following example shows the section of the opmn.xml file for the dcm-daemon component with the start timeout parameter's retry interval set to 5:

```
<ias-component id="dcm-daemon" status="enabled" id-matching="true">
    <process-type id="dcm-daemon" module-id="DCMDaemon">
        <start timeout="600" retry="5"/>
        <stop timeout="120"/>
        <process-set id="dcm" numprocs="1">
```

If you do not take this step prior to performing a cloning operation on Windows, you may see errors similar to these during a cloning operation:

```
stajz09: -->ASG_SYSTEM-100: Command "C:\work\im/opmn/bin/opmnctl.exe shutdown"
failed, check log file C:\work\im\dsa\bkup\log/2007-07-17_01-41-51_loha.log
for detail.
stajz09: -->ASG_SYSTEM-100: Failure : prepare failed.
stajz09: -->ASG_SYSTEM-100:
stajz09: -->ASG_SYSTEM-100: OPMN managed processes could not be stopped.
stajz09: -->ASG_SYSTEM-100: Status code:
stajz09: -->ASG_SYSTEM-100: opmnctl shutdown failed.
```

Use the `no standby` parameter with the `clone topology` command to clone the current configuration, but not establish the targets as a standby site. The ability to create Application Server cluster topologies was introduced in Application Server release 10.1.3.1, but the clustering attributes cannot be propagated to the target without interfering with the original cluster if the targets are on the same network. To alleviate this problem, ASG removes the topology entry `[<topology>...</topology>]` from the `opmn.xml` configuration file and leaves the resulting targets unclustered. After the `clone topology` command with the `no standby` parameter has completed, you can cluster the resulting target configurations by editing and renaming the `opmn.xml_asg` files in each Oracle home. When the targets are configured and maintained as a standby site, the standby cluster is automatically reconfigured during failover and switchover operations.

It is important to understand that during every asgctl clone operation the following are copied from the production site host (or hosts) to the standby site peer host (or hosts):

- the Oracle home directory and files for each of the Oracle Application Server instances involved in the clone operation
- the entire Oracle software registry for each production site host for which an Oracle Application Server home is copied

Be aware of the following implications of the cloning behavior described in the previous list:

- You cannot successfully clone one Application Server instance from one production site host and then a second Application Server instance from a different production site host to a single standby site host. Each `clone instance` command *will* successfully copy the Oracle home directory and files for the specified Application Server instance from the production site host to the standby site host. Because the entire Central Inventory for the production site host is also copied to the standby site host during the clone operation, the Central Inventory from the production site host overwrites the Central Inventory on the standby site host.

  Therefore, in this cloning scenario, during the second clone operation the Central Inventory from the second production site host is copied to the standby site host, and it overwrites the Central Inventory that was copied to the standby site host during the first clone operation. After the second clone operation, there will no longer be a Central Inventory entry at the standby site host for the first Application Server instance that was cloned, so you will not be able to use the software for the first Application Server instance that was cloned to the standby site host.

However, after the second clone operation in this scenario, you can use Oracle Universal Installer to register the Application Server home that was copied to the standby site host during the first clone operation with the Central Inventory on that host. Registering the Application Server home that was copied during the first clone operation will allow you to use that software when the standby site becomes the production site (after a switchover or failover operation). Use the `-attachHome` flag of Oracle Universal Installer to register the home with the Central Inventory on the standby site host. See *Oracle Universal Installer and OPatch User's Guide* for information on using the `-attachHome` flag.

- If a production site host has several Application Server instances installed and you clone one of those instances to a standby site host, the clone operation will copy the Oracle home directory and files for the specified Application Server instance to the standby site host and the entire Central Inventory for the production site host. After the clone operation completes, the standby site host's Central Inventory includes references to *all* of the Oracle homes that were installed on the production site host. However, the home for the Application Server instance that was specified with the `clone instance` command is the *only* Oracle home that was actually copied to the standby site host. The other Oracle homes referenced in the Central Inventory on the standby site were *not* copied from the production site host to the standby site host.

  In this scenario, after the clone operation completes, you can use Oracle Universal Installer to remove references to the Oracle homes that were not copied to the standby site host from the Central Inventory on that host. Removing a reference to an Oracle home from the Central Inventory is called detaching an Oracle home. After detaching the Oracle homes that do not exist at the standby site host, the Central Inventory for that host will have the correct Oracle home information. Use the `-detachHome` flag of Oracle Universal Installer to detach Oracle homes from the Central Inventory on the standby site host. See *Oracle Universal Installer and OPatch User's Guide* for information on using the `-detachHome` flag.

See Section 2.7, "Oracle Application Server Guard Operations -- Standby Site Cloning of One or More Production Instances to a Standby System" for more information.

The basic procedure consists of the following pre-clone steps and clone steps.

### Pre-Clone Steps

Perform the following steps:

1. Log in as su - root on UNIX or as Administrator on Windows to each production site host where an Application Server instance that you want to clone is installed.
2. CD to the instance homes on the production site hosts for each instance that you want to clone.
3. On Windows, make sure that release 5.0.2134.1 or higher of the services kit (`sc.exe`) is installed in the `C:\WINDOWS\system32` directory on the production site hosts.
4. Shut down the Oracle Application Server Guard server.

   ```
   For 10.1.2.x and 10.1.4.x releases, this command stops OracleAS Guard on UNIX:
   ```

```
> <ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=DSA

For 10.1.3.x releases, this command stops OracleAS Guard on UNIX:
> <ORACLE_HOME>/opmn/bin/opmnctl stopproc ias-component=ASG

For 10.1.2.x and 10.1.4.x releases, this command stops OracleAS Guard on
Windows:
> <ORACLE_HOME>\opmn\bin\opmnctl stopproc ias-component=DSA

For 10.1.3.x releases, this command stops OracleAS Guard on Windows:
> <ORACLE_HOME>\opmn\bin\opmnctl stopproc ias-component=ASG
```

5. On UNIX, log in as root on each standby site host that an Application Server instance home will be cloned to and make sure `dsaServer.sh` in *<ORACLE_HOME>*`/dsa/bin` is executable by everyone. If it is not, record the permission, then change the executable permission by issuing the following command:

   ```
   chmod +x dsaServer.sh
   chmod u+x asgexec
   ```

6. On Windows, on each standby site host that an Application Server instance home will be cloned to:
   1. Add the jdk\bin path to the system path on the standby site host.
   2. Create a new command window.
   3. In the new command window, run the jar command with no parameters to make sure the jar utility is found.
   4. Make sure that release 5.0.2134.1 or higher of the services kit (`sc.exe`) is installed in the `C:\WINDOWS\system32` directory on the standby site host.
7. On both the production site host or hosts and the standby site host or hosts that will be involved in the clone operation, invoke asgctl and issue the startup command.

   ```
   From the <ORACLE_HOME>/dsa/bin directory on a UNIX host:
   > asgctl.sh startup

   From the <ORACLE_HOME>\asg\bin directory in the new command window on a Windows
   host:
   > asgctl startup
   ```

8. On UNIX hosts, log out as root.

### Clone Steps

Perform the following steps:

1. Log in as user (non root user on UNIX hosts) to the production site host or hosts where the Application Server instances that you want to clone are installed.
2. CD to the instance homes on the production site hosts for each Application Server instance that you want to clone.
3. Invoke asgctl on any of the production site hosts and run the `clone topology` command to clone the Application Server instances and homes for all of the production site hosts to the same directories on the standby site peer hosts. Remember that you can use a policy file to exclude specific Application Server instance homes on production site hosts from a clone topology operation.

---

**Note:** In the command output, you will see a number of connect messages. This is normal as the Oracle Application Server Guard server is recycled during these operations.

---

4. Log out of the system.

---

**Note:** If Oracle Application Server Guard does not run as root on UNIX hosts, the user will be prompted by the Oracle Application Server Guard client to run the underlying operations at each of the instance homes as root (manually) in order to continue with the operation.

---

This last step completes the cloning topology operation and brings the hosts back to where they were before you started the clone topology operation. At this point you could invoke asgctl, connect to a production site host, discover the topology, and then perform a verify operation to determine whether the production and standby topologies are valid and consistent with one another as you would expect them to be.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example shows how to copy multiple Application Server instances from production site hosts to standby site peer hosts.

```
1. Check the prerequisites as described in the Usage Notes.
2. Perform the Pre-Clone steps as described in the Usage Notes.
3. Perform the Clone steps as described in the Usage Notes.
   a. Log in as user to all the production site hosts where an Application Server
instance home you want to clone is installed.
   b. CD to the Oracle home for each Application Server instance to be cloned.
   c. Invoke asgctl on one of the production site hosts and perform the clone
topology command.
> asgctl.sh

For 10.1.2.x and 10.1.4.x releases, use this command to connect to an ASG server:
ASGCTL> connect asg prodoc4j ias_admin/adminpwd
Successfully connected to prodoc4j:7890

For 10.1.3.x releases, use this command to connect to an ASG server:
ASGCTL> connect asg prodoc4j oc4jadmin/adminpwd
Successfully connected to prodoc4j:7890

ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb

# Command to use if you are cloning all the Application Server
# instances at all the production site hosts to the standby site
# peer hosts:
ASGCTL> clone topology to standbyinfra.oracle.com
.
.
.
```

```
# Command to use if you are using a policy file (where <file>
# is the full path and file specification of the clone policy file)
# to clone a subset of the Application Server instances at the production
# site hosts to the standby site peer hosts:
ASGCTL> clone topology to standbyinfra.oracle.com using policy <file>
.
.
.
ASGCTL> disconnect
ASGCTL> exit
>
   d. Log off the system
```

# connect asg

Connects the Oracle Application Server Guard client to the Oracle Application Server Guard server on a system on which Oracle Application Server services are running.

## Format

```
connect asg [<host-name>[:<port>]] <ias_administrative_account>/<password>
```

## Parameters

**host-name = <host-name>**
The network hostname of the host system for the Oracle Application Server Guard server to which you want the Oracle Application Server Guard client to connect. This Oracle Application Server Guard server will be the coordinating server for all operations performed on the host systems being configured. The host name is optional if the Oracle Application Server Guard client and Oracle Application Server Guard server are on the same node.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the Oracle Application Server Guard server in its Oracle home.

**<ias_administrative_account>/password**
If this is an OracleAS 10.1.3 installation, the user name must be `oc4jadmin` and the password for the `oc4jadmin` account created during the Oracle Application Server installation. If this is an OracleAS 10.1.2.0.2 or lower installation, the user name must be the `ias_admin` account name and the password for the `ias_admin` account created during the Oracle Application Server installation.

> **Note:** If this is an OracleAS 10.1.3 installation, the user name must be `oc4jadmin` and the password for the `oc4jadmin` account created during the Oracle Application Server 10.1.3 installation.

## Usage Notes

- The Oracle Application Server Guard client system must have network access to the Oracle Application Server Guard host system specified with the host-name parameter.
- The Oracle Application Server Guard host system must have network access to all systems in the OracleAS Disaster Recovery configuration.
- The specified `ias_admin` or `oc4jadmin` account name must be configured with the necessary rights and privileges to permit OracleAS Disaster Recovery site operations (read and write access to all required files and directories, and so forth)
- An IP address can be used in place of a host name.

- If a password for the ias_admin or oc4jadmin account is not specified in the connect command, you will be prompted to enter a password.

## Example

The command in the following example results in the Oracle Application Server Guard client connecting to the Oracle Application Server Guard server running on a host named prodinfra using the ias_admin username and adminpwd password, respectively.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
```

## create standby database

Copies the database files for a database at a production site host to the standby site peer host and configures Oracle Data Guard for the primary database and standby database.

### Format

```
create standby database <database_identifier> on <remote_host>
```

### Parameters

**database_identifier**
Primary database identifier used to create the standby database on the remote host sytem. If the database is a RAC database, the database identifier is the database SID. If the database is a single instance Linux database, then the database identifier is the database unique name.

**remote_host**
Name of the host system on which the standby database is to be created. In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

### Usage Notes

- If a production site host has a single instance or RAC database that is installed outside an Application Server home and the database does *not* use the Oracle Managed Files (OMF) or Automatic Storage Management (ASM) database storage options, you can use the `create standby database` command to create a standby database on the standby site peer host and configure Oracle Data Guard for the primary and standby databases.

  > **Note:** For more information on including databases in your OracleAS Disaster Recovery topology and configuring Oracle Data Guard for those databases, refer to Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology."
  >
  > For more information about the Oracle Data Guard configuration set up by this command, refer to Section 1.1.1.2, "Understanding the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard."

- The `create standby database` command is designed to automate the creation of simple standby databases. It does not support some database options, such as the OMF or ASM storage options. Therefore, if you plan to use the `create standby database` command to create a database at the standby site, create the database instance on the primary site without specifying the OMF or ASM storage options.

- The `create standby database` command does *not* work properly if a database exists in the database home on the standby site peer host when the command is executed. If a database already exists in the home on the standby site peer host, follow the instructions in Section A.1.22, "Database Already Exists Errors During Create Standby" before using the `create standby database` command.
- Oracle software and Oracle Application Server Guard software are required to be installed on the node designated as `<remote_host>`.
- The `init.ora` parameter file generated for the standby database is configured assuming a non Oracle RAC enabled standby database. If the standby database is to be Oracle RAC enabled, the following initialization parameters must be defined appropriately:
  - cluster_database
  - cluster_database_instances
  - remote_listener
- In an Oracle Clusterware environment, Oracle Clusterware is configured to bring up the database automatically upon failure. However, this automatic online feature of Oracle Clusterware must be disabled on all Oracle RAC instances during Oracle Application Server Guard operations. To guarantee the disabling of automatic startup of the database, perform the following two commands:

  ```
  $DBHOME\bin\srvctl disable database -d ORCL
  $DBHOME\bin\srvctl stop database -d ORCL
  ```

  In addition, you must avoid shutting down the entire Oracle Clusterware daemons; otherwise, the database will not start up and the following error will be returned:

  ```
  SQL> startup
  ORA-29702: error occurred in Cluster Group Service operation
  SQL> exit
  ```

- If the asgctl `create standby database` command is performed in an Oracle RAC environment from one of the primary instances, for example orcl2, within a 2 node RAC instance with instances orcl1 on one node and orcl2 on another node, to the standby single node non RAC Database, a heartbeat error log will be seen in the alert logs of the other Oracle RAC instances
- If the asgctl `create standby database` command is performed in an Oracle RAC environment from one of the primary instances to the standby single node non RAC Database, a heartbeat error log will be seen in the alert logs of the other Oracle RAC instances as follows:

  ```
  ----------------------------------------------------------------------------
  PING[ARC0]: Heartbeat failed to connect to standby 'orcl2_remote1'. Error is
  12154.
  Mon Aug 28 09:53:43 2006
  Error 12154 received logging on to the standby
  Mon Aug 28 09:53:43 2006
  Errors in file c:\oracle\product\10.2.0\admin\orcl\bdump\orcl1_arc0_2752.trc:
  ORA-12154: TNS:could not resolve the connect identifier specified
  ```

```
--------------------------------------------------------------------------
```

The reason for this error is because the asgctl `create standby database` command did not configure the hosts. To work around this problem, you must manually configure the remaining hosts not configured by this command by appending the appropriate remote service name entry into the remaining Oracle RAC `tnsnames.ora` file instances. For example:

```
ORCL2_REMOTE1.MYCOMPANY.COM =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = 140.87.23.35)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl2.mycompany.com)
    )
  )
```

- In an Oracle RAC environment, the `create standby database` command will only set up an Oracle RAC to a standalone non Oracle RAC configuration with clustering options disabled. However, if the Disaster Recovery Administrator wants the standby (new production) database to be brought up as part of a cluster following either an Oracle Application Server Guard failover or switchover operation, this can be accomplished by modifying the `init.ora` file parameters as follows:

  – Change `cluster_database=false` to `cluster_database=true`

  – Change `cluster_database_instances=1` to a value *n* where *n* represents the number of instances in the cluster.

  – Set up the `remote_listener` parameter.

  Note that these changes can be set up in a script and invoked using the asgctl run command. See the run command for more information.

## Example

The command in this example takes the primary database unique name named `orcl` and creates the standby database on the remote host system named asmid1.

```
ASGCTL> create standby database orcl on asmid1
```

## disconnect

Disconnects the Oracle Application Server Guard client from the Oracle Application Server Guard server to which it is currently connected.

### Format

disconnect

### Usage Notes

The Oracle Application Server Guard client must be connected to a Oracle Application Server Guard server when you issue this command.

### Example

The command in the following example disconnects the Oracle Application Server Guard client from the Oracle Application Server Guard server to which it is currently connected.

```
ASGCTL> disconnect
ASGCTL>
```

# discover topology

Directs asgctl to query Oracle Internet Directory and determine all instances within the topology that share the same Oracle Internet Directory for a production site and generates a `topology.xml` file that describes the topology.

## Format

discover topology [oidhost=<host>] [oidsslport=<sslport>] [oiduser=<user>] oidpassword=<pass>

## Parameters

**host**
Name of the host system where Oracle Internet Directory is installed. In the Disaster Recovery environment, if this host has a virtual hostname or a virtual hostname alias, then the virtual hostname must be used because it is this value that is placed in the topology file. For hosts with a virtual hostname or a virtual hostname aliases, use the fully qualified virtual hostname or fully qualified virtual hostname alias (the fully qualified name includes the domain name) for the instance host. The host name is optional if you are currently connected to the Oracle Application Server Guard server on the host through the use of the asgctl `connect asg` command.

If the host does not have a virtual hostname or virtual hostname alias, use its fully qualified physical hostname (the fully qualified name includes the domain name).

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**sslport**
The port number of the host system where Oracle Internet Directory and Secure Sockets Layer (SSL) is installed.

**user**
The Oracle Internet Directory user name.

**pass**
The password for the specified Oracle Internet Directory user name.

## Usage Notes

You should perform a discover topology operation whenever you procure another Oracle home in a production site or change roles from a production to a standby site through a switchover or failover operation.

Discover topology creates the topology (stored in `topology.xml`) on which to perform all Oracle Application Server Guard operations. This command utilizes the information in Oracle Internet Directory to define the instances included in the topology. Additionally, it gathers local information about each instance. For this reason, it requires all production site instances to have OPMN running. For instances not managed using a DCM farm, the Oracle Application Server Guard service on the

Oracle home has to be started. If the services are not started locally, a warning will be produced and the `topology.xml` file will contain only the instances discovered.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The command in the following example discovers all the instances within the topology that share the same Oracle Internet Directory (OID) for a production site, and generates a topology XML file that describes the topology.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> discover topology oidpassword=oidpwd
Discovering topology on host "infra" with IP address "123.1.2.111" prodinfra:7890
    Connecting to the OID server on host "infra.us.oracle.com" using SSL port
"636" and username "orcladmin"
    Getting the list of databases from OID
    Gathering database information for SID "asdb" from host "infra.us.oracle.com"
    Getting the list of instances from OID
    Gathering instance information for "asr1012.infra.us.oracle.com" from host
"infra.us.oracle.com"
    Gathering instance information for "asmid1.asmid1.us.oracle.com" from host
"asmid1.us.oracle.com"
    Gathering instance information for "asmid2.asmid2.us.oracle.com" from host
"asmid2.us.oracle.com"
The topology has been discovered. A topology.xml file has been written to each
home in the topology.
ASGCTL>
```

# discover topology within farm

Directs asgctl to discover the topology within a farm at a production site for those special cases where a farm does not have Oracle Internet Directory available. For 10.1.3.x environments, Oracle recommends using the `discover topology within farm` command.

## Format

discover topology within farm

## Parameters

None.

## Usage Notes

The Oracle Application Server Guard client must be connected to a Oracle Application Server Guard server when you issue this command.

See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

## Example

The command in the following example for a special case in which Oracle Internet Directory (OID) is not available, uses OPMN to discover the application server topology within a farm of the Oracle Application Server Guard server to which the Oracle Application Server Guard client is currently connected.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> discover topology within farm
Warning: If OID is part of your environment, you should use it for discovery
Discovering topology on host "infra" with IP address "123.1.2.111"
prodinfra:7890
    Discovering instances within the topology using OPMN
    Gathering instance information for "asr1012.infra.us.oracle.com" from host
"infra.us.oracle.com"
The topology has been discovered. A topology.xml file has been written to each
home in the topology.
ASGCTL>
```

## dump policies

Directs Oracle Application Server Guard Server to write detailed, default policy information in XML formatted output for the different asgctl commands to a set of policy files located on the local host at the *<ORACLE_HOME>*`/dsa/conf` directory on UNIX systems or *<ORACLE_HOME>*`\dsa\conf` directory on Windows systems.

### Format

dump policies

### Parameters

None.

### Usage Notes

A set of XML formatted policy files are written for each of the following asgctl commands: clone topology, dump topology, failover, instantiate topology, sync topology, switchover topology, and verify topology. You can edit the respective command's policy file, then specify it in the `using policy <file>` clause for the appropriate command. This parameter lets you define the topology's disaster recovery policy for each of these Oracle Application Server Guard operations.

For the dump policy file, by default the success requirement attribute is set to optional for all instances (middle tier and OracleAS Metadata Repository).

For the failover policy file, by default the success requirement attribute is set to optional for all instances (middle tier and OracleAS Metadata Repository) and mandatory for the Oracle Internet Directory home.

For the instantiate policy file, by default the success requirement attribute is set to mandatory for all instances.

For the switchover policy file, by default the success requirement attribute is set to optional for all instances (middle tier and OracleAS Metadata Repository) and mandatory for the Oracle Internet Directory home.

For the sync policy file, by default the success requirement attribute is set to mandatory for all instances.

For the verify policy file, by default the success requirement attribute is set to optional for all instances (middle tier and OracleAS Metadata Repository) and mandatory for the Oracle Internet Directory home.

### Example

The following example writes detailed, default policy information in XML formatted output for the different asgctl commands to a set of respective policy files located on the local host.

```
ASGCTL> dump policies
Generating default policy for this operation
Creating policy files on local host in directory
"/private1/OraHome2/asr1012/dsa/conf/"
ASGCTL>
```

# dump topology

Directs asgctl to write detailed information about the topology to the specified file.

## Format

dump topology [to <file>] [using policy <file>]

## Parameters

**to <file>**
Name of file on the Oracle Application Server Guard client node where the detailed output is to be written.

**using policy <file>**
Full path and file specification for the XML policy file.

## Usage Notes

For the dump policy file, by default the success requirement attribute is set to optional for all OracleAS homes (middle tier and OracleAS Metadata Repository).

## Example

The following example writes detailed information about the topology to a local file.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> dump topology to c:\dump_mid_1.txt

Contents of file c:\dump_mid_1.txt are:

Generating default policy for this operation

 Instance: asr1012.infra.us.oracle.com
    Type: Infrastructure
    Oracle Home Name: asr1012
    Oracle Home Path: /private1/OraHome
    Version: 10.1.2.0.2
    OidHost: infra.us.oracle.com
    OidPort: 389
    VirtualHost: infra.us.oracle.com
    Host: prodinfra
    Ip: 123.1.2.111
    Operation System Arch: sparc
    Operation System Version: 5.8
    Operation System Name: SunOS

 Instance: asmid2.asmid2.us.oracle.com
    Type: Core
    Oracle Home Name: asmid2
    Oracle Home Path: /private1/OraHome2
    Version: 10.1.2.0.2
    OidHost: infra.us.oracle.com
    OidPort: 389
    VirtualHost: asmid2.us.oracle.com
    Host: asmid2
    Ip: 123.1.2.333
    Operation System Arch: sparc
    Operation System Version: 5.8
    Operation System Name: SunOS

 Instance: asmid1.asmid1.us.oracle.com
    Type: Core
    Oracle Home Name: asmid1
    Oracle Home Path: /private1/OraHome
    Version: 10.1.2.0.2
    OidHost: infra.us.oracle.com
    OidPort: 389
    VirtualHost: asmid1.us.oracle.com
    Host: asmid1
    Ip: 123.1.2.334
    Operation System Arch: sparc
    Operation System Version: 5.8
    Operation System Name: SunOS
ASGCTL>
```

The following example writes detailed information about the topology to a local file. Any instances that you want left out of the output can be specified in the policy file.

```
# Command to use if you are using a policy file
ASGCTL> dump topology to c:\dump_mid_1.txt using policy <file>
```

# exit

Disconnects from any existing connections to Oracle Application Server Guard servers and exits from the Oracle Application Server Guard client.

## Format

exit

## Parameters

**None**

## Usage Notes

None.

## Example

```
ASGCTL> exit
>
```

# failover

During an unscheduled outage of the production site, performs the failover operation on the standby site to make it the primary site.

## Format

failover [using policy <file>]

## Parameters

**using policy <file>**
Full path and file specification for the XML policy file.

## Usage Notes

Make sure OracleAS Infrastructure database is running on the standby topology before performing a failover operation. Also, the OracleAS Infrastructure database information must be set by using the set new primary database asgctl command.

The global DNS names are used to direct the failover. This will be different than the HA naming utilized in the OracleAS Disaster Recovery environment. The discovery mechanism automatically maps the topology to the corresponding peer, based off local name resolution.

For the failover policy file, by default the success requirement attribute is set to optional for all OracleAS homes (middle tier and OracleAS Metadata Repository) and mandatory for the Oracle Internet Directory home.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example performs a failover operation to a standby site.

```
ASGCTL> connect asg standbyinfra ias_admin/adminpwd
Successfully connected to standbyinfra:7890
ASGCTL> set new primary database sys/testpwd@asdb
ASGCTL> failover
Generating default policy for this operation
standbyinfra:7890
    Failover each instance in the topology from standby to primary topology
standbyinfra:7890 (home /private1/OraHome2/asr1012)
    Shutting down each instance in the topology
.
.
.
    Executing opmnctl startall command
standbyinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
ASGCTL>
```

```
# Command to use if you are using a policy file
# failover using policy <file>
```

# help

Displays help information.

## Format

help [<command>]

## Parameters

**command**
Name of the command for which you want help.

## Usage Notes

None.

## Example

The following example displays help about all commands.

```
ASGCTL> help
    connect asg [<host>] [<ias_administrator_account>/<password>]
    disconnect
    exit
    quit
    add instance <instance_name> on <instance_host> [to topology]
    clone topology to <standby_topology_host> [using policy <file>] [no standby]
    clone instance <instance> to <standby_topology_host> [no standby]
    discover topology [oidhost=<host>] [oidsslport=<sslport>] [oiduser=<user>] oidpassword=<pass>
    discover topology within farm
    dump farm [to <file>]  (Deprecated)
    dump topology  [to <file>] [using policy <file>]
    dump policies
    failover [using policy <file>]
    help [<command>]
    instantiate farm to <standby_farm_host> (Deprecated)
    instantiate topology to <standby_topology_host> [using policy <file>]
    remove instance <instance_name> [from topology]
    set asg credentials <host> <ias_administrator_account>/<password> [for topology]
    set asg credentials <host> ias_admin/<password> [for farm] (Deprecated)
    set primary database <username>/<password>@<servicename> [pfile <filename> | spfile <filename>]
    set new primary database <username>/<password>@<servicename> [pfile <filename> | spfile <filename>]
    set noprompt
    set trace on|off <traceflags>
    sync farm to <standby_farm_host> [full | incr[emental]] (Deprecated)
    sync topology to <standby_topology_host> [full | incr[emental]] [using policy <file>]
    startup [asg]
    startup farm (Deprecated)
    startup topology
    shutdown [local]
    shutdown farm (Deprecated)
    shutdown topology
    show op[eration] [full] [[his]tory]
    show env
    stop op[eration] <op#>
    switchover farm to <standby_farm_host> (Deprecated)
    switchover topology to <standby_topology_host> [using policy <file>]
    verify farm [with <host>](Deprecated)
    verify topology [with <host>] [using policy <file>]
ASGCTL>
```

# instantiate topology

Using the primary site topology, creates the standby site topology by establishing the relationship between primary site and standby site Application Server instances, mirroring the configuration, and then synchronizing the standby site with the primary site.

## Format

```
instantiate topology to <standby_topology_host>[:<port>] [using policy <file>]
```

## Parameters

**standby_topology_host**
Name of the standby host system. This parameter is required because it directs the coordinating Oracle Application Server Guard server instance to discover the instances that make up the standby site. This host system must be a member of the standby site topology.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the Oracle Application Server Guard server in its Oracle home.

**using policy <file>**
Full path and file specification for the XML policy file.

## Usage Notes

The `instantiate topology` command establishes the Disaster Recovery relationship between the Oracle Application Server homes at the Disaster Recovery production site hosts and the equivalent Oracle homes at the standby site peer hosts.

The OracleAS Infrastructure database with the Metadata Repository schemas must be running on the primary topology before you execute the `instantiate topology` command. You must also use the asgctl `set primary database` command to log in to and validate the connection to the OracleAS Infrastructure database before you execute the `instantiate topology` command.

Before you use the `instantiate topology` command, you must install the same Application Server instances into the same Oracle home directories on the production site hosts and the standby site peer hosts. If during an Application Server Infrastructure installation on a production site host you created a database with the Metadata Repository schemas in that Application Server home, you must perform the same installation in the equivalent directory on the standby site peer host.

For Application Server homes at the production site and standby site that include a database with the Metadata Repository schemas that was created during an Application Server Infrastructure installation, the `instantiate topology` command configures Oracle Data Guard for the primary database and standby database.

> **Note:** The *only* type of database that the `instantiate topology` command configures Oracle Data Guard for is described in the previous paragraph.
>
> For more information about the Oracle Data Guard configuration set up for this type of database by this command, refer to Section 1.1.1.2, "Understanding the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard."
>
> For more information on including databases in your OracleAS Disaster Recovery topology and configuring Oracle Data Guard for those databases, refer to Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology."

The global DNS names are used to direct the instantiation. The discovery mechanism automatically maps the topology to the corresponding peer, based off local name resolution.

The `instantiate topology` command performs an implicit asgctl `verify` command.

For the instantiate policy file, by default the success requirement attribute is set to mandatory for all instances.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example instantiates a standby topology by attaching the coordinating Oracle Application Server Guard server and discovering the topology of the production and standby sites, performing site verification, and establishing a OracleAS Disaster Recovery environment with the topology containing the standby topology host known by DNS as standbyinfra. Note that part way through the operation you will be prompted to answer a question regarding whether you want to shut down the database. Reply by entering `y` or `yes`.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> instantiate topology to standbyinfra
Generating default policy for this operation
prodinfra:7890
    Instantiating each instance in the topology to standby topology
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
standbyinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
```

```
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
asmid2:7890
    Verifying that the topology is symmetrical in both primary and standby configuration
.
.
.
This operation requires the database to be shutdown. Do you want to continue? Yes or No
y
.
.
.
asmid2:7890 (home /private1/oracle/asr1012)
    Starting backup/synchronization of database "orcl.us.oracle.com"
    Starting restore/synchronization of database "orcl.us.oracle.com"
    Synchronizing topology completed successfully
asmid2:7890
    Synchronizing topology completed successfully

ASGCTL>

# Command to use if you are using a policy file
# instantiate topology to standbyinfra using policy <file>
```

# quit

Instructs the Oracle Application Server Guard client to disconnect from any existing connections and exit from asgctl.

## Format

quit

## Parameters

**None**

## Usage Notes

None.

## Example

The following example exits from asgctl.

```
ASGCTL> quit
>
```

## remove instance

Removes from the local topology file, the specified instance name, and if specified, propagates this updated topology file to all instances in the Disaster Recovery production environment.

### Format

```
remove instance <instance_name> [from topology]
```

### Parameters

**instance_name**
The name of the instance to be removed from the topology file.

**from topology**
A keyword, that if present in the command line, directs Oracle Application Server Guard to propagate the updated topology file to all instances in the Disaster Recovery production environment. Any Oracle Application Server Guard operation that affects the standby site, such as verify, instantiate, sync, and switchover will automatically propagate the production topology file across the standby environment.

### Usage Notes

This command is useful for managing an instance on an Oracle Application Server Guard server to which the Oracle Application Server Guard client is connected. For example, you may have used the remove instance command to remove from the local topology file or from all topology files within the topology, an instance on this local host system because of a problem with it. Now you want to add to the local topology file or to all topology files in the topology, the good instance. In this case, you may not have wanted to manage the bad instance through the policy file where you could have set the success requirement attribute to `Ignore` for this instance when invoking asgctl commands to run across the entire topology.

This command is particularly useful for managing Disaster Recovery farms in which Oracle Internet Directory (OID) is not available, in other words, an OracleAS Release 10.1.3 only topology. You must use the discover topology within farm command to initially create the topology file for each instance within this farm. Then you can manage instances by adding or removing individual instances from the local topology file using the add instance and remove instance commands. If you specify the `to topology` or `from topology` keywords, the updated local topology file changes are propagated to all instances in the Disaster Recovery environment. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

This command is useful for adding an OracleAS 10.1.3 J2EE instance to an OID based 10.1.2.0.2 topology to support a mixed version Disaster Recovery environment. For example, you can use the add instance command to add an OracleAS 10.1.3 J2EE instance to your OID based 10.1.2.0.2 topology. See Section 2.6, "Adding or Removing Oracle Application Server 10.1.3 Instances to Redundant Single Oracle Application Server 10.1.3 Home J2EE Topology Integrated with an Existing Oracle Identity Management 10.1.4.2 Topology" for a use case. See Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information about topology files.

## Example

The following command in the example removes from the local topology file an instance named oc4j1.

```
ASGCTL> remove instance oc4j1
```

# run

Remotely executes a script or program that resides in any home where Oracle Application Server Guard is installed. The run command can be executed within a topology or at the specified instance.

## Format

```
run [at topology [using policy <file>]] <command>

run [at instance <instance_name>] <command>
```

## Parameters

**at topology**
A keyword, that if present in the command line, directs Oracle Application Server Guard to perform the run operation across the topology.

**using policy <file>**
Full path and file specification for the XML policy file.

**command**
The name as a command string of the script or binary program to be executed.

**at instance**
A keyword, that if present in the command line, directs Oracle Application Server Guard to perform the run operation at the specified instance.

**instance_name**
The name of the instance where the run command is to be executed.

## Usage Notes

This command is useful for remotely executing a script or program that resides in any Oracle home where Oracle Application Server Guard is installed. The script or program must be physically located in each Oracle home across the topology or at that specified instance where it is expected to be executed. The asgctl user must first connect to the Oracle Application Server Guard server specifying the Application Server JAZN credentials (ias_admin or oc4jadmin) before invoking this asgctl run command. It is assumed that if the user knows the JAZN credentials, then the user should be allowed to execute a script or program in the home. Upon receiving a run command invocation, Oracle Application Server Guard will verify that the file specified in the command string exists in the Oracle home where the Oracle Application Server Guard server is running before executing the script or program. The output of the script is echoed back to the asgctl console.

## Example

The command in this example remotely runs the script `my_script.sh` for the instance named asdb.

```
ASGCTL> run at instance asdb my_script.sh
```

## set asg credentials

Sets the credentials used to authenticate the Oracle Application Server Guard connections to Oracle Application Server Guard servers.

### Format

```
set asg credentials <host>[:<port>] <ias_administrative_account>/<password> [for farm] [for topology]
```

### Parameters

**host**
Name of the host system to which the credentials apply. When Oracle Application Server Guard connects to that host, it will use these credentials.

Use the network hostname of the host. The host name is optional if you are currently connected to the Oracle Application Server Guard server on the host through the use of the asgctl `connect asg` command.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the Oracle Application Server Guard server in its Oracle home.

**<ias_administrative_account>/password**
If this is an OracleAS 10.1.3 installation, the user name must be `oc4jadmin` and the password for the `oc4jadmin` account created during the Oracle Application Server 10.1.3 installation. If this is an OracleAS 10.1.2.0.2 or lower installation, the user name must be the `ias_admin` account name and the password for the `ias_admin` account created during the Oracle Application Server installation. This account name must be the same as the account name on at least one of the Oracle Application Server homes.

**for farm (deprecated)**
A keyword, that if present in the command line, directs Oracle Application Server Guard to set the credentials for all of the host systems that belong to the same farm as the local host system.

**for topology**
A keyword, that if present in the command line, directs Oracle Application Server Guard to set the credentials for all of the host systems that belong to the same topology as the local host system.

### Usage Notes

By default, the credentials used in the asgctl connect command are used whenever a Oracle Application Server Guard server needs to connect to another Oracle Application Server Guard server. However, there may be cases where you want to use different credentials for a specific server. This command enables you to use the same credentials for all nodes in a topology. For example, you may want to use a common set of credentials in the standby topology that is different from the credentials used in the primary topology.

If you set the credentials for a topology, these credentials are inherited for the entire topology. If you set the credentials for an individual host on the topology, the credentials (for this host) override the default credentials set for the topology.

For topologies that have more than one Infrastructure, such as a collocated Oracle Internet Directory+OracleAS Metadata Repository and a separate Portal OracleAS Metadata Repository, Oracle Application Server Guard requires that you set the credentials for each system on which an Infrastructure resides before performing any important Oracle Application Server Guard operations, such as instantiate, sync, switchover, and failover. This is actually a two step process in which you must first identify all OracleAS Infrastructure databases on the topology using the set the primary database command for each Infrastructure, then you must set the credentials used to authenticate the Oracle Application Server Guard connections to Oracle Application Server Guard servers on which these Infrastructures reside. The following example illustrates this concept. Assume your production topology and standby topology consists of the following systems with installed Infrastructure and middle tier software applications.

Production topology:

host01 (Identity Management+OracleAS Metadata Repository), host04 (OracleAS Metadata Repository only), host06 (J2EE), host06 (Portal & Wireless)

Standby Topology:

host02 (Identity Management+OracleAS Metadata Repository), host05 (OracleAS Metadata Repository only), host07 (J2EE), host07 (Portal & Wireless)

The following Oracle Application Server Guard set primary database and set asg credentials commands would be required to properly identify the Infrastructures and authenticate Oracle Application Server Guard connections to Oracle Application Server Guard servers prior to performing an instantiate, sync, switchover, or failover operation. Assuming that the Oracle Identity Management+OracleAS Metadata Repository Infrastructure has a service name of `orcl` and the separate Portal OracleAS Metadata Repository has a service name of `asdb`.

```
ASGCTL> set primary database sys/<password>@orcl.us.oracle.com
ASGCTL> set primary database sys/<password>@asdb.us.oracle.com
ASGCTL> set asg credentials host01.us.oracle.com ias_admin/<password>
ASGCTL> set asg credentials host04.us.oracle.com ias_admin/<password>
```

Note that for a failover operation, these steps would be carried out on the standby topology and are as follows with a change in the host system names:

```
ASGCTL> set primary database sys/<password>@orcl.us.oracle.com
ASGCTL> set primary database sys/<password>@asdb.us.oracle.com
ASGCTL> set asg credentials host02.us.oracle.com ias_admin/<password>
ASGCTL> set asg credentials host05.us.oracle.com ias_admin/<password>
```

The Oracle Application Server Guard client must be connected to a Oracle Application Server Guard server before using this command.

An IP address can be used in place of a host name.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example sets the Oracle Application Server Guard credentials of host system standbyinfra to all host systems that belong to this topology.

```
ASGCTL> set asg credentials standbyinfra ias_admin/<password> for topology
```

# set echo

Sets command-echoing on or off in an asgctl script.

## Format

set echo on | off

## Parameters

**on | off**
Specifying "on" turns on command-echoing in an asgctl script. Specifying "off" turns off command-echoing in an asgctl script.

## Usage Notes

This command is useful when running large asgctl scripts. For example, if the asgctl script has error test cases with comments entered before each test case or before each asgctl command, setting echo on displays the comment before each test case or before each asgctl command that is run to give you an explanation of what the test case is or what asgctl command is about to be run.

This command also works with nested scripts.

## Example

The following example is an asgctl script that turns on command-echoing, runs a test case, connects to a Oracle Application Server Guard server, displays detailed information about the topology, then turns echo off, disconnects from the Oracle Application Server Guard server, and exits from the Oracle Application Server Guard client.

```
> ASGCTL @myasgctltestscript.txt

# myasgctltestscript.txt
# turn on echo
set echo on

# make sure you are not connected
disconnect

# not connected, should get an error message
dump topology

# connect to an ASG server
connect asg prodinfra ias_admin/adminpwd

#display detailed info about the topology
dump topology

#disconnect
disconnect

# turn off echo
echo off
exit
```

## set new primary database

Identifies the OracleAS Infrastructure database on the standby topology as the new primary database preceding a failover operation. This command is only used as part of a failover operation.

### Format

set new primary database <username>/<password>@<servicename> [pfile <filename> | spfile <filename>]

### Parameters

**username/password**
User name and password for the database account with sysdba privileges.

**servicename**
The TNS service name of the OracleAS Infrastructure database. The name must be defined on the OracleAS Infrastructure host system; it does not need to be defined on the Oracle Application Server Guard client host system.

**pfile filename**
The filename of the primary (OracleAS Infrastructure) database initialization file that will be used when the primary database is started.

**spfile filename**
The filename of the server (OracleAS Infrastructure) initialization file that will be used when the database is started.

### Usage Notes

Before performing a failover operation, you are required to connect to the Infrastructure node of the standby topology and define the new primary database. Once the Oracle Infrastructure database on the standby site is identified as the new primary database, then you can proceed to begin the failover operation.

### Example

The following example sets the OracleAS Infrastructure database information for the standby topology as the new primary/production topology preceding a failover operation.

```
ASGCTL> connect asg standbyinfra ias_admin/adminpwd
Successfully connected to standbyinfra:7890
ASGCTL> set new primary database sys/testpwd@asdb
ASGCTL> failover
.
.
.
ASGCTL>
```

## set noprompt

Sets the noprompt state for user interaction for use in executing commands in an asgctl script.

### Format

set noprompt

### Parameters

**None**

### Usage Notes

The default value, if supplied, is taken for all interactive prompts. A prompt for a user name and password returns an error message in the noprompt state.

### Example

The following example is an asgctl script containing an asgctl set noprompt command part way through the script that thereafter ignores all subsequent interactive prompting.

```
> ASGCTL @myasgctltestscript.txt

# myasgctltestscript.txt

# connect to an ASG server
connect asg prodinfra ias_admin/adminpwd

# set the primary database
set primary database sys/testpwd@asdb

# discover the production topology
discover topology oidpassword=oidpwd

# set the noprompt state
set noprompt

#display detailed info about the topology
dump topology

#disconnect
disconnect

exit
```

## set primary database

Identifies the OracleAS Infrastructure database on the primary topology.

### Format

```
set primary database <username>/<password>@<servicename> [pfile <filename> | spfile <filename>]
```

### Parameters

**username/password**
Username and password for the database account with sysdba privileges.

**servicename**
The TNS service name of the OracleAS Infrastructure database. The name must be defined on the OracleAS Infrastructure host system; it does not need to be defined on the Oracle Application Server Guard client host system.

**pfile filename**
The filename of the primary (OracleAS Infrastructure) database initialization file that will be used when the primary database is started.

**spfile filename**
The filename of the server (OracleAS Infrastructure) initialization file that will be used when the database is started.

### Usage Notes

You must always set the primary database before performing an instantiate, sync, or switchover operation.

When you set the primary database, Oracle Application Server Guard server logs into and validates the connection to the database.

If a production or standby site has multiple OracleAS Metadata Repository instances installed and you are performing an instantiate, sync, switchover, or failover operation, you must identify all of the OracleAS Metadata Repository instances by performing a set primary database command for each and every OracleAS Metadata Repository instance prior to performing either an instantiate, sync, switchover, or failover operation. In addition, for topologies that have more than one Infrastructure, such as a collocated Oracle Internet Directory+OracleAS Metadata Repository and a separate Portal OracleAS Metadata Repository, Oracle Application Server Guard requires that you set the credentials for each system on which an Infrastructure resides before performing any important Oracle Application Server Guard operations, such as instantiate, sync, switchover, and failover. See set asg credentials for an example.

Oracle Application Server Guard requires the database to have password file authentication. If the database does not have a password file, you must use the `orapwd` utility to create a password file. Also, set the `REMOTE_LOGIN_PASSWORDFILE` initialization parameter to `EXCLUSIVE`.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example sets the OracleAS Infrastructure database information for the primary or production topology.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL>
```

The following example sets OracleAS Infrastructure database information for each OracleAS Metadata Repository installed for the primary/production topology prior to a switchover operation.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@portal_1
Checking connection to database portal_1
ASGCTL> set primary database sys/testpwd@portal_2
Checking connection to database portal_2
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> discover topology oidpassword=oidpwd
ASGCTL> switchover topology to standbyinfra
.
.
.
```

## set trace

Sets a trace flag on or off to log output to the Oracle Application Server Guard log files.

### Format

set trace on | off <traceflags>

### Parameters

**on | off**
Specifying "on" enables tracing. Specifying "off" disables tracing.

**traceflags**
The traceflags to be enabled. Two or more specified traceflags entries must be separated by a comma (,). The traceflags are as follows:

- ALL -- the most extensive tracing information available
- DB -- trace information regarding processing in the Oracle Database environment
- CLIPBOARD -- trace information regarding clipboard processing
- COPY -- trace information regarding file copy processing
- FLOW -- trace information regarding work flow processing
- HOME -- trace information with regard to Oracle homes
- IAS -- trace information regarding processing in Oracle Application Server
- IP -- trace information regarding network access and address translation
- NET -- trace information regarding network processing
- OPMN -- trace information regarding access to OracleAS OPMN calls
- RUNCMD -- trace information regarding the running of external commands
- SESSION -- trace information regarding session management
- TOPOLOGY -- trace information regarding processing of topology information

### Usage Notes

This command applies to all hosts that might be involved in an asgctl command during the lifetime of the connection.

The Oracle Application Server Guard client must be connected to a Oracle Application Server Guard server before using this command.

### Example

The following example turns on trace for database operations.

```
ASGCTL> set trace on db
```

# show env

Shows the current environment for the Oracle Application Server Guard server to which the Oracle Application Server Guard client is connected.

## Format

show env

## Parameters

None.

## Usage Notes

None.

## Example

The following examples show the environment of the Oracle Application Server Guard server to which the Oracle Application Server Guard client is connected. In the first example, the primary database and new primary database are not yet set on host prodinfra and in the second example, the primary database has already been set on host standbyinfra.

Example 1.

```
ASGCTL> show env

     ASG Server Connection:
        Host: prodinfra
        Port: 7890

     Primary database: <not set>
     New primary database:  <not set>
```

Example 2.

```
ASGCTL> ASGCTL> show env

     ASG Server Connection:
        Host: standbyinfra
        Port: 7890

Gathering information from the database orcl

     Primary database: :
        User: sys
        Service: orcl
        Role: The database role is
               PHYSICAL STANDBY


     New primary database:  <not set>
```

## show operation

Shows all operations on all nodes of the topology to which the Oracle Application Server Guard client is connected for the current session.

### Format

show op[eration] [full] [[his]tory]

### Parameters

**full**
For all operations, shows the operation number, the job name, the job owner's user name, the job ID, the time the operation began, the time the operation ended, the elapsed time for the operation, and all tasks belonging to this job.

**history**
For only operations that are not running, shows the operation number and the job name.

### Usage Notes

None.

### Example

The following examples show the status of the current operation.

```
ASGCTL> show operation
*************************************
OPERATION: 19
  Status: running
  Elapsed Time: 0 days, 0 hours, 0 minutes, 28 secs
  TASK: syncFarm
    TASK: backupFarm
      TASK: fileCopyRemote
      TASK: fileCopyRemote
    TASK: restoreFarm
      TASK: fileCopyLocal
```

The following example shows the history of all operations.

```
ASGCTL> show op his
*************************************
OPERATION: 7
  Status: success
  Elapsed Time: 0 days, 0 hours, 0 minutes, 0 secs
  TASK: getTopology
    TASK: getInstance
*************************************
OPERATION: 16
  Status: success
  Elapsed Time: 0
 days, 0 hours, 0 minutes, 0 secs
  TASK: getTopology
    TASK: getInstance
*************************************
OPERATION: 19
```

```
Status: success
Elapsed Time: 0 days, 0 hours, 1 minutes, 55 secs
TASK: syncFarm
  TASK: backupFarm
    TASK: fileCopyRemote
    TASK: fileCopyRemote
  TASK: restoreFarm
    TASK: fileCopyLocall
```

## shutdown

Shuts down a running Oracle Application Server Guard server to which the Oracle Application Server Guard client is connected. Use this command only on a host system where OPMN is not running and you are following the procedure to clone an instance or clone a topology.

### Format

shutdown [local]

### Parameters

**local**
When specified shuts down the Oracle Application Server Guard server of the local Oracle home of asgctl.

### Usage Notes

The Oracle Application Server Guard server must have been started using the asgctl startup command and not the OPMN opmnctl command startproc.

### Example

The following example shuts down the Oracle Application Server Guard server on a host system in which OPMN is not running.

```
> asgctl.sh shutdown
```

## shutdown topology

Shuts down the OracleAS component services across the topology, while Oracle Application Server Guard server and OPMN will continue to run.

### Format

shutdown topology

### Parameters

None.

### Usage Notes

This is a convenient command for shutting down the entire topology. Use the startup topology command to start it up again.

This command will shutdown OracleAS services such as OID, OC4J, WebCache, and so forth.

### Example

The following example shuts down the prodinfra production topology.

```
ASGCTL> shutdown topology
Generating default policy for this operation

prodinfra:7890
    Shutting down each instance in the topology

asmid2:7890 (home /private1/OraHome2/asmid2)
    Shutting down component HTTP_Server
    Shutting down component OC4J
    Shutting down component dcm-daemon
    Shutting down component LogLoader

asmid1:7890 (home /private1/OraHome/asmid1)
    Shutting down component HTTP_Server
    Shutting down component OC4J
    Shutting down component dcm-daemon
    Shutting down component LogLoader

prodinfra:7890 (home /private1/OraHome2/asr1012)
    Shutting down component OID
    Shutting down component HTTP_Server
    Shutting down component OC4J
    Shutting down component dcm-daemon
    Shutting down component LogLoader
ASGCTL>
```

## startup

Starts up an Oracle Application Server Guard server from the asgctl prompt. Use this command only on a host system where OPMN is not running and you are following the procedure to clone an instance or clone a topology.

### Format

startup [asg]

### Parameters

**asg**
Optional keyword acronym for Application Server Guard. This parameter has no other meaning other than to show a format similar to the connect asg and set asg credentials commands.

### Usage Notes

None.

### Example

The following example starts up the Oracle Application Server Guard server on a host system in which OPMN is not running.

```
> asgctl.sh startup
```

## startup topology

Starts up a shutdown topology by starting up the OracleAS component services across the topology.

### Format

startup topology

### Parameters

**none**

### Usage Notes

This is a convenient command for starting up the entire topology after it was shut down using the shutdown topology command.

This command will start up OracleAS services such as OID, OC4J, WebCache, and so forth. The startup topology command will perform the equivalent of an opmnctl startup command across each instance of the topology.

### Example

The following example starts up the production topology.

```
ASGCTL> startup topology
Generating default policy for this operation

profinfra:7890
    Starting each instance in the topology

prodinfra:7890 (home /private1/OraHome2/asr1012)
    Executing opmnctl startall command

asmid1:7890 (home /private1/OraHome/asmid1)
    Executing opmnctl startall command

asmid2:7890 (home /private1/OraHome2/asmid2)
    Executing opmnctl startall command
ASGCTL>
```

## stop operation

Stops a specific operation that is running on the server.

### Format

stop op[eration] <op #>

### Parameters

**op #**
The number of the operation.

### Usage Notes

The number of the operation that is running on the server can be determined from a show operation command.

### Example

The following example first shows the running operation (15) on the server and then the stop operation command stops this operation.

```
ASGCTL> show operation
*************************************
OPERATION: 15
  Status: running
  Elapsed Time: 0 days, 0 hours, 1 minutes, 35 secs
  TASK: instantiateFarm
    TASK: verifyFarm

ASGCTL> stop operation 15
```

## switchover topology

During a scheduled outage of the production site, performs the switchover operation from the production site to the standby site.

### Format

switchover topology to <standby_topology_host>[:<port>] [using policy <file>]

### Parameters

**standby_topology_host**
Name of the standby host system. This parameter is required because it directs the coordinating Oracle Application Server Guard server instance to discover the instances that make up the standby site. This host system must be a member of the standby site topology.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the standby host system for the Oracle Application Server Guard server in its Oracle home.

**using policy <file>**
Full path and file specification for the XML policy file.

### Usage Notes

On the primary infrastructure system, make sure the emagent process is stopped. Otherwise, you may run into the following error when doing a switchover operation because the emagent process has a connection to the database:

```
prodinfra: -->ASG_DGA-13051: Error performing a physical standby switchover.
prodinfra: -->ASG_DGA-13052: The primary database is not in the proper state to
perform a switchover.  State is "SESSIONS ACTIVE"
```

On UNIX systems, to stop the emagent process, stop the Application Server Control, which is called iasconsole, as follows:

```
> <ORACLE_HOME>/bin/emctl stop iasconsole
```

On UNIX systems, to check to see if there is an emagent process running, do the following:

```
> ps -ef | grep emagent
```

On UNIX systems, if after performing the stop iasconsole operation, the emagent process is still running, get its process ID (PID) as determined from the previous ps command and stop it as follows:

```
> kill -9 <emagent-pid>
```

On Windows systems, open the Services control panel. Locate the OracleAS10gASControl service and stop this service.

Make sure OracleAS Infrastructure database is running on the primary topology before performing a switchover operation. Also, the OracleAS Infrastructure database information must be set by using the set primary database asgctl command.

The global DNS names are used to direct the switchover. This will be different than the HA naming utilized in the OracleAS Disaster Recovery environment. The discovery mechanism automatically maps the topology to the corresponding peer, based off local name resolution.

As part of the Oracle Application Server Guard switchover operation, an implicit sync topology operation is performed to make sure the topologies are identical. In addition OPMN automatically starts the Oracle Application Server Guard server on the "new" standby Infrastructure node and this server will run indefinitely, and in turn, starts the Oracle Application Server Guard server on the other nodes in the "new" standby topology and each of these is a transient server.

For the switchover policy file, by default the success requirement attribute is set to optional for all instances (middle tier and OracleAS Metadata Repository) and mandatory for the Oracle Internet Directory home.

During a switchover operation, the `opmn.xml` file is copied from the primary site to the standby site. For this reason, the value of the TMP variable must be defined the same in the `opmn.xml` file on both the primary and standby sites, otherwise this switchover operation will fail with a message that it could not find a directory. Therefore, make sure the TMP variable is defined identically and resolves to the same directory structure on both sites before attempting a switchover operation.

When performing a switchover operation from a primary site with two Oracle Identity Management instances running (im.systemA.us.oracle.com and im.systemB.us.oracle.com) to a standby site representing an asymmetric topology with only one Oracle Identity Management instance running (im.systemA.us.oracle.com), meaning that the other node (im.systemB.us.oracle.com) is to be ignored on the switchover site, the system administrator must not only edit the `switchover_policy.xml` policy file to indicate that this other node is to be set to Ignore, but the system administrator must also shut down all processes running on that node (im.systemB.us.oracle.com) in order for the switchover operation to be successful.

When performing a switchover operation from a primary site with two middle tiers, for example core1 and core2 instances registered in the Oracle Internet Directory, to a standby site representing an asymmetric topology with only one middle tier core1, the standby site actually has both core1 and core2 middle tiers registered in the Oracle Internet Directory. The `switchover_policy.xml` policy file is edited to ignore the core2 middle tier that does not exist on the standby site during the switchover operation. However, it should be noted that the Oracle Internet Directory, which is stored in an Oracle database, is identical for both the production site topology and the standby site topology and therefore a core2 middle tier is also shown to be registered in the Oracle Internet Directory on the standby site topology. For this reason, you cannot install to that standby site topology the same core2 middle tier with the hope of making this into a symmetric topology again. This is a strict limitation for switchover operations using asymmetric standby topologies.

When the discover topology command is issued following a switchover operation and the asymmetric standby site topology originally had one or more fewer middle tiers (for example, instA and instB) than there were in the original production site topology (instA, instB, and instC), a warning error message displays for each missing instance of a middle tier (instC, in this case). This warning error message is expected and can be ignored. When a discover topology command is issued following a switchover operation, OracleAS Server Guard reads the Oracle Internet Directory information, which is an exact copy of the original primary site Oracle Internet Directory information on this new primary site (former standby site). Because this Oracle Internet Directory information is identical to the original primary site Oracle Internet Directory information, when OracleAS Server Guard visits the host/home of each instance of these middle tiers to verify their existence, it finds that some do not exist, and issues the warning.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example performs a switchover operation to a standby site known by DNS as standbyinfra.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
ASGCTL> switchover topology to standbyinfra
Generating default policy for this operation
prodinfra:7890
    Switchover each instance in the topology to standby topology
prodinfra:7890 (home /private1/OraHome2/asr1012)
    Connecting to the primary database asdb.us.oracle.com
    Gathering information from the primary database asdb.us.oracle.com
    Shutting down each instance in the topology
.
.
.
prodinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
standbyinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
prodinfra:7890
    Verifying that the topology is symmetrical in both primary and standby configuration
ASGCTL>

# Command to use if you are using a policy file
# switchover topology to standbyinfra using policy <file>
```

# sync topology

Synchronizes the standby site with the primary site to ensure that the two sites are consistent. The sync topology operation applies database redo logs for OracleAS Infrastructures to the standby site in conjunction with synchronizing external configuration files across the topology.

## Format

```
sync topology to <standby_topology_host>[:<port>] [full | incr[emental]] [using policy <file>]
```

## Parameters

**standby_topology_host**
Name of the standby host system. This parameter is required because it directs the coordinating Oracle Application Server Guard server instance to discover the instances that make up the standby site. This host system must be a member of the standby site topology.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the standby host system for the Oracle Application Server Guard server in its Oracle home.

**full | incremental**
The synchronization of the standby site with the primary site to make the standby site consistent can be either "full" or "incremental". The default is "incremental". By default, if a full backup has not been performed, an incremental backup operation will not be performed. Instead, a full backup operation will be performed.

**using policy <file>**
Full path and file specification for the XML policy file.

## Usage Notes

By default an incremental synchronization is performed to make the standby site consistent with the primary site, which offers the best performance. However, there may be three circumstances when specifying a full synchronization should be used.

- When you want to force a full synchronization to happen, such as synchronizing the standby site completely at a specific point in time (currently) with the primary site.
- When you know there are many transactional changes over a short period of time on the primary site that must be synchronized with the secondary site.

- When you know that there are a large accumulation of transactional changes over a long period of time on the primary site that must be synchronized with the secondary site.

The sync operation performs an implicit verify operation.

For the sync policy file, by default the success requirement attribute is set to mandatory for all instances.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example synchronizes the specified standby site with the coordinating Oracle Application Server Guard server (the primary site). By default the sync mode is incremental.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> sync topology to standbyinfra
Generating default policy for this operation
prodinfra:7890
    Synchronizing each instance in the topology to standby topology
prodinfra:7890 (home /private1/OraHome2/asr1012)
    Starting backup of topology ""
       Backing up and copying data to the standby topology
    Backing up each instance in the topology
    Starting backup of instance "asr1012.infra.us.oracle.com"
    Configuring the backup script
asmid1:7890 (home /private1/OraHome/asmid1)
    Starting backup of instance "asmid1.asmid1.us.oracle.com"
asmid2:7891 (home /private1/OraHome/asmid2)
    Starting backup of instance "asmid2.asmid2.us.oracle.com"
.
.
.
asmid2:7890 (home /private1/OraHome2/asr1012)
    Starting backup/synchronization of database "asdb.us.oracle.com"
    Starting restore/synchronization of database "asdb.us.oracle.com"
    Synchronizing topology completed successfully
ASGCTL>

# Command to use if you are using a policy file
# sync topology to standbyinfra using policy <file>
```

## verify topology

Validates that the primary topology is running and the configuration is valid. If a standby topology is specified, compares the primary topology to which the local host system is a member with the standby topology to validate that they are consistent with one another and conform to the requirements for OracleAS Disaster Recovery.

### Format

```
verify topology [with <host>[:<port>]] [using policy <file>]
```

### Parameters

**host**
Name of the standby host system. This host system must be a member of the standby topology.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the host system for the Oracle Application Server Guard server in its Oracle home.

**using policy <file>**
Full path and file specification for the XML policy file.

### Usage Notes

If the host system name is not specified, the topology in which the local host system participates will be verified for local OracleAS Disaster Recovery rules.

If the standby host system name is specified, the topology at the standby site will be verified along with the production topology for both local rules and distributed OracleAS Disaster Recovery rules, and the symmetry between the primary and standby sites is also checked.

For the verify policy file, by default the success requirement attribute is set to optional for all OracleAS homes (middle tier and OracleAS Metadata Repository) and mandatory for the Oracle Internet Directory home.

See Section 5.1, "Information Common to Oracle Application Server Guard asgctl Commands" and Section 5.2, "Information Specific to a Small Set of Oracle Application Server Guard Commands" for more information.

## Example

The following example validates that the primary topology is running and the configuration is valid.

```
ASGCTL> connect asg ias_admin/iastest2
Successfully connected to prodinfra:7890
ASGCTL> verify topology
Generating default policy for this operation
prodinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
ASGCTL>
```

The following example validates that the topology to which the local host system is a member is consistent with the standby topology to which the host system standbyinfra is a member.

```
ASGCTL> connect asg prodinfra ias_admin/adminpwd
Successfully connected to prodinfra:7890
ASGCTL> set primary database sys/testpwd@asdb
Checking connection to database asdb
ASGCTL> verify topology with standbyinfra
Generating default policy for this operation
prodinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
standbyinfra:7890
     HA directory exists for instance asr1012.infra.us.oracle.com
asmid2:7890
     HA directory exists for instance asmid2.asmid2.us.oracle.com
asmid1:7890
     HA directory exists for instance asmid1.asmid1.us.oracle.com
prodinfra:7890
    Verifying that the topology is symmetrical in both primary and standby configuration
ASGCTL>

# Command to use if you are using a policy file
# verify topology using policy <file>
```

# dump farm (Deprecated)

Directs asgctl to write detailed information about the farm to the specified file.

> **Note:** The dump farm command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the dump topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

## Format

```
dump farm [to <file>]
```

## Parameters

**to <file>**
Name of file on the Oracle Application Server Guard client node where the detailed output is to be written.

## Usage Notes

None.

## Example

See the dump topology command for an example.

## instantiate farm (Deprecated)

Instantiates a farm to a standby site by discovering the current farm definition at the production and standby sites, verifying that each complies with the OracleAS Disaster Recovery rules and restrictions of the current OracleAS software deployed on these systems prior to creation. Also synchronizes the standby site with the primary site so that the primary and standby sites are consistent.

> **Note:** The instantiate farm to command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the instantiate topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

### Format

```
instantiate farm to <standby_farm_host>[:<port>]
```

### Parameters

**standby_farm_host**
Name of the standby host system. This parameter is required because it directs the coordinating Oracle Application Server Guard server instance to discover the instances that make up the standby site. This host system must be a member of the standby site topology.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the Oracle Application Server Guard server in its Oracle home.

### Usage Notes

The production local system must be part of an Oracle Notification Server (ONS) farm for the site.

The standby host must be part of an ONS farm for the standby site and must be symmetrical to the farm of the production farm.

Make sure OracleAS Infrastructure database is running on the primary farm before performing an instantiating farm operation. Also, the OracleAS Infrastructure database information must be set by using the set primary database asgctl command.

The global DNS names are used to direct the instantiation. This will be different than the HA naming utilized in the OracleAS Disaster Recovery environment. The discovery mechanism automatically maps the farm to the corresponding peer, based off local name resolution.

## Example

See the instantiate topology command for an example.

## shutdown farm (Deprecated)

Shuts down a running farm.

> **Note:** The shutdown farm command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the shutdown topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

### Format

```
shutdown farm
```

### Parameters

None.

### Usage Notes

This is a convenient command for shutting down the entire farm. Use the startup farm command to start it up again.

### Example

See the shutdown topology command for an example.

## startup farm (Deprecated)

Starts up a shutdown farm.

> **Note:** The startup farm command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the startup topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

### Format

```
startup farm
```

### Parameters

None

### Usage Notes

This is a convenient command for starting up the entire farm after it was shut down using the shutdown farm command.

### Example

See the startup topology command for an example.

## switchover farm (Deprecated)

During a scheduled outage of the production site, performs the switchover operation from the production site to the standby site.

> **Note:** The switchover farm to command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the switchover topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

### Format

switchover farm to <standby_farm_host>[:<port>]

### Parameters

**standby_farm_host**
Name of the farm host system. This parameter is required because it directs the coordinating Oracle Application Server Guard server instance to discover the instances that make up the standby site. This host system must be a member of the standby farm.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the standby host system for the Oracle Application Server Guard server in its Oracle home.

### Usage Notes

On the primary Infrastructure system, make sure the emagent process is stopped. Otherwise, you may run into the following error when doing a switchover operation because the emagent process has a connection to the database:

```
prodinfra: -->ASG_DGA-13051: Error performing a physical standby switchover.
prodinfra: -->ASG_DGA-13052: The primary database is not in the proper state to
perform a switchover.  State is "SESSIONS ACTIVE"
```

On UNIX systems, to stop the emagent process, stop the Application Server Control, which is called iasconsole, as follows:

```
> <ORACLE_HOME>/bin/emctl stop iasconsole
```

On UNIX systems, to check to see if there is an emagent process running, do the following:

```
> ps -ef | grep emagent
```

On UNIX systems, if after performing the stop iasconsole operation, the emagent process is still running, get its process ID (PID) as determined from the previous ps command and stop it as follows:

```
> kill -9 <emagent-pid>
```

On Windows systems, open the Services control panel. Locate the OracleAS10gASControl service and stop this service.

The production local system must be part of an Oracle Notification Server (ONS) farm for the site.

The standby host must be part of an ONS farm for the standby site and must be symmetrical to the farm of the production farm.

Make sure OracleAS Infrastructure database is running on the primary farm before performing a switchover operation. Also, the OracleAS Infrastructure database information must be set by using the set primary database asgctl command.

The global DNS names are used to direct the switchover. This will be different than the HA naming utilized in the OracleAS Disaster Recovery environment. The discovery mechanism automatically maps the farm to the corresponding peer, based off local name resolution.

As part of the Oracle Application Server Guard switchover operation, an implicit sync farm operation is performed to make sure the farms are identical. In addition, OPMN automatically starts the Oracle Application Server Guard server on the "new" standby Infrastructure node and this server will run indefinitely. In turn, it starts the Oracle Application Server Guard server on the other nodes in the "new" standby farm and each of these is a transient server.

## Example

See the switchover topology command for an example.

## sync farm (Deprecated)

Synchronizes the standby site with the primary site to ensure that the two sites are consistent. The sync topology operation applies database redo logs for OracleAS Infrastructures to the standby site in conjunction with synchronizing external configuration files across the topology.

> **Note:** The sync farm to command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the sync topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

### Format

```
sync farm to <standby_farm_host>[:<port>] [full | incr[emental]]
```

### Parameters

**standby_farm_host**
Name of the standby site host system. This parameter is required because it directs the coordinating Oracle Application Server Guard server instance to discover the instances that make up the standby site. This host system must be a member of the standby farm.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the standby host system for the Oracle Application Server Guard server in its Oracle home.

**full | incremental**
The synchronization of the standby site with the primary site to make the standby site consistent can be either "full" or "incremental". The default is "incremental". By default, if a full backup has not been performed, an incremental backup operation will not be performed. Instead, a full backup operation will be performed.

### Usage Notes

By default `sync_mode` is incremental and offers the best performance. However, there may be three circumstances when specifying a `sync_mode` of full should be used.

- When you want to force a full synchronization to happen, such as synchronizing the standby site completely at a specific point in time (currently) with the primary site.

- When you know there are many transactional changes over a short period of time on the primary site that must be synchronized with the secondary site.
- When you know that there is a large accumulation of transactional changes over a long period of time on the primary site that must be synchronized with the secondary site.

## Example

See the sync topology command for an example.

# verify farm (Deprecated)

Validates that the primary farm is running and the configuration is valid. If a standby farm is specified, compares the primary farm to which the local host system is a member with the standby farm to validate that they are consistent with one another and conform to the requirements for OracleAS Disaster Recovery.

> **Note:** The verify farm command is deprecated beginning with OracleAS release 10.1.2.0.2. Use the verify topology command, which supports the OracleAS Disaster Recovery topology concept in current and future OracleAS releases.

## Format

```
verify farm [with <host>[:<port>]]
```

## Parameters

**host**
Name of the standby host system. This host system must be a member of the standby farm.

In the Disaster Recovery environment, if this host has a physical hostname, use the physical hostname for the host.

If the host does not have a physical hostname (for example, if it is an Infrastructure host), use its network hostname.

> **Note:** See Section 1.2.1, "Planning and Assigning Hostnames" for more information about planning and assigning physical hostnames, network hostnames, virtual hostnames, and virtual hostname aliases for hosts in your Disaster Recovery environment.

**port**
The port number of the Oracle Application Server Guard server in its Oracle home.

## Usage Notes

If the host system name is not specified, the farm in which the local host system participates will be verified for local OracleAS Disaster Recovery rules.

If the standby host system name is specified, the farm at the standby site will be verified along with the production farm for both local rules and distributed OracleAS Disaster Recovery rules, and the symmetry between the primary and standby sites is also checked.

## Example

See the verify topology command for an example.

# 6

# Setting Up a DNS Server

This chapter provides instructions on setting up a DNS server in UNIX. These instructions are applicable for setting up the site-specific DNS zones used for hostname resolution in the example in Figure 1–9, "DNS Resolution Topology Overview".

> **Note:** The DNS setup information provided in this chapter is an example to aid in the understanding of OracleAS Disaster Recovery operations. It is generic to DNS, and other appropriate DNS documentation should be consulted for comprehensive DNS information.

For the discussion in this chapter, the DNS server that is set up creates and services a new DNS zone with the unique domain `oracleas`. Within the zone, this DNS server resolves all requests for the `oracleas` domain and forwards other requests to the overall wide area company DNS server(s).

On the UNIX host that will act as the DNS zone server, perform the following steps:

1. Create the name server configuration file `/var/named.conf`. Assuming the wide area company DNS server IP address is 123.1.15.245, the contents of this file should be as follows:

```
options {
          directory "/var/named";
          forwarders {
           123.1.15.245;
         };
};

zone "." in {
              type hint;
              file "named.ca";
};

zone "oracleas" {
                  type master;
                  file "oracleas.zone";
};

zone "0.0.127.IN-ADDR.ARPA {
                           type master;
                           file "127.zone";
};
```

2. Create the root hint file `/var/named/named.ca`, which has the following contents (123.1.2.117 is the IP of the zone DNS server):

   ```
   .        999999   IN    NS    ourroot.private.
   ourroot.private.   IN    A     123.1.2.117
   ```

3. Create the loopback address file `/var/named/127.zone`, which has the following contents (assume the zone DNS server's hostname is `aszone1`):

   ```
   $ORIGIN    0.0.127.IN-ADDR.ARPA.
   0.0.127.IN-ADDR.ARPA.    IN    SOA  aszone1.oracleas.  root.aszone1.oracleas.
   (
              25           ; serial number
              900          ; refresh
              600          ; retry
              86400        ; expire
              3600       ) ; minimum TTL

   0.0.127.IN-ADDR.ARPA.    IN    NS    aszone1.oracleas.
   1                        IN    PTR   localhost.oracleas.
   ```

4. Create the zone data file `/var/named/oracleas.dns`, which has the following contents (values shown are applicable to the example of the production site in Figure 1–9):

   ```
   ;
   ;  Database file oracleas.dns for oracleas zone.
   ;    Zone version:  25
   ;
   $ORIGIN oracleas.
   oracleas.       IN    SOA    aszone1.oracleas.  root.aszone1.oracleas (
                   25        ; serial number
                   900       ; refresh
                   600       ; retry
                   86400     ; expire
                   3600    ) ; minimum TTL


   ;
   ;    Zone NS records
   ;
   oracleas.       IN        NS    aszone1.oracleas.

   ;
   ;    Zone records
   ;
   localhost       IN     A     127.0.0.1

   asmid1          IN     A     123.1.2.333
   asmid2          IN     A     123.1.2.334
   infra           IN     A     123.1.2.111
   remoteinfra     IN     A     213.2.2.210
   ```

5. Run the following command to start the name server:

   ```
   /sbin/in.named
   ```

6. On all the hosts in the domain that is serviced by this DNS server, edit the `domain` and `nameserver` settings in the file `/etc/resolv.conf` as follows (all previous `nameserver` settings should be removed; 123.1.2.117 is assumed to the zone DNS server's IP address):

```
domain    oracleas
nameserver 123.1.2.117
```

# 7 Secure Shell (SSH) Port Forwarding

This chapter describes how secure shell (SSH) port forwarding may be used with Oracle Data Guard.

## 7.1 SSH Port Forwarding

Oracle Application Server Guard automates the use of Oracle Data Guard, which sends redo data across the network to the standby system using Oracle Net. SSH tunneling may be used with Oracle Data Guard as an integrated way to encrypt and compress the redo data before it is transmitted by the production system and subsequently decrypt and uncompress the redo data when it is received by the standby system.

**See Also:**

- Implementing SSH port forwarding with Data Guard: https://metalink.oracle.com/metalink/plsql/showdoc?db=NOT&id=225633.1
- Troubleshooting Data Guard network issues: https://metalink.oracle.com/metalink/plsql/showdoc?db=NOT&id=241925.1

# A

# Troubleshooting High Availability

This appendix describes common problems that you might encounter when deploying and managing Oracle Application Server in high availability configurations, and explains how to solve them. It contains the following topics:

- Section A.1, "Troubleshooting OracleAS Disaster Recovery Topologies"
- Section A.2, "Troubleshooting Middle-Tier Components"
- Section A.3, "Need More Help?"

## A.1 Troubleshooting OracleAS Disaster Recovery Topologies

This section describes common problems and solutions in OracleAS Disaster Recovery configurations. It contains the following topics:

- Section A.1.1, "Changing the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard"
- Section A.1.2, "Failure to Bring Up Standby Instances After Failover or Switchover"
- Section A.1.3, "Switchover Operation Fails At the Step dcmctl resyncInstance -force -script"
- Section A.1.4, "An Oracle Application Server Guard asgctl verify Operation Does Not Check Temp Directories"
- Section A.1.5, "Unable to Start Standalone OracleAS Web Cache Installations at the Standby Site"
- Section A.1.6, "Standby Site Middle-tier Installation Uses Wrong Hostname"
- Section A.1.7, "Failure of Farm Verification Operation with Standby Farm"
- Section A.1.8, "Sync Farm Operation Returns Error Message"
- Section A.1.9, "On Windows Systems Use of asgctl startup Command May Fail If the PATH Environment Variable Has Exceeded 1024 Characters"
- Section A.1.10, "Adding an Instance from a Remote Client Adds an Instance on the Local Instance and Not on the Remote Instance"
- Section A.1.11, "Oracle Application Server Guard Returns an Inappropriate Message When It Cannot Find the User Specified Database Identifier"
- Section A.1.12, "Database Instance on Standby Site Must Be Shut Down Before Issuing an asgctl create standby database Command"
- Section A.1.13, "Known Issue with Disaster Recovery Cloning on Windows"

## A.1.1 Changing the Default Oracle Data Guard Configuration Set Up by Oracle Application Server Guard

In some cases, you may want to change the default Data Guard configuration that is set up by Oracle Application Server Guard.

### Problem

For example, if you are running Oracle BPEL Process Manager in an OracleAS Disaster Recovery topology, you want to ensure that:

- The Oracle BPEL Process Manager dehydration data is stored in databases that are included in the OracleAS Disaster Recovery topology. This ensures that when a Disaster Recover switchover or failover operation is performed, the database and related services are switched over or failed over in coordination with the Oracle Application Server services in the Disaster Recovery topology.
- The Oracle BPEL Process Manager dehydration data stored in databases at the primary and standby sites are continuously synchronized.
- When a switchover operation or failover operation occurs, Oracle BPEL Process Manager uses the database at the standby site.

### Solution

To achieve this:

1. Store the dehydration data in a database.

> **Note:** If you create the standby database using the asgctl `create standby database` command, then the following two steps will be performed for you by the `create standby database` command.

2. Set the Oracle Data Guard data protection mode for the primary database to maximum availability mode (instead of maximum protection mode). Using maximum availability mode allows logs to be applied continuously at the standby site without shutting down the primary database if the standby database is taken offline.

   Run this command on the primary database:

   ```
   SQL> alter database set standby database to maximize availability;
   ```

3. Place the standby database in managed recovery mode. This puts the standby database in a constant state of media recovery. Configuring the standby database for managed recovery is not a requirement of maximum availability, but it provides for shorter failover times.

   On the standby database, run the following command to place the standby database in managed recovery mode. Add the optional `disconnect from session` clause if you want to end the session after the command:

   ```
   SQL> alter database recovery managed standby database disconnect from session;
   ```

These steps change the Oracle Data Guard protection mode of the primary database from maximum performance to maximum availability. For details on the different Oracle Data Guard protection modes (maximum protection, maximum availability, and maximum performance), see *Oracle Data Guard Concepts and Administration* in the Oracle Database documentation set.

Running the primary database in maximum availability mode may cause a hang waiting for an available online log file. A maximum availability primary database will not reuse an online log file until it has been archived to the standby database. This could happen if the standby database is taken offline for a long time.

Only data with same synchronization requirements should be stored in the same database. For example, the Oracle BPEL Process Manager dehydration store and the OracleAS Portal data should be stored in separate databases because the synchronization objectives of Oracle BPEL Process Manager and OracleAS Portal are different. The synchronization objective of Oracle BPEL Process Manager dehydration store is to maintain consistency between the dehydration store and the BPEL process, while the synchronization objective of OracleAS Portal is to ensure that data and configuration maintained within the middle tier and database do not diverge.

### Actions Performed by the sync topology Command

When the primary database is in maximum availability mode and the standby database is in managed recovery mode, the asgctl `sync topology` command does the following:

- Performs a log switch at the primary and ensures that the log is shipped and archived.

- Performs process management at the primary and standby sites.
- Encapsulates the incremental changes for all the data in the Oracle homes.
- Restores the standby peers to the configuration level of the primary.
- Propagates the changes to all standby instances.
- For standby databases:
  - With managed recovery running, the `sync topology` command simply reports the sync SCN and the current database SCN of the standby database. For this configuration, the standby database SCN is guaranteed to be beyond the sync SCN. ASG logs the sync SCN level as it corresponds to the current SCN level of the standby database.
  - Without managed recovery running, the `sync topology` command recovers the standby database to the sync SCN. It is equivalent to running the following command:

    ```
    alter database recover managed standby database until change <sync-scn>
    ```

### A.1.2 Failure to Bring Up Standby Instances After Failover or Switchover

Standby instances are not started after a failover or switchover operation.

**Problem**

IP addresses are used in instance configuration. OracleAS Disaster Recovery setup does not require identical IP addresses in peer instances between the production and standby site. OracleAS Disaster Recovery synchronization does not reconcile IP address differences between the production and standby sites. Thus, if you use explicit IP address xxx.xx.xxx.xx in your configuration, the standby configuration after synchronization will not work.

**Solution**

Avoid using explicit IP addresses. For example, in OracleAS Web Cache and Oracle HTTP Server configurations, use ANY or host names instead of IP addresses as listening addresses

### A.1.3 Switchover Operation Fails At the Step dcmctl resyncInstance -force -script

The OracleAS Disaster Recovery asgctl switchover operation requires that the value of the TMP environment variable be defined the same in the `opmn.xml` file on both the primary and standby sites.

**Problem**

OracleAS Disaster Recovery switchover fails at the step dcmctl resyncInstance -force -script and displays a message that a directory could not be found.

**Solution**

During a switchover operation, the opmn.xml file is copied from the primary site to the standby site. For this reason, the value of the TMP variable must be defined the same in the `opmn.xml` file on both primary and standby sites; otherwise, the switchover operation will fail. Make sure the TMP variable is defined identically in the opmn.xml files and resolves to the same directory structure on both sites before attempting to perform an asgctl switchover operation.

For example, the following code snippets for a Windows and UNIX environment show a sample definition of the TMP variable.

```
Example in Windows Environment:
-------------------------------
.
.
.
<ias-instance id="infraprod.iasha28.us.oracle.com">
 <environment>
 <variable id="TMP" value="C:\DOCUME~1\ntregres\LOCALS~1\Temp"/>
 </environment>
.
.
.
Example in UNIX Environment:
----------------------------
.
.
.
<ias-instance id="infraprod.iasha28.us.oracle.com">
 <environment>
 <variable id="TMP" value="/tmp"/>
 </environment>
.
.
.
```

A workaround to this problem is to change the value of the TMP variable in the `opmn.xml` file on the primary site, perform a dcmctl `update config` operation, then perform the asgctl switchover operation. This approach saves you having to reinstall the mid-tiers to make use of an altered TMP variable.

### A.1.4 An Oracle Application Server Guard asgctl verify Operation Does Not Check Temp Directories

The same TEMP directory structure that exists on a primary site must be set up on the standby site.

**Problem**

DCM does not work properly when the same TEMP directory structure that exists on a primary site is not set up on the standby site. An Oracle Application Server Guard verify operation does not detect this problem.

**Solution**

Maintain the same TEMP directories on both the primary and standby sites. When creating environment variables for the standby site, ensure that each standby peer's environment is a replica of the production home. An area that is commonly forgotten or overlooked is the TEMP directory.

### A.1.5 Unable to Start Standalone OracleAS Web Cache Installations at the Standby Site

OracleAS Web Cache cannot be started at the standby site possibly due to misconfigured. This is applicable only for 10.1.2.x and 10.1.4.x environments, and not for 10.1.3.x environments.

#### Problem

OracleAS Disaster Recovery synchronization does not synchronize standalone OracleAS Web Cache installations.

#### Solution

Use the standard Oracle Application Server full CD image to install the OracleAS Web Cache component

### A.1.6 Standby Site Middle-tier Installation Uses Wrong Hostname

A middle-tier installation in the standby site uses the wrong hostname even after the system's physical hostname is changed.

#### Problem

Depending on the Oracle Application Server installation, there are different methods of specifying a physical hostname. Before performing an Oracle Application Server installation, you must use the appropriate method or methods of specifying a physical hostname for that Oracle Application Server release to ensure that the installer uses the correct physical hostname.

#### Solution

Section 1.2.1.1, "Physical Hostnames"and its subsections provide the instructions for creating a physical hostname prior to installing an instance for a particular Oracle Application Server release. Follow the instructions for the Oracle Application Server release you are installing.

### A.1.7 Failure of Farm Verification Operation with Standby Farm

When performing a verify farm with standby farm operation, the operation fails with an error message indicating that the middle-tier system instance cannot be found and that the standby farm is not symmetrical with the production farm.

#### Problem

The verify farm with standby farm operation is trying to verify that the production and standby farms are symmetrical to one another, that they are consistent, and conform to the requirements for disaster recovery.

One part of the verify operation is a check to confirm that hostname resolution is the same for the hosts at the production and standby site.

For example, suppose that the `/etc/hosts` file for node1 at the production site has this entry for node1:

```
123.45.67.890 node1.us.oracle.com node1 infra
```

In this case, the entries for node1 in other `/etc/hosts` files in the topology should also have `node1.us.oracle.com` in the second column of the entry. For example, this would be a valid entry for node1 in the `etc/hosts` file for node1 at the standby site:

```
123.45.68.891 node1.us.oracle.com node1 infra
```

### Solution

All of the /etc/hosts file entries for a particular host must have the same name in the second column of the /etc/hosts file entry for the host. Otherwise, the verify operation will not succeed.

## A.1.8 Sync Farm Operation Returns Error Message

A sync farm to operation returns the error message: "Cannot Connect to asdb"

### Problem

Occasionally, an administrator may forget to set the primary database using the asgctl command line utility in performing an operation that requires that the asdb database connection be established prior to an operation. The following example shows this scenario for a sync farm to operation:

```
ASGCTL> connect asg hsunnab13 ias_admin/iastest2
Successfully connected to hsunnab13:7890
ASGCTL>
.
.
.
<Other asgctl operations may follow, such as verify farm, dump farm,
<and show operation history, and so forth that do not require the connection
<to the asdb database to be established or a time span may elapse of no activity
<and the administrator may miss performing this vital command.
.
.
.
ASGCTL> sync farm to usunnaa11
prodinfra(asr1012): Syncronizing each instance in the farm to standby farm
prodinfra: -->ASG_ORACLE-300: ORA-01031: insufficient privileges
prodinfra: -->ASG_DUF-3700: Failed in SQL*Plus executing SQL statement:  connect
null/******@asdb.us.oracle.com as sysdba;.
prodinfra: -->ASG_DUF-3502: Failed to connect to database asdb.us.oracle.com.
prodinfra: -->ASG_DUF-3504: Failed to start database asdb.us.oracle.com.
prodinfra: -->ASG_DUF-3027: Error while executing Syncronizing each instance in
the farm to standby farm at step - init step.
```

### Solution

Perform the asgctl set primary database command. This command sets the connection parameters required to open the asdb database in order to perform the sync farm to operation. Note that the set primary database command must also precede the instantiate farm to command and switchover farm to command if the primary database has not been specified in the current connection session.

## A.1.9 On Windows Systems Use of asgctl startup Command May Fail If the PATH Environment Variable Has Exceeded 1024 Characters

On Windows systems, if your system PATH environment variable has exceeded the 1024 character limit because you have many Oracle Application Server instances installed or many third party software installations, or both on your system, the asgctl startup command may fail because you are starting the Oracle Application Server Guard server outside of OPMN and the system cannot resolve the directory path.

### Problem

Occasionally, on Windows systems with many installations, Oracle Application Server instances or third party software, or both, the asgctl startup command, which is run outside of OPMN, may return a popup error stating it could not find a dynamic link library for a particular file, `orawsec9.dll`, followed by a DufException. For example:

```
C:\product\10.1.3\OC4J_1\dsa\bin> asgctl startup
<<Popup Error:>>
The dynamic link library *orawsec9.dll* could not be found.
<<The exception:>>
oracle.duf.DufException
        at oracle.duf.DufOsBase.constructInstance(DufOsBase.java:1331)
        at oracle.duf.DufOsBase.getDufOs(DufOsBase.java:122)
        at
oracle.duf.DufHomeMgr.getCurrentHomePath(DufHomeMgr.java:582)
        at oracle.duf.dufclient.DufClient.main(DufClient.java:132)
stado42: -->ASG_SYSTEM-100: oracle.duf.DufException
----------------------------------------------------------------------------
```

However, this dll does exist in the ORACLE_HOME\bin directory.

This error is not seen in Oracle Application Server Guard standalone kit because the file `orawsec9.dll` exists in the ORACLE_HOME\dsa\bin folder.

### Solution

The workaround is to either manually edit the system PATH variable with the required path information or manually override the PATH in the command prompt by specifying the relevant %PATH% variables. For example:

```
C:\set PATH=C:\product\10.1.3\OracleAS_OC4J_2\bin;
C:\product\10.1.3\OracleAS_OHS1\jre\1.4.2\bin\client;
C:\product\10.1.3\OracleAS_OHS1\jre\1.4.2\bin;
C:\product\10.1.3\OracleAS_OHS1\bin;C:\product\10.1.3\OC4J_1\bin

C:\product\10.1.3\OC4J_1\dsa\bin> asgctl startup
```

## A.1.10 Adding an Instance from a Remote Client Adds an Instance on the Local Instance and Not on the Remote Instance

When using the asgctl `add instance` command, the Oracle Application Server Guard client must be run from a system that is already included in the topology.

### Problem

For example, when an Oracle Application Server Guard client is connected to the Oracle Application Server Guard server that is to be added to an existing topology, the following error is returned:

```
ASG_IAS-15785: ERROR: The topology is missing the instance that exists in the home
where the ASG server is running.
You must first discover or add the instance in home
ASGCTL>
```

### Solution

Use an Oracle Application Server Guard client from a system that is already included in the topology to perform the asgctl `add instance` command to add an instance to the topology.

### A.1.11 Oracle Application Server Guard Returns an Inappropriate Message When It Cannot Find the User Specified Database Identifier

When adding an Oracle RAC instance to the topology using the Oracle Application Server Guard the `add instance` command and Oracle Application Server Guard cannot find the user specified identifier, an inappropriate error message is returned. If the user entered the database name rather that the Oracle instance SID, there is no indication that this is the problem.

**Problem**

If Oracle Application Server Guard is unable to locate the oratab entry (on UNIX) or the system registry service (on Windows) for the user specified database identifier, the following ASG_SYSTEM-100 message now precedes the existing ASG_DUF-3554 message and both messages will be displayed to the console:

On UNIX systems:

```
ASG_SYSTEM-100: An Oracle database is identified by its database unique name (db_
name)
ASG_DUF-3554: The Oracle home that contains SID <user specified identifier> cannot
be found
```

On Windows systems:

```
ASG_SYSTEM-100: An Oracle database is identified by its system identifier (SID)
ASG_DUF-3554: The Oracle home that contains SID <user specified identifier> cannot
be found
```

**Solution**

When you encounter the message shown in the preceding example, be sure you entered the Oracle instance SID, not the database name.

### A.1.12 Database Instance on Standby Site Must Be Shut Down Before Issuing an asgctl create standby database Command

You must shut down a standby site database instance if it is running in order for the asgctl `create standby database` command to succeed.

**Problem**

If you run the asgctl `create standby database` command without shutting down the database on the standby site, the following error is returned:

```
ASG_DGA-12500: Standby database instance "<instance_name>" already exists on host
"<hostname>"
```

**Solution**

Shut down the database on the standby site if it is up and running before issuing the asgctl `create standby database` command.

### A.1.13 Known Issue with Disaster Recovery Cloning on Windows

For the Windows platform, you must add the directory that contains the jar utility to the PATH when installing a JDK on the standby system.

### Problem

If you do not add the directory that contains the jar utility to the PATH when installing a JDK on the standby system, the ASG on the standalone system cannot access the jar.exe utility, and you receive the following error while cloning:

```
standbynode: -->ASG_SYSTEM-100: operable program or batch file.
standbynode: -->ASG_DUF-4040: Error executing the external program or script.
The error code is "1"
standbynode: -->ASG_IAS-15690: Error running the restore script
standbynode: -->ASG_IAS-15698: Error during backup topology operation - copy step
standbynode: -->ASG_DUF-3027: Error while executing Clone Instance at step -
unpack step.
```

### Solution

If you receive this error, add the jar utility to the PATH on the standby system and restart the ASG server.

## A.1.14 The asgctl shutdown topology Command Does Not Shut Down an MRCA Database That is Detected To Be of a repCA Type Database

The asgctl `shutdown topology` command only handles non-database instances.

### Problem

In a repCA environment when Oracle Application Server Guard detects an instance and determines it to be a repCA type database, its instance is ignored in a `shutdown topology` operation. Any repCA type database is considered to be managed outside of Oracle Application Server Guard. Therefore, within an environment where an MRCA database has been added to the topology, the database will not be handled by the asgctl `shutdown topology` command.

### Solution

Shut down any repCA type database by alternative methods other than the asgctl `shutdown topology` command.

## A.1.15 Connecting to an Oracle Application Server Guard Server May Return an Authentication Error

An authentication error occurs when trying to connect to an Oracle Application Server Guard 10.1.2.0.2 or 10.1.2.1 server, even though the correct user name and password were entered.

### Problem

When a user connects to an Oracle Application Server Guard server and gets an authentication error even though the correct user name and password were entered.

> **Note:** This DSA configuration file parameter is not documented in the "Oracle Application Server Guard Configuration File Parameters" section of the Oracle Application Server Guard Release Information `readme.txt` file.

#### Solution

Put the following flag in the `dsa.conf` file in the *<ORACLE_HOME>*`/dsa` directory and try the operation again:

```
_realm_override=1
```

### A.1.16 Running Instantiate Topology Across Nodes After Executing a Failover Operation Results in an ORA-01665 Error

After running the asgctl `failover` operation, you must first perform an asgctl `create standby database` command to create the standby database on the remote host before performing an asgctl `instantiate topology` operation.

#### Problem

If you attempt to perform an asgctl instantiate topology operation immediately following an asgctl failover operation, an "ORA-01665: control file is not a standby control file" error message is returned.

#### Solution

To work around this problem, you must first perform an asgctl create standby database command to create the standby database on the remote host.

### A.1.17 Oracle Application Server Guard Is Unable to Shutdown the Database Because More Than One Instance of Oracle RAC is Running

When you are running Oracle Application Server Guard in an Oracle RAC environment, you should have only one Oracle RAC instance running while performing Oracle Application Server Guard operations.

#### Problem

If you have more than one Oracle RAC instance running while performing Oracle Application Server Guard operations, an error occurs where the primary database complains that it is mounted by more than one instance, which prevents a shutdown. As a result, the following error will be seen:

```
ASGCTL> create standby database orcl1 on stanb06v3
.
.
.
This operation requires the database to be shutdown. Do you want to
continue? Yes or No
y
Database must be mounted exclusive
stanb06v1: -->ASG_DUF-4950: An error occurred on host "stanb06v1" with IP
"141.86.22.32" and port "7890"
stanb06v1: -->ASG_DUF-3514: Failed to stop database orcl1.us.oracle.com.
stanb06v1: -->ASG_DGA-13002: Error during Create Physical Standby:
Prepare-primary processing.
stanb06v1: -->ASG_DUF-3027: Error while executing Creating physical standby
database - prepare phase at step - primary processing step.
```

#### Solution

Be sure to have only one Oracle RAC instance running while performing Oracle Application Server Guard operations

## A.1.18 Create Standby Fails if Initiated on a Different ASGCTL Shell

The `create standby database` command fails if initiated by ASG clients from any node other than the source primary node where the database resides.

### Problem

If you ran the `create standby` command from the production database to the standby database where prodnode1 is the primary site database nodename and standbynode1 is its standby database nodename. The `ASGCTL shell` should always be invoked and connected to prodnode1. If you try to run `ASGCTL shell` from standbynode1 and connect to prodnode1, the `create standby` command fails.

### Solution

Run the `create standby` command from the same primary (source) node, where the database for the primary site resides.

## A.1.19 Resolve Missing Archived Logs

The `sync topology` command in a RAC-RAC Linux environment returns missing archive logs errors.

### Problem

The `sync topology` command in a RAC-RAC Linux environment fails and returns missing archive logs errors such as the following:

```
ASG_SYSTEM_-100: Please resolve missing archived logs and try again.
```

### Solution

Ping the standby node using tnsping. If you are unable to ping the standby node, stop and restart the listener for that node and retry the tnsping.

## A.1.20 Heartbeat Failure After Failover in Alert Logs

A warning appears in the alert logs of the database after a failover scenario.

### Problem

The following warning appears in the alert logs of the database after a failover scenario, where the new primary database fails to tnsping its remote database instance.

```
Errors in file c:\oracle\product\10.2.0\admin\orcl\udump\orcl1_rfs_1816.trc:
ORA-16009: remote archive log destination must be a STANDBY database
.
Fri Sep 08 09:11:13 2006
Errors in file c:\oracle\product\10.2.0\admin\orcl\bdump\orcl1_arc1_496.trc:
ORA-16009: remote archive log destination must be a STANDBY database
.
Fri Sep 08 09:11:13 2006
PING[ARC1]: Heartbeat failed to connect to standby 'orcl1_remote1'. Error is
16009.
Fri Sep 08 09:11:50 2006
Redo Shipping Client Connected as PUBLIC
-- Connected User is Valid
RFS[67]: Assigned to RFS process 628
RFS[67]: Database mount ID mismatch [0x4342404d:0x4341ffb0]
Fri Sep 08 09:11:50 2006
```

```
Errors in file c:\oracle\product\10.2.0\admin\orcl\udump\orcl1_rfs_628.trc:
ORA-16009: remote archive log destination must be a STANDBY database
.
Redo Shipping Client Connected as PUBLIC
-- Connected User is Valid
RFS[68]: Assigned to RFS process 2488
RFS[68]: Database mount ID mismatch [0x4342404d:0x4341ffb0]
Fri Sep 08 09:12:05 2006
Errors in file c:\oracle\product\10.2.0\admin\orcl\udump\orcl1_rfs_2488.trc:
ORA-16009: remote archive log destination must be a STANDBY database
.
Fri Sep 08 09:12:14 2006
Errors in file c:\oracle\product\10.2.0\admin\orcl\bdump\orcl1_arc1_496.trc:
ORA-16009: remote archive log destination must be a STANDBY database
```

### Solution

To avoid these error messages in the alert logs, null the log_archive_dest_2 parameter using the following commands:

```
alter system set log_archive_dest_2='SERVICE=null LGWR ASYNC REOPEN=60';
alter system set log_archive_dest_state_2='defer';
```

## A.1.21 Create Standby Database Fails If Database Uses OMF Storage or ASM Storage

The `create standby database` command fails with ASG_ORACLE-300: ORA-01276 errors with some database storage options.

### Problem

The `create standby database` command fails with ASG_ORACLE-300: ORA-01276 errors if the database storage option uses OMF (Oracle Managed Files) or ASM (Automatic Storage Management).

### Solution

Create a new database instance using DBCA on the primary site with alternate storage options before running the `create standby database` command.

## A.1.22 Database Already Exists Errors During Create Standby

Error messages appear when attempting to overwrite an existing database.

### Problem

If you run a `create standby database` command and the database already exists at the target host, you get the following error messages:

```
Checking whether standby instance already exists
proddnode1: -->ASG_DUF-4950: An error occurred on host "proddnode1" with IP
"a.b.c.d" and port "7891"
standbynode1: -->ASG_DUF-4950: An error occurred on host "standbynode1" with IP
"e.f.g.h" and port "7891"
standbynode1: -->ASG_DGA-12500: Standby database instance "orcl" already exists
on host "standbynode1".
standbynode1: -->ASG_DGA-13001: Error during Create Physical Standby:
Prepare-check standby.
standbynode1: -->ASG_DUF-3027: Error while executing Creating physical standby
database - prepare phase at step - check standby step.
```

### Solution

The `create standby database` command uses a database at a primary site to create a standby database at a standby site. It assumes that *only* the Oracle database software has been installed on the standby site peer host in the same Oracle database home directory as the primary database at the primary site; the command fails with the error messages in the preceding example if an actual database exists in the standby site peer host's Oracle database home.

Perform the following steps, then run the `create standby database` command again to create the standby database at the standby site peer host:

1. If the database at the standby site peer host is running, shut it down using the SQL*Plus `shutdown immediate` or `shutdown abort` command.
2. Remove references to the database from the standby site peer host as follows:

   **On Windows:**

   Run the oradim command to delete the Oracle SID:

   ```
   > oradim -delete -sid <database-sid>
   ```

   **On UNIX:**

   Delete the database entry from the oratab file.

   For non-Real Application Clusters databases, the entry has this format:

   ```
   DBSID:oracle_home
   ```

   For Real Applications Clusters databases, the entry has this format:

   ```
   DBuniqueName:oracle_home
   ```

3. Delete any initialization files in the Oracle database home at the standby site peer host.

## A.1.23 Oracle Application Server Guard Add Instance Command Fails When Attempting to Add an Oracle RAC Database to the Topology

When using the add instance command to add an Oracle RAC database instance to the topology, the method of referring to the database is different on Linux systems than on Windows systems.

On Linux systems, the oratab entry is used for discovery of the home. For non-RAC installations the database SID is used in the oratab entry and for RAC installations the database unique name is used in the oratab entry.

On Windows systems, the system registry is used for discovery of the home, and the database SID located in the registry is used. Consequently, on Windows systems when adding an instance to the topology that is an Oracle RAC database, you must use the database SID instead of the database name when referring to the Oracle RAC database instance.

There must be an oratab entry (on Linux) or registry entry (on Windows) with the SID of the primary database instance that ASG attaches to with the asgctl `add instance` command.

See Section A.1.22 for examples of database SIDs for Windows and UNIX databases and of a database unique name for UNIX systems.

### Problem

The Oracle RAC database install on Windows does not store the Oracle RAC database name or the global database name anywhere in the registry. Therefore, the workaround to this problem for Windows systems is as follows. When using the asgctl `add instance` command, always use the database SID of a RAC database on Windows and proceed with rest of the Oracle Disaster Recovery cycle of operations, such as `create standby database`, `instantiate topology`, `sync topology`, and `switchover topology`. For example:

```
asgctl> add instance <database SID of RAC database on Windows> on <virtualhost>

asgctl> add instance orcl1 on asinfra.us.oracle.com
```

### Solution

Use the database SID of an Oracle RAC database on Windows in asgctl commands.

## A.1.24 A Create Standby Database Operation Fails with an ASG_DGA-12500 Error Message on Windows

An error occurs when Oracle Application Server Guard issues a `create standby database` command on Windows and the target standby database environment has not been cleaned up.

### Problem

When Oracle Application Server Guard issues a create standby database command on Windows if the target standby database environment has not been cleaned up, the following error occurs:

```
ASG_DGA-12500: Standby database instance "db25" already exists on host <hostame>
```

The target environment may not be clean because a previous attempted setup of the standby failed for some system reason or because of the operations being attempted to 'reestablish' an existing standby database.

### Solution

Clean up the environment using the following command.

```
oradim -delete -sid db25
```

After cleaning up the environment, the asgctl `create standby database` command can be reissued.

## A.1.25 Use Fully Qualified Instance Names to Ensure Uniqueness

When you add an instance to an OracleAS Disaster Recovery topology, the instance name must be unique within the topology. This condition is validated by Oracle Application Server Guard when the instance is being added. The instance name can be fully qualified with the host on which it is deployed to ensure uniqueness.

### Problem

If the instance name of each Oracle Application Server tier is not unique across all the homes on all the nodes in the primary site, when you execute an `add instance` command for the second instance with the same instance name as an already added instance, you get the following error:

```
ASGCTL> add instance ohs on mt1
ASGCTL> add instance ohs on mt2
host2: -->ASG_IAS-15782: Error: Instance "ohs" already exists in the
topology
ASGCTL>
```

#### Solution

Use fully qualified instance names to ensure that the instance names are unique within the topology, for example:

```
ASGCTL> add instance ohs.mt1.mycompany.com on mt1
ASGCTL> add instance ohs.mt2.mycompany.com on mt2
ASGCTL>
```

### A.1.26 Misleading Message on JSSO Page

A misleading message appears under the Instances and Properties tab of the Java SSO Configuration page.

#### Problem

When you use Application Server Control to configure Java Single Sign-On (Java SSO), the following message appears at the top of the Java SSO Configuration page if no Java SSO applications are running in the cluster:

```
There are no active Java SSO applications in the cluster. At least one Java SSO
application (javasso) must be running before you can configure Java SSO.
```

However, the following additional and misleading message appears under the Instances and Property tab in the Java SSO Configuration page:

```
Java SSO is configured for this cluster.
```

#### Solution

When an error message appears at the top of the Java SSO Configuration page, ignore the message that appears under the Instances and Properties tab. In fact, Java SSO cannot be configured until at least one instance of the Java SSO application is running in the cluster.

For more information, see "OC4J Java Single Sign-On" in the *Oracle Application Server Containers for J2EE Security Guide*

### A.1.27 Instantiate Topology Fails if TNS Alias Includes Domain

Instantiate topology fails with error messages if the TNS alias entries for the standby database include the domain.

#### Problem

The following errors are returned when the TNS alias entries for the standby database include domain:

```
ORCL.ORACLE.COM =
 (DESCRIPTION =
 (ADDRESS = (PROTOCOL = TCP)(HOST = standbynode.oracle.com)(PORT = 1521))
 (CONNECT_DATA =
 (SERVER = DEDICATED)
 (SERVICE_NAME = ORCL.ORACLE.COM)
 )
 )
.
ASG_ORACLE-300: ORA-12514: TNS:listener does not currently know of service
@ requested in connect descriptor
```

#### Solution

Add the correct domain to the NAMES.DEFAULT_DOMAIN parameter in sqlnet.ora on the standby database before running the instantiate command.

### A.1.28 ORA-32001 Errors during Create Standby Database

In Windows operating systems, errors are returned after executing the `create standby database` command.

#### Problem

The `create standby database` command creates the SPFILE under ORACLE_HOME/dbs directory on the standby instead of ORACLE_HOME/database. As a result, the whenever the database is started up on the standby site, it fails to use the SPFILE under ORACLE_HOME/dbs and uses pfile instead. When the `create standby` command is executed again from standby site (for role reversal) it fails because the database does not use spfile.

```
stanbynode1: -->ASG_DUF-4950: An error occurred on host "stada26" with IP
"140.87.5.
@ 102" and port "7892"
standbynode1: -->ASG_ORACLE-300: ORA-32001: write to SPFILE requested but no
SPFILE specified at startup
standbynode1: -->ASG_DUF-3700: Failed in SQL*Plus executing SQL statement:
alter sys
tem set db_file_name_convert=
'C:\WORK\ORADATA\ASDB01','C:\WORK\ORADATA\ASDB01'
SCOPE=SPFILE /* ASG_DGA */;.
standbynode1: -->ASG_DGA-13010: Error during Create Physical Standby:
Finish-configure primary.
standbynode1: -->ASG_DUF-3027: Error while executing Creating physical standby
database - finish phase at step - finish step.
ASG_ORACLE-300: ORA-12514: TNS:listener does not currently know of service
requested in connect descriptor
```

#### Solution

On Windows only, after executing the `create standby database` command, copy the `SPFILE from ORACLE_HOME/dbs` to `ORACLE_HOME/database` on the standby database site.

### A.1.29 ORA-09925 Errors when Bringing Up RAC Database Manually after Switchover

ORA-09925 errors appear when bringing up a RAC database manually after a switchover operation.

#### Problem

The following errors appear after a ASG switchover operation, when bringing up some of the RAC database instances manually.

```
SQL> startup;
ORA-09925: Unable to create audit trail file
Linux Error: 2: No such file or directory
Additional information: 9925
```

### Solution

Make sure the directory pointed at by the `audit_file_dest` parameter in your init file exists.

For example:

```
mkdir <ORACLE_HOME>/admin/<dbname>/admin
```

When the asgctl `create standby database` command is used to create a RAC database at the standby site, the `audit_file_dest` init.ora parameter will be defined at the standby site database if it was defined for the production site database.

## A.1.30 Recommended Method of Patching an Oracle Application Server Disaster Recovery Site

This section describes how to apply an Oracle Application Server patch set to upgrade the Oracle homes that participate in an Oracle Application Server Disaster Recovery site.

### Problem

You are unsure how to apply an Oracle Application Server patch set to upgrade the Oracle homes in your Oracle Application Server Disaster Recovery site.

### Solution

The list in this section describes the steps for applying an Oracle Application Server patch set to upgrade the Oracle homes that participate in an Oracle Application Server Disaster Recovery site.

**Note:** It is also possible to upgrade or update the version of Oracle Application Server Guard (OracleAS Guard) that is installed in the existing Oracle home for an Application Server instance. This OracleAS Guard-only upgrade only upgrades the OracleAS Guard (ASG) utility; it does not affect the runtime operation of the other components in the Application Server home. See Section 1.1.2, "Using Oracle Application Server Guard in an OracleAS Disaster Recovery Topology" for more information about how to upgrade OracleAS Guard in an Oracle Application Server home.

Use the following procedure to upgrade Oracle Application Server patch versions:

1. Perform a backup of the production site to ensure that the starting state is secured.
2. Perform an ASG `sync topology` operation using a mandatory policy to synchronize all the instances in the topology. This ensures that prior to patching the configuration is updated at the standby site.
3. Perform an ASG `failover` operation, but do not perform a DNS switchover for the topology. This breaks the production/standby relationship of the topology and forms two sites. Starting with this step, the backup operation is the last resort recovery of the site prior to the upgrade procedure.

4. Perform the upgrade at the former standby site. The upgrade of the former standby site is a test that the upgrade will be smooth and successful. Because a DNS switchover was not performed in the previous step, access to the site is still maintained at the former production site. Your recovery point is effectively the point of the backup.
5. If problems occur in the previous step, you will remedy them when upgrading the former production site.
6. When the standby site upgrade is complete, upgrade the former production site.
7. Perform an ASG `discover topology` operation at the former production site.
8. Perform an ASG `instantiate topology` operation at the production site to establish the relationship between the production and standby sites, mirror the configuration, and synchronize the standby site with the production site.
9. The upgrade is now complete. Your Disaster Recovery topology is ready to resume processing.

## A.2 Troubleshooting Middle-Tier Components

This section describes common problems and solutions for middle-tier components in high availability configurations. It contains the following topics:

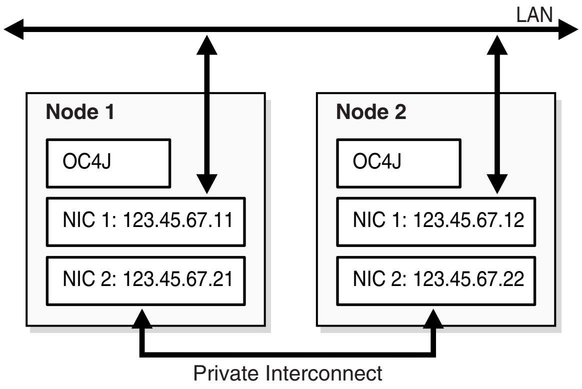
- Section A.2.1, "Using Multiple NICs with OracleAS Cluster (OC4J-EJB)"
- Section A.2.2, "Performance Is Slow When Using the "opmn:" URL Prefix"

### A.2.1 Using Multiple NICs with OracleAS Cluster (OC4J-EJB)

#### Problem

If you are running OracleAS Cluster (OC4J-EJB) on computers with two NICs (network interface cards) and you are using one NIC for connecting to the network and the second NIC for connecting to the other node in the cluster, multicast messages may not be sent or received correctly. This means that session information does not get replicated between the nodes in the cluster.

*Figure A–1 OracleAS Cluster (OC4J-EJB) Running on Computers with Two NICs*



### Solution

You must start up the OC4J instances by setting the `oc4j.multicast.bindInterface` parameter to the name or IP address of the other NIC on the node.

For example, using the values shown in Figure A–1, you would start up the OC4J instances with these parameters:

On node 1, configure the OC4J instance to start with up with this parameter:

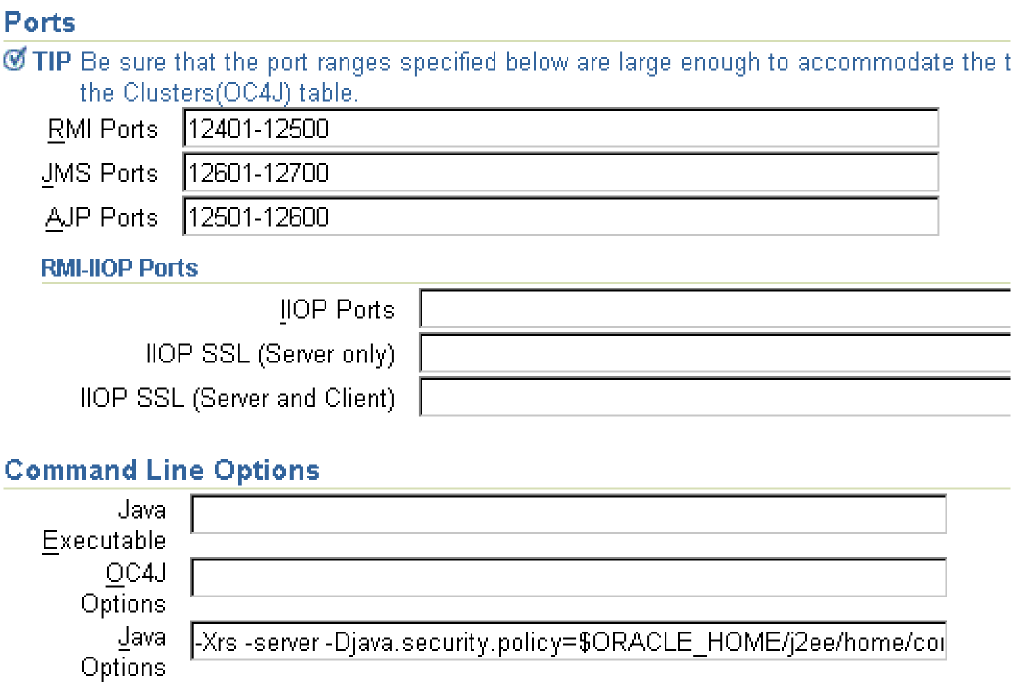
```
-Doc4j.multicast.bindInterface=123.45.67.21
```

On node 2, configure the OC4J instance to start with up with this parameter:

```
-Doc4j.multicast.bindInterface=123.45.67.22
```

You specify this parameter and its value in the "Java Options" field in the "Command Line Options" section in the Server Properties page in the Application Server Control Console (Figure A–2).

***Figure A–2 Server Properties Page in Application Server Control Console***



Ports

TIP Be sure that the port ranges specified below are large enough to accommodate the t the Clusters(OC4J) table.

RMI Ports: 12401-12500

JMS Ports: 12601-12700

AJP Ports: 12501-12600

RMI-IIOP Ports

IIOP Ports:

IIOP SSL (Server only):

IIOP SSL (Server and Client):

Command Line Options

Java Executable:

OC4J Options:

Java Options: -Xrs -server -Djava.security.policy=$ORACLE_HOME/j2ee/home/co

## A.2.2 Performance Is Slow When Using the "opmn:" URL Prefix

### Problem

If you have applications that use the "`opmn:`" prefix in their `Context.PROVIDER_URL` property, you may experience slow performance in the `InitialContext` method.

The following sample code sets the `PROVIDER_URL` to a URL with an `opmn:` prefix.

```
Hashtable env = new Hashtable();
env.put(Context.PROVIDER_URL, "opmn:ormi://hostname:port/cmpapp");
// ... set other properties ...
Context context = new InitialContext(env);
```

If the host specified in PROVIDER_URL is down, the application has to make a network connection to OPMN to locate another host. Going through the network to OPMN takes time.

### Solution

To avoid making another network connection to OPMN to get another host, set the oracle.j2ee.naming.cache.timeout property so that the values returned from OPMN the first time are cached, and the application can use the values in the cache.

The following sample code sets the oracle.j2ee.naming.cache.timeout property.

```
Hashtable env = new Hashtable();
env.put(Context.PROVIDER_URL, "opmn:ormi://hostname:port/cmpapp");

// set the cache value
env.put("oracle.j2ee.naming.cache.timeout", "30");

// ... set other properties ...

Context context = new InitialContext(env);
```

Table A–1 shows valid values for the oracle.j2ee.naming.cache.timeout property:

*Table A–1 Values for the oracle.j2ee.naming.cache.timeout Property*

| Value | Meaning |
|---|---|
| -1 | No caching. |
| 0 | Cache only once, without any refreshing. |
| Greater than 0 | Number of seconds after which the cache can be refreshed. Note that this is **not automatic**; the refresh occurs only when you invoke "new InitialContext()" again.<br>If the property is not set, the default value is 60. |

With the property set, you will still see some delay on the first "new InitialContext()" call, but subsequent calls should be faster because they are retrieving data from the cache instead of making a network connection to OPMN.

Note that for optimal performance, you should also set Dedicated.Connection to either YES or DEFAULT, and set Dedicated.RMIcontext to FALSE.

## A.3 Need More Help?

In case the information in the previous section is not sufficient, you can find more solutions on Oracle *MetaLink*, https://metalink.oracle.com. If you do not find a solution for your problem, log a service request.

**See Also:**

- *Oracle Application Server Release Notes*, available on the Oracle Technology Network: http://www.oracle.com/technology/documentation/index.html

# B

# Oracle Application Server Guard Error Messages

The following sections describe the Oracle Application Server Guard error messages. Though not shown, Oracle Application Server Guard error messages are preceded by an ASG prefix. Error messages are categorized into the following groups and subgroups:

- DGA Error Messages
  - LRO Error Messages
  - Undo Error Messages
  - Create Template Error Messages
  - Switchover Physical Standby Error Messages
- Duf Error Messages
  - Database Error Messages
  - Connection and Network Error Messages
  - SQL*Plus Error Messages
  - JDBC Error Messages
  - OPMN Error Messages
  - Net Services Error Messages
  - System Error Messages
  - Warning Error Messages
  - OracleAS Database Error Messages
  - OracleAS Topology Error Messages
  - OracleAS Backup and Restore Error Messages
  - Oracle Application Server Guard Synchronize Error Messages
  - Oracle Application Server Guard Instantiate Error Messages

## B.1 DGA Error Messages

The following are DGA error messages.

**Note:** The symbols {0}, {1}, and {2} are variables that will be replaced by the name of the object.

**12001, Error while creating a DGA template.**

**Cause:** An error occurred while creating a template file.

**Action:** See secondary error.

**12500, Standby database instance {0} already exists on host {1}.**

**Cause:** The standby database instance specified already exists on target host.

**Action:** Either select a new instance or remove the current instance.

### B.1.1 LRO Error Messages

The following are LRO error messages.

**13000, Error during Create Physical Standby: Prepare-init.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13001, Error during Create Physical Standby: Prepare-check standby.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13002, Error during Create Physical Standby: Prepare-primary processing.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13003, Error during Create Physical Standby: Prepare-standby processing.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13004, Error during Create Physical Standby: Prepare-sqlnet configuration.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13005, Error during Create Physical Standby: Copy-init.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13006, Error during Create Physical Standby: Copy-validate standby.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13007, Error during Create Physical Standby: Copy-file copy.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13008, Error during Create Physical Standby: Finish-init.**

**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13009, Error during Create Physical Standby: Finish-prepare primary.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13010, Error during Create Physical Standby: Finish-configure primary.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13011, Error during Create Physical Standby: Finish-configure standby.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

### B.1.2 Undo Error Messages

The following are undo error messages.

**13015, Error trying to Undo Create Physical Standby: Prepare.**
**Cause:** Error occurred during undo of the prepare task.

**Action:** See secondary error.

**13016, Error trying to Undo Create Physical Standby: Copy.**
**Cause:** Error occurred during undo of the copy task.

**Action:** See secondary error.

**13017, Error trying to Undo Create Physical Standby: Finish.**
**Cause:** Error occurred during undo of the finish task.

**Action:** See secondary error.

### B.1.3 Create Template Error Messages

The following are create template error messages.

**13020, Error during Create Template: init.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13021, Error during Create Template: primary processing.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13022, Error during Create Template: standby processing.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

**13023, Error during Create Template: finish.**
**Cause:** Error occurred during specified step.

**Action:** See secondary error.

### B.1.4 Switchover Physical Standby Error Messages

The following are switchover physical standby error messages.

**13051, Error performing a physical standby switchover.**

**Cause:** Error occurred in performing a switchover.

**Action:** See secondary error.

**13052, The primary database is not in the proper state to perform a switchover.**

**Cause:** The switchover status of the primary database must be either "TO STANDBY" or "SESSIONS ACTIVE".

**Action:** Make sure the SWITCHOVER_STATUS of the V$DATABASE table is either "TO STANDBY" or "SESSIONS ACTIVE".

**13053, The standby database is not in the proper state to perform a switchover.**

**Cause:** The switchover status of the standby database must be either "TO PRIMARY" or "SWITCHOVER PENDING".

**Action:** Make sure the SWITCHOVER_STATUS of the V$DATABASE table is either "TO PRIMARY" or "SWITCHOVER PENDING".

**13504, Error switching the database role from primary to standby.**

**Cause:** Failed to switchover database role from primary to standby.

**Action:** See secondary error.

**13505, Error switching the database role from standby to primary.**

**Cause:** Failed to switchover database role from standby to primary.

**Action:** See secondary error.

**13061, Error failing over physical standby database.**

**Cause:** Error occurred in performing a failover of a standby database.

**Action:** See secondary error.

## B.2 Duf Error Messages

The following are Duf error messages.

**3000, Server error {0}.**

**Cause:** Invalid argument was supplied.

**Action:** Pass in a valid argument.

**3001, Invalid argument {0}.**

**Cause:** Invalid argument was supplied.

**Action:** Pass in a valid argument.

**3002, Invalid log path {0}.**

**Cause:** Invalid log path specification.

**Action:** Specify a valid log path.

**3003, Invalid command line value {0} specified.**

**Cause:** Invalid command line specification.

**Action:** Correct the command line option and retry.

**3004, Invalid command action {0} specified.**

**Cause:** Invalid command action specification.

**Action:** Correct the command line action and retry.

**3005, Invalid command argument {0} specified, commands must begin with a hyphen.**

**Cause:** Command argument did not start with a hyphen.

**Action:** Enter a correct command line argument.

**3006, Command line argument {0} missing a required value.**

**Cause:** Command argument missing a required value.

**Action:** Enter a correct command line argument value.

**3007, Command line argument {0} given an incorrect value {1}.**

**Cause:** Command argument value is incorrect.

**Action:** Enter a correct command line argument value.

**3008, Command line argument {0} is required but missing.**

**Cause:** Command argument value is missing.

**Action:** Enter a correct command line argument value.

**3009, Invalid session ID.**

**Cause:** The client passed an invalid session ID.

**Action:** Enter a correct command line argument value.

**3010, Duplicate session ID.**

**Cause:** The session ID is already in use.

**Action:** Enter a correct command line argument value.

**3011, Unsatisfied link error for {0} in library DufNatives.**

**Cause:** An attempt to make a call using the DufNatives library failed.

**Action:** Make sure the DufNatives library is correctly installed.

**3012, Checksum error in password.**

**Cause:** The login password has a checksum error.

**Action:** Try to reconnect.

**3013, Operation failed.**

**Cause:** The specified operation failed.

**Action:** See secondary error.

**3014, Invalid command line specified.**

**Cause:** Invalid command line specification.

**Action:** Correct command line option and retry.

**3015, Error getting local host name.**

**Cause:** Error trying to get local host name.

**Action:** See secondary error.

**3016, No encrypt key.**

**Cause:** No encrypt key supplied. Encryption requires an encrypt key.

**Action:** This is an internal programming error.

**3017, Error encrypting data.**

**Cause:** Failed to encrypt the given data.

**Action:** See secondary error.

**3018, Error missing plan for specified request {0}, cannot process.**

**Cause:** Could not find plan for specified request.

**Action:** Either specify a valid request or supply valid plan.

**3019, Server does not recognize application ID.**

**Cause:** Client specified an application ID that the server does not support.

**Action:** Contact Oracle support.

**3020, Failed to authenticate user {0}. Please enter the correct user name and password.**

**Cause:** Client supplied incorrect operating system user name or password or both.

**Action:** Make sure the correct operating system user name and password are used.

**3021, No user name and/or password are supplied for authentication.**

**Cause:** Client did not supply a user name or password or both through the "connect duf" command.

**Action:** Make sure user issue a "connect duf" command before any other commands.

**3022, Failed to authorize user {0}. User must have administrator privilege on the server system.**

**Cause:** The user name provided by the client to connect to DUF server must have administrator privilege on the server system. This error is applicable on Windows system only.

**Action:** Make sure the user account belongs to the administrator group on the server system.

**3023, Error: There is no connection to a DUF server.**

**Cause:** You must connect to DUF server before issues other commands.

**Action:** Connect to the DUF server.

**3024, Failed to authorize user {0}. The owner account of the Oracle home must be used.**

**Cause:** The user name provided by the client to connect to DUF server must be the same user from which the Oracle home is installed. This error is applicable on UNIX system only.

**Action:** Make sure the user account is the same as that of the Oracle home.

**3025, The operation has been canceled.**

**Cause:** The operation has been canceled by either the user or the DUF internal software.

**Action:** None.

**3026, The {0} task must complete successfully before running the {1} task.**

**Cause:** An attempt was made to run the specified task before the required previous task has successfully completed.

**Action:** Rerun the required previous task.

**3027, Error while executing {0} at step - {1}.**

**Cause:** Error during specified step of specified operation.

**Action:** Check secondary error.

**3028, Failed to start DUF server on host {0}.**

**Cause:** Error during specified step of specified operation.

**Action:** Check DUF log file for more information.

**3029, Failed to start {0} server with exception.**

**Cause:** Error trying to start the server.

**Action:** See secondary error.

**3030, Error, cannot resolve host name {0}.**

**Cause:** Error trying to resolve specified host name.

**Action:** Check that the host name is correctly specified.

**3031, Error, Invalid user name {0}. Only {1} account can connect to a ASG server.**

**Cause:** Only ias_admin can connect to a ASG server.

**Action:** Please use ias_admin to connect to a ASG server.

**3032, Failed to start {0} server on host {1}. Start server on specified host and reconnect.**

**Cause:** Error trying to start the server on the specified host while trying to connect.

**Action:** Start the server manually and retry the connect.

**3033, Error, the server is shutting down.**

**Cause:** Error communicating with the server.

**Action:** Retry the operation.

**3034, Invalid command line specified: - {0}.**

**Cause:** Invalid command line specification.

**Action:** Correct the command line option and retry.

**3035, Failed to kill the OracleAS Guard (ASG) server process {0} with error {1}.**

**Cause:** OracleAS Guard client is unable to kill the OracleAS Guard (ASG) server process.

**Action:** Use "kill -9 <pid>" command to kill the process from the command line prompt.

**3100, Error reading file {0}.**

**Cause:** Error trying to read from file.

**Action:** See secondary error.

**3101, Error writing file {0}.**

**Cause:** Error trying to write file.

**Action:** See secondary error.

**3102, Error creating file {0}.**

**Cause:** Error trying to create specified file.

**Action:** See secondary error.

**3103, Error deleting file {0}.**

**Cause:** Error trying to delete a file.

**Action:** See secondary error.

**3104, Error opening  file {0}.**

**Cause:** Error trying to open file.

**Action:** See secondary error.

**3105, File {0} not found.**

**Cause:** Error trying to open file.

**Action:** See secondary error.

**3106, No read access to file {0}.**

**Cause:** Error trying to open file.

**Action:** See secondary error.

**3107, No write access to file {0}.**

**Cause:** Error trying to open file.

**Action:** See secondary error.

**3108, File specification {0} must be absolute.**

**Cause:** Error trying to open file.

**Action:** See secondary error.

**3109, Error closing file {0}.**

**Cause:** Error trying to close the file.

**Action:** See secondary error.

**3110, Error creating dir {0}.**

**Cause:** Error trying to create specified directory.

**Action:** See secondary error.

**3111, Error deleting dir {0}.**

**Cause:** Error trying to delete specified directory.

**Action:** See secondary error.

**3112, Error expanding file wildcard specification {0}.**

**Cause:** Error trying to process file wildcard specification.

**Action:** See secondary error.

**3120, Error opening configuration file {0}.**

**Cause:** Error trying to open configuration file.

**Action:** Make sure configuration file exists or specified correctly.

**3121, Error creating zip file {0}.**

**Cause:** Error trying to create a zip file.

**Action:** See secondary error.

**3122, There are no files to be zipped.**

**Cause:** The directory to be zipped has no files in it.

**Action:** Make sure the directory to be zipped has files in it.

**3123, Error adding files in directory {0} to zip file.**

**Cause:** Error adding files in the given directory to zip file.

**Action:** See secondary error.

**3124, No zip file is specified.**

**Cause:** No zip file is specified.

**Action:** Internal error.

**3125, Error extracting files from zip file {0}.**

**Cause:** Error extracting files from a zip file.

**Action:** See secondary error.

**3400, Error processing XML document.**

**Cause:** Error processing XML document.

**Action:** See secondary error.

**3401, Error processing XML node.**

**Cause:** Error processing XML node.

**Action:** See secondary error.

**3402, Error parsing XML request message.**

**Cause:** There was an error parsing the XML request message.

**Action:** Contact Oracle support.

**3403, Error parsing XML response message.**

**Cause:** There was an error parsing the XML response message.

**Action:** Contact Oracle support.

**3404, Error parsing XML body string.**

**Cause:** There was an error parsing the XML body string.

**Action:** Contact Oracle support.

**3405, Error writing the body to an XML DOM.**

**Cause:** There was an error writing the XML body string.

**Action:** Contact Oracle support.

**3406, Error reading the body from an XML DOM.**

**Cause:** There was an error reading the XML body string.

**Action:** Contact Oracle support.

**3407, Error writing a work item to an XML DOM.**

**Cause:** There was an error writing the XML body string.

**Action:** Contact Oracle support.

**3408, Error reading a work item from an XML DOM.**

**Cause:** There was an error reading the XML body string.

**Action:** Contact Oracle support.

**3409, Error parsing an XML string.**

**Cause:** There was an error parsing the XML string.

**Action:** Contact Oracle support.

**3410, Error converting XML DOM to string.**

**Cause:** There was an error converting the DOM tree to a XML string.

**Action:** Contact Oracle support.

**3411, Error reading XML DOM tree.**

**Cause:** There was an error reading the XML DOM tree.

**Action:** Contact Oracle support.

## B.2.1 Database Error Messages

The following are database error messages.

**3501, Failed to initialize DufDb class.**

**Cause:** There was an error creating the DufDb class.

**Action:** See secondary error.

**3502, Failed to connect to database {0}.**

**Cause:** There was an error connecting to the database.

**Action:** See secondary error.

**3503, Failed to verify database {0}.**

**Cause:** There was an error verifying the database.

**Action:** See secondary error.

**3504, Failed to start database {0}.**

**Cause:** There was an error starting the database.

**Action:** See secondary error.

**3505, Failed to create pfile to include spfile.**

**Cause:** There was an error creating the given pfile.

**Action:** See secondary error.

**3506, Failed to turn on archivelog mode for the database.**

**Cause:** There was an error turning on archivelog mode.

**Action:** See secondary error.

**3507, Failed to create the standby database control file.**

**Cause:** There was an error creating the standby database control file.

**Action:** See secondary error.

**3508, Failed to create the pfile.**

**Cause:** There was an error creating the database init parameter file.

**Action:** See secondary error.

**3509, Failed to create the spfile.**

**Cause:** There was an error creating the database spfile.

**Action:** See secondary error.

**3510, Output reader thread for {0} terminated.**

**Cause:** The output reader thread is terminated.

**Action:** Contact Oracle support.

**3511, Error creating local worker on node {0}.**

**Cause:** This is an internal error.

**Action:** Contact Oracle support.

**3512, Error creating remote worker on node {0}.**

**Cause:** There is a problem communicating with the remote server.

**Action:** Make sure that the remote server is accessible.

**3513, Database is not started {0}.**

**Cause:** The specified database has not been started.

**Action:** Start the specified database.

**3514, Failed to stop database {0}.**

**Cause:** There was an error stopping the database.

**Action:** See secondary error.

**3515, Failed querying database to determine current archivelog mode.**

**Cause:** There was an error querying the database to determine current archive mode.

**Action:** See secondary error.

**3516, Failed to query redo log information for database.**

**Cause:** There was an error querying the database redo log information.

**Action:** See secondary error.

**3517, Failed to drop standby redo log for database.**

**Cause:** There was an error dropping the standby redo log.

**Action:** See secondary error.

**3518, Failed to start managed recovery for standby database.**

**Cause:** There was an error starting managed recovery for the standby database.

**Action:** See secondary error.

**3519, Failed to cancel managed recovery for standby database.**

**Cause:** There was an error canceling managed recovery for the standby database.

**Action:** See secondary error.

**3520, Failed to determine the existence of database instance.**

**Cause:** There was an error determining the existence of the given database instance.

**Action:** See secondary error.

**3521, Invalid database instance {0} specified in the template file; DUF found instance {1}.**

**Cause:** The standby database instance specified in the template file is different from the one DUF found on the system.

**Action:** Please either rerun the prepare and copy phases with the new standby instance or specify the correct standby database instance found on the system.

**3522, The pfile {0} needed to generate an spfile is missing.**

**Cause:** A pfile needed to create the spfile used by the standby database is missing.

**Action:** Please either rerun the prepare and copy phases to generate the pfile or manually create one with the correct values.

**3523, The standby database cannot have the same service name as the primary database.**

**Cause:** The standby service name is the same as the primary.

**Action:** Change the standby service name.

**3524, Error: The primary database is not set.**

**Cause:** The primary database is not defined.

**Action:** Set the primary database first.

**3525, Error: The standby database is not set.**

**Cause:** The standby database is not defined.

**Action:** Set the standby database first.

**3526, Set the primary database before setting the standby database.**

**Cause:** The standby service name is the same as the primary on the same host.

**Action:** Change the standby service name.

**3527, The database tablespace map is NULL.**

**Cause:** This is an internal error.

**Action:** Contact Oracle support.

**3528, Error initializing init parameter file {0}.**

**Cause:** An error occurred trying to initialize the parameter file.

**Action:** See secondary error.

**3529, Error writing init parameter file {0}.**

**Cause:** An error occurred trying to write the parameter file.

**Action:** See secondary error.

**3530, Error in setting the protection mode for database {0}.**

**Cause:** An error occurred trying to set the protection mode.

**Action:** See secondary error.

**3531, Error opening database in read only mode for database {0}.**

**Cause:** An error occurred trying to open the database in read only mode.

**Action:** See secondary error.

**3532, Failed to get init parameter value from {0}.**

**Cause:** Error trying to get the parameter value from the init parameter file.

**Action:** See secondary error.

**3533, No user name and/or password is specified for database {0}.**

**Cause:** Error trying to get the parameter value from the init parameter file.

**Action:** User must specify the user name and password to be used to connect to the database using "set primary database" or "set standby database" command.

**3534, The standby database cannnot have the same host as the primary database.**

**Cause:** The standby host is the same as the primary.

**Action:** Change the standby or primary database host name.

**3535, Failed to create standby redo log.**

**Cause:** An error occurred trying to create a standby redo log.

**Action:** See secondary error.

**3536, Failed to get a list of standby database(s) from log archive destination.**

**Cause:** An error occurred trying to get a list of standby databases from the log archive destination parameters.

**Action:** See secondary error.

**3537, Failed to add standby database as a log archive destination.**

**Cause:** An error occurred trying to add a standby database as a log archive destination.

**Action:** See secondary error.

**3538, Failed to remove standby database as a log archive destination.**

**Cause:** An error occurred trying to remove a standby database as a log archive destination.

**Action:** See secondary error.

**3539, Error: The new primary database is not set.**

**Cause:** The new primary database is not defined.

**Action:** Set the new primary database first.

**3540, Error processing template file {0}.**

**Cause:** Error trying to process template file.

**Action:** Correct protection and retry operation.

**3541, Invalid database protection specified in template file {0}.**

**Cause:** Error trying to process protection value in template file.

**Action:** Correct protection and retry operation.

**3542, Failed to query database role.**

**Cause:** Error trying to query the database role.

**Action:** See secondary error.

**3543, Error processing command, must be connected to a OracleAS Guard server in the primary topology.**

**Cause:** User is connected to server on a topology that is not the primary topology.

**Action:** Connect to primary topology node.

**3544, Error processing command, must be connected to a OracleAS Guard server in the standby topology.**

**Cause:** User is connected to server on a topology that is not the standby topology.

**Action:** Connect to primary topology node.

**3545, Error trying to remove old passwd file %1 while creating new db.**

**Cause:** Could not delete the old password file as part of a delete database operation. This is a problem when trying to create a new database.

**Action:** Delete the stale password file.

**3546, Error, database SID was expected to have value but it is empty.**

**Cause:** The database SID was suppose to have a value but it is empty.

**Action:** This is an internal error.

**3547, Error storing DB Credentials in the clipboard of the server.**

**Cause:** Failed to store DB credentials in the clipboard on the specified server.

**Action:** Internal error.

**3548, Error storing DB info in the clipboard of the server.**

**Cause:** Failed to store DB information in the clipboard on the specified server.

**Action:** Internal error.

**3549, Error cleaning up the database on the standby host.**

**Cause:** Failed to clean up the database on the standby host.

**Action:** See secondary error.

**3550, Failed to find a valid Oracle home.**

**Cause:** A valid Oracle home was not found for this operation.

**Action:** Create a valid Oracle home.

**3551, Oracle Data Guard Home must have the same owner as the database server home.**

**Cause:** The Oracle Data Guard Home is owned by a different user than the database server home.

**Action:** Reinstall Oracle Data Guard user from the owner of Oracle database server.

**3552, Specified Oracle home {0} could not be found.**

**Cause:** The specified Oracle home could not be found.

**Action:** Please specify a valid Oracle home.

**3553, An error occurred getting the list of Oracle Homes on the system.**

**Cause:** The list of Oracle homes could not be read.

**Action:** Make sure the Oracle inventory is valid.

**3554, The Oracle home that contains SID {0} cannot be found.**

**Cause:** The Oracle home that contains a specific SID cannot be found.

**Action:** Make sure the Oracle home inventory is valid.

**3555, Error accessing the Oracle home inventory.  Make sure the inventory file exists.**

**Cause:** The Oracle home inventory cannot be accessed.

**Action:** Make sure the Oracle home inventory exists

**3556, Error: Unable to find the Oracle home within path {0}.**

**Cause:** The Oracle home within the given path cannot be found.

**Action:** Make sure the Oracle home inventory exists.

## B.2.2 Connection and Network Error Messages

The following are connection and network error messages.

**3600, Error connecting to server: Unknown node {0}.**

**Cause:** The server host is unknown to the client.

**Action:** Contact Oracle support.

**3601, Error connecting to server node {0}.**

**Cause:** The client cannot connect to the server.

**Action:** Contact Oracle support.

**3602, File Copy protocol error.**

**Cause:** There was an internal protocol error while copying files.

**Action:** Contact Oracle support

**3603, Error sending data across network.**

**Cause:** There was a network error.

**Action:** Retry operation.

**3604, Error receiving data across network.**

**Cause:** There was a network error.

**Action:** Retry operation.

**3605, The file copy operation has been terminated.**

**Cause:** The copy aborted due to an error.

**Action:** Retry operation.

**3606, Error connecting to file copy server {0} on port {0}.**

**Cause:** The copy server is not running.

**Action:** Contact Oracle support.

**3607, Error opening file copy server socket on {0} with port {0}.**

**Cause:** The copy aborted due to an error.

**Action:** Retry operation.

**3608, Error connecting to clipboard.**

**Cause:** There is no connection to the clipboard server.

**Action:** Retry operation.

**3609, Error while copying {0} to {1}.**

**Cause:** Error occurred during a file copy.

**Action:** See secondary error.

**3610, Error starting online backup.**

**Cause:** Error occurred while putting tablespace in online backup mode.

**Action:** See secondary error.

**3611, Error ending online backup.**

**Cause:** Error occurred while restore tablespace from online backup mode.

**Action:** See secondary error.

**3612, Error listening on server port {0}.**

**Cause:** Error occurred while listening on port.

**Action:** Check if server is already running.

**3613, Network Buffer Overflow Detected.**

**Cause:** The network protocol detected a buffer overflow due to a bug or attack.

**Action:** Call Oracle Support.

### B.2.3 SQL*Plus Error Messages

The following are SQL*Plus error messages.

**3700, Failed in SQL*Plus executing SQL statement: {0}.**

**Cause:** Failed to execute the specified SQL statement.

**Action:** See secondary error.

**3701, Failed starting SQL*Plus : {0}.**

**Cause:** Failed to execute the specified SQL statement.

**Action:** See secondary error.

### B.2.4 JDBC Error Messages

The following are JDBC error messages.

**3751, Failed to register Oracle JDBC driver: oracle.jdbc.OracleDriver.**

**Cause:** Failed to register the Oracle JDBC driver.

**Action:** Make sure that Oracle JDBC driver is installed on the local system.

**3752, There is no JDBC connection to the database.**

**Cause:** There is no connection to the database server.

**Action:** Connect to a database server first, then try the operation again.

**3753, Failed to connect to the database.**

**Cause:** Unable to connect to the database server.

**Action:** See secondary error.

**3754, Failed to disconnect from the database.**

**Cause:** Unable to disconnect from the database server.

**Action:** See secondary error.

**3755, Failed to execute the SQL statement.**

**Cause:** Failed to execute the SQL statement.

**Action:** See secondary error.

**3756, Failed to run the SQL query.**

**Cause:** Failed to run the SQL query statement.

**Action:** See secondary error.

**3757, Failed to close the Oracle result set or the Statement object.**

**Cause:** Failed to close the Oracle result set or the Statement object.

**Action:** See secondary error.

**3758, This method cannot be used to verify the physical standby database.**
**Cause:** This is a programming error.

**Action:** Contact Oracle support.

**3759, Verify DB query returned no data.**
**Cause:** Verify database query returned no data.

**Action:** See secondary error.

**3760, Failed to query the archive log destination information.**
**Cause:** Failed to query the archive log destination information.

**Action:** See secondary error.

**3761, Failed to query the redo log information.**
**Cause:** Failed to query the redo log information.

**Action:** See secondary error.

**3762, Failed to process the results from SQL statement.**
**Cause:** Failed to process the results from the SQL statement.

**Action:** See secondary error.

**3763, Failed to query the data files of the database.**
**Cause:** Failed to query the data files from the database.

**Action:** See secondary error.

**3764, Failed to query the log files used by the database.**
**Cause:** Failed to query the log files used by the database.

**Action:** See secondary error.

**3765, Failed to query table space information.**
**Cause:** Failed to query tablespace information from the database.

**Action:** See secondary error.

### B.2.5 OPMN Error Messages

The following are OPMN error messages.

**3800, Failed trying to connect to OPMN Manager.**
**Cause:** Error trying to connect to OPMN manager.

**Action:** Make sure OPMN manager is started.

**3801, Failed trying to get topology information from OPMN Manager on {0}.**
**Cause:** Error trying to get topology information from OPMN manager.

**Action:** Make sure OPMN manager is started and working correctly.

**3802, Failed trying to stop OPMN Component {0}.**
**Cause:** Failed trying to stop the specified OPMN component.

**Action:** See secondary error.

**3803, Failed trying to start OPMN Component {0}.**
**Cause:** Failed trying to start the specified OPMN component.

**Action:** See secondary error.

**3804, Failed trying to remove topology entry from {0}.**

**Cause:** Failed trying to remove topology entry from opmn.xml.

**Action:** See secondary error.

**3900, Error creating Oracle database service because the service has already been marked for deletion. Please exit the Windows Service Control Manager on node {0}. Would you like to retry?**

**Cause:** The user has the SCM open causing a service operation to fail.

**Action:** User must exit SCM GUI.

### B.2.6 Net Services Error Messages

The following are Net Services error messages.

**4000, Failed trying to get Net Services default domain for {0}.**

**Cause:** Failed trying to get the Net Services default domain.

**Action:** See secondary error.

**4001, Error trying to add net service name entry for {0}.**

**Cause:** Failed trying to add the specified service name.

**Action:** See secondary error.

**4002, Error trying to get net service name entry for {0}.**

**Cause:** Failed trying to get the specified service name.

**Action:** See secondary error.

**4003, Error trying to get host name from net service entry for {0}.**

**Cause:** Failed trying to get the host name from the net service entry.

**Action:** See secondary error.

**4004, Error trying to get host name from net service description.**

**Cause:** Failed trying to get the host name from the net service description.

**Action:** See secondary error.

**4005, Error trying to get net service listener information.**

**Cause:** Failed trying to get the net service listener information.

**Action:** See secondary error.

**4006, Error trying to create a net service default listener.**

**Cause:** Failed trying to create a default listener.

**Action:** See secondary error.

**4007, Error trying to add SID entry {0} to net service listener {1}.**

**Cause:** Failed trying to add a SID entry to the listener.

**Action:** See secondary error.

**4008, Error generating a command a script for the net service listener command: {0}.**

**Cause:** Failed generating a command script for the listener.

**Action:** See secondary error.

**4009, Error running the command script for the net service listener command: {0}.**

**Cause:** Failed running the command script for the listener.

**Action:** See secondary error.

**4010, Error adding the net service TNS entry for {0}.**

**Cause:** Failed adding a TNS entry.

**Action:** See secondary error.

**4011, Error trying to delete SID entry {0} to the net service listener {1}.**

**Cause:** Failed trying to delete the SID entry to the listener.

**Action:** See secondary error.

**4012, Error trying to save the listener configuration.**

**Cause:** Listener information was modified and an attempt to save information failed.

**Action:** See secondary error.

**4013, Error deleting net service TNS entry for {0}.**

**Cause:** Failed deleting a TNS entry.

**Action:** See secondary error.

**4014, Error starting the TNS listener using the lsnrctl command.**

**Cause:** Failed to start the TNS listener.

**Action:** See secondary error.

**4030, The command \"{0}\" failed due to timeout.**

**Cause:** Command timed out.

**Action:** Increase timeout values in configuration file.

**4031, Error getting environment variables using the env command.**

**Cause:** The env command does not work.

**Action:** Make sure the `/bin` or `/usr/bin` directory contains the env executable.

**4040, Error executing the external program or script.**

**Cause:** The execution of the specified command failed.

**Action:** See secondary error.

**4041, Failed to get the value of {0} from the TNS name descriptor {1}.**

**Cause:** Failed to get the value for the given parameter from the TNS name descriptor.

**Action:** See secondary error.

**4042, Failed to update the value of {0} for the TNS name descriptor {1}.**

**Cause:** Failed to update the value of a given parameter in the TNS name descriptor.

**Action:** See secondary error.

**4043, Failed to compare the TNS descriptor entry {0} with entry {1}.**

**Cause:** Failed to compare the two TNS entries.

**Action:** See secondary error.

**4044, Failed to generate a remote TNS name descriptor for the service name.**

**Cause:** Failed to generate a remote TNS name descriptor for the given local database.

**Action:** See secondary error.

**4045, Failed to get the remote TNS service name for the service name.**

**Cause:** Failed to get the remote TNS service name for the given local database.

**Action:** See secondary error.

### B.2.7 LDAP or OID Error Messages

The following are LDAP (Lightweight Directory Access Protocol) or Oracle Internet Directory (OID) error messages.

**4101, Failed to connect to OID server on host {0}, port {1}.**

**Cause:** Failed to connect to the Oracle Internet Directory server on a given host and port.

**Action:** See secondary error.

**4102, Failed to connect to OID server via SSL on host {0}, port {1}.**

**Cause:** Failed to connect to the Oracle Internet Directory server through SSL on a given host and port.

**Action:** See secondary error.

**4103, User must specify host, port, user name, and password for the OID server.**

**Cause:** User did not specify all the preceding parameters.

**Action:** User must specify all the preceding parameters in order to access the Oracle Internet Directory server.

**4104, Failed to get the value of attribute {0} from OID server.**

**Cause:** Failed to get the value of the given attribute.

**Action:** See secondary error.

**4105, Failed to get the attributes for DN {0} from OID server**

**Cause:** Failed to get the attributes of the given DN.

**Action:** See secondary error.

**4106, Failed to get Oracle Application Server instances from OID server**

**Cause:** Failed to get Oracle Application Server instances from the Oracle Internet Directory server.

**Action:** See secondary error.

**4107, Failed to get infrastructure databases from OID server.**

**Cause:** Failed to get infrastructure databases from the Oracle Internet Directory server.

**Action:** See secondary error.

**4110, Cannot set current topology to file {0} because the file does not exist.**

**Cause:** The topology file does not exist.

**Action:** Specify a filename of a file that exists.

**4111, The current topology file \"{0}\" does not exist. Use the "set topology" command to specify a valid topology file.**

**Cause:** The topology file does not exist.

**Action:** Specify a filename of a file that exists in dsa.conf.

## B.2.8 System Error Messages

The following are system error messages.

**4900, An exception occurred on the server.**

**Cause:** A server exception occurred.

**Action:** See secondary error.

**4901, A null pointer exception occurred on the server.**

**Cause:** Software error.

**Action:** See secondary error.

**4902, Object not found in clipboard for key {0}.**

**Cause:** Software error.

**Action:** See secondary error.

**4903, The minimum succeed value of {0} was not met for the workers in group {1}.**

**Cause:** A group of workers belonging to the same group requires that a minimum number of them succeed. That minimum succeed value was not met.

**Action:** See secondary error.

**4950, An error occurred on host {0} with IP {1} and port {2}.**

**Cause:** An error occurred on the server.

**Action:** See secondary error.

## B.2.9 Warning Error Messages

The following are warning error messages.

**15305, Warning: Problem gathering summary information for backup.**

**Cause:** Error during the gatherInfo step of the backup topology operation.

**Action:** Check secondary error.

**15306, Warning during undo processing.**

**Cause:** Error occurred during undo processing.

**Action:** Check secondary error.

## B.2.10 OracleAS Database Error Messages

The following are OracleAS database error messages.

**15604, Error finishing up creating the physical standby database.**

**Cause:** Failed to finish creating the standby database.

**Action:** See secondary error.

**15605, Error creating the physical standby database.**

**Cause:** Failed to the create the standby database.

**Action:** See secondary error.

**15606, Failed to perform a sync database operation on the primary topology.**

**Cause:** Failed to perform a sync database operation on the primary topology.

**Action:** See secondary error.

**15607, Failed to perform a sync database operation on the standby topology.**

**Cause:** Failed to perform a sync database operation on the standby topology.

**Action:** See secondary error and log file for more information.

**15608, Invalid backup mode specified in the template file {0}.**

**Cause:** Error trying to process a backup mode value in the template file.

**Action:** Correct backup mode and retry the operation.

**15609, Failed to get database backup files.**

**Cause:** Error trying to get the database backup files.

**Action:** See secondary error.

### B.2.11 OracleAS Topology Error Messages

The following are OracleAS topology error messages.

**15620, An Invalid Topology was specified.**

**Cause:** Error trying to process a topology object.

**Action:** Retrieve a valid topology object.

**15621, Error trying to verify topology {0}.**

**Cause:** The specified topology had an error during the verify operation.

**Action:** See secondary error for more information.

**15622, Error trying to verify instance {0}.**

**Cause:** The specified instance had an error during the verify operation.

**Action:** See secondary error for more information.

**15623, Topology {0} is not symmetrical with topology {1}.**

**Cause:** The specified topologies are not symmetrical.

**Action:** See secondary error for more information.

**15624, An Invalid Topology was specified. Topology {0} does not contain any valid instances.**

**Cause:** Error trying to process a topology object. Topology object did not contain a valid instance.

**Action:** Retrieve a valid topology object with at least one instance.

**15625, Could not find matching instance {0} in Topology {1}.**

**Cause:** Could not get matching instances. Topologies do not appear to be symmetrical.

**Action:** Make the topologies symmetrical.

**15626, Topologies are not symmetrical because topology name {0} is not the same as topology name {1}.**

**Cause:** Topology names are not the same and therefore topologies are not symmetrical.

**Action:** Make the topologies symmetrical.

**15627, Instance {0} is not symmetrical because of different Oracle home names {1}.**

**Cause:** Instance Home names are not symmetrical in the specified topologies.

**Action:** Make the topologies symmetrical.

**15628, Instance {0} is not symmetrical because of different Oracle home paths {1}.**

**Cause:** Instance Home paths are not symmetrical in the specified topologies.

**Action:** Make the topologies symmetrical.

**15629, Instance {0} is not symmetrical, because of different host names {1}, {2}.**

**Cause:** Instance host names are not symmetrical in the specified topologies.

**Action:** Make the topologies symmetrical.

**15630, The specified instance {0} could not be found.**

**Cause:** The specified instance information could not be found on this node.

**Action:** Either the wrong instance name or host name was specified on the request to the server.

**15631, The primary and standby topologies appear to be identical because both have instance {0} on host {1}.**

**Cause:** An instance can only be in a member of one topology, it appears that the primary and standby topologies are the same.

**Action:** Specify a primary and separate standby topology.

**15632, The Home that contains instance {0} could not be found.**

**Cause:** The specified instance could not be found in any Home on this node.

**Action:** The Oracle home information on the system is incorrect.

**15633, An Invalid Topology was specified. Topology contains a duplicate instance named {0}.**

**Cause:** Topology information obtained from OPMN contains a duplicate instance.

**Action:** Check OPMN to ensure that the topology information listed is correct.

## B.2.12 OracleAS Backup and Restore Error Messages

The following are OracleAS backup and restore error messages.

**15681, Must specify a backup directory.**

**Cause:** A backup directory must be specified for the operation to complete successfully.

**Action:** Check secondary error.

**15682, Failed to initialize configure file: {0}.**

**Cause:** Failed to initialize the configure file for backup script.

**Action:** Check secondary error.

**15683, The ha directory does not exist in Oracle home {0}.**

**Cause:** The ha directory does not exist in the OracleAS Oracle home.

**Action:** Make sure the ha directory which contains the backup and restore scripts is copied to the OracleAS Oracle home.

**15684, Failed to generate the configuration file for the backup and restore script.**

**Cause:** Failed to generate the configure file for the backup and restore script.

**Action:** Check secondary error.

**15685, Failed to backup configuration data for instance {0}.**

**Cause:** Failed to backup configuration data for the specified instance.

**Action:** Check secondary error.

**15686, Failed to restore configuration data for instance {0}.**

**Cause:** Failed to restore configuration data for the specified instance.

**Action:** Check secondary error.

**15687, Failed to get the database backup files.**

**Cause:** Failed to get the database backup file names from the log.

**Action:** Check secondary error.

**15688, Error running the config script.**

**Cause:** Failed to run the config script.

**Action:** Check the log file generated by the config script.

**15689, Error running the backup script.**

**Cause:** Failed to run the backup script.

**Action:** Check the log file generated by the backup script.

**15690, Error running the restore script.**

**Cause:** Failed to run the restore script.

**Action:** Check the log file generated by the restore script.

**15691, No zip file was found.**

**Cause:** No zip file was found.

**Action:** Make sure a successful backup has been performed.

**15692, The config file {0} is empty.**

**Cause:** The specified configure file is empty.

**Action:** Copy the original configure file from the "ha" directory where backup restore scripts are located.

**15693, No zip file was specified.**

**Cause:** User did not specify a zip file for the unzip operation.

**Action:** Internal error.

**15694, Error executing step - {0} of Backup topology.**

**Cause:** Backup topology failed at the specified step.

**Action:** Check secondary error.

**15695, Error executing step - {0} of Restore topology.**

**Cause:** Restore topology failed at the specified step.

**Action:** Check secondary error.

**15696, Error initializing backup topology operation.**

**Cause:** Error initializing backup topology operation.

**Action:** Check secondary error.

**15697, Error during backup topology operation - backup step.**

**Cause:** Error during backup step processing of backup topology.

**Action:** Check secondary error.

**15698, Error during backup topology operation - copy step.**

**Cause:** Error during copy step processing of backup topology.

**Action:** Check secondary error.

**15699, Error initializing restore topology operation.**

**Cause:** Error initializing restore topology operation.

**Action:** Check secondary error.

**15700, No backup file was found.**

**Cause:** No backup file was found.

**Action:** Make sure a successful backup has been performed.

**15701, Failed to restore configuration with the DCM-resyncforce option for instance {0}.**

**Cause:** Failed to restore configuration with the DCM-resyncforce option.

**Action:** Check secondary error.

**15702, Error initializing the clone instance operation.**

**Cause:** Error initializing the clone instance operation.

**Action:** Check the secondary error.

**15703, Error initializing the clone topology operation.**

**Cause:** Error initializing the clone home operation.

**Action:** Check the secondary error.

**15704, Error: Oracle home of the instance to be cloned {0} already exists.**

**Cause:** Error cloning instance, the Oracle home already exists

**Action:** Clean up the Oracle home and retry.

**15705, cloning instance {0}. Cloning requires OPMN to be stopped, therefore the OracleAS Guard server must be started using asgctl .**

**Cause:** Cloning requires that OPMN be stopped, which will cause the OracleAS Guard server (ASG server process) to be stopped. This will cause the clone to fail.

**Action:** Use opmnctl to stop the OracleAS Guard server (ASG server process). Then use the asgctl startup topology command to restart OracleAS Guard server for this instance.

**15706, Stop the backup home image operation in response to the user's request.**

**Cause:** Stop the backup home operation because the user entered NO.

**Action:** None.

**15707, Stop the restore home image operation in response to the user's request.**

**Cause:** Stop the restore home operation because the user entered NO.

**Action:** None.

### B.2.13 Oracle Application Server Guard Synchronize Error Messages

The following are Oracle Application Server Guard synchronize error messages.

**15721, Failed to initialize a DUF database object.**

**Cause:** Failed to initialize a DufDb object.

**Action:** Check secondary error.

**15722, No topology information is available to perform the topology operation.**

**Cause:** No topology information is available to perform the topology operation.

**Action:** Check secondary error.

**15723, No instances are found in the topology's backup list.**

**Cause:** The topology's backup list is empty.

**Action:** Check secondary error.

**15724, Failed to get the standby host list.**

**Cause:** Failed to get the standby host list.

**Action:** Check secondary error.

**15725, Failed to backup OracleAS configuration data for topology {0}.**

**Cause:** Failed to backup OracleAS topology configuration data.

**Action:** Check secondary error.

**15726, Failed to restore OracleAS configuration data for topology.**

**Cause:** Failed to restore OracleAS topology configuration data.

**Action:** Check secondary error.

**15727, Failed to backup OracleAS infrastructure database {0}.**

**Cause:** Failed to backup OracleAS topology infrastructure database.

**Action:** Check secondary error.

**15728, Failed to restore OracleAS infrastructure database {0}.**

**Cause:** Failed to restore OracleAS topology infrastructure database.

**Action:** Check secondary error.

**15729, Failed to perform the sync topology operation.**

**Cause:** Failed to perform the sync topology operation.

**Action:** Check secondary error.

### B.2.14 Oracle Application Server Guard Instantiate Error Messages

The following are Oracle Application Server Guard instantiate error messages.

**15751, Error executing step {0} of instantiate topology operation.**

**Cause:** The instantiate topology operation failed at the specified step.

**Action:** Check secondary error.

**15752, Failed to load remote topology information.**

**Cause:** Failed to load remote topology information.

**Action:** Make sure the user specified the correct host name for the topology and that the OPMN processes are running on the topologies.

**15753, Error preparing to instantiate topology on host {0}.**

**Cause:** Error preparing to instantiate topology.

**Action:** Check secondary error.

**15754, Error instantiating database {0}.**

**Cause:** Error instantiating the database.

**Action:** Check secondary error.

**15755, Error finishing up instantiating database {0}.**

**Cause:** Error finishing up instantiating the database.

**Action:** Check secondary error.

**15756, Error initializing instantiate topology operation.**

**Cause:** Error initializing the instantiate topology operation.

**Action:** Check secondary error.

**15757, Error initializing switchover topology operation.**

**Cause:** Error initializing the switchover topology operation.

**Action:** Check secondary error.

**15770, The instance {0} specified in the topology file does not match the instance {1} in home {2}.**

**Cause:** The topology file is incorrect.

**Action:** Run the "discover topology command" from asgctl.

**15771, The topology file {0} is the wrong version  Please delete the file and rediscover the topology.**

**Cause:** The topology file is incorrect.

**Action:** Run the "discover topology command" from asgctl.

**15772, The topology file {0} does not contain an entry for the discovery host {1}.**

**Cause:** The topology file is incorrect.

**Action:** Run the "discover topology command" from asgctl.

**15773, The standby topology does not contain a entry for the mandatory primary instance {0}.**

**Cause:** The topology file is incorrect.

**Action:** Run the "discover topology command" from asgctl.

**15774, The host name {0} in the standby topology net descriptor for database {1} resolves to a primary host address {2}.**

**Cause:** The topology file is incorrect.

**Action:** Run the "discover topology command" from asgctl.

**15775, The standby topology host name {0} for the instance {1} resolves to a primary host address.**

**Cause:** The topology file is incorrect.

**Action:** Run the "discover topology command" from asgctl.

**15776, Error accessing the OID server.**

**Cause:** Unable to access the Oracle Internet Directory server.

**Action:** Specify the correct Oracle Internet Directory information and make sure the Oracle Internet Directory server is running.

**15777, Error: OID information needed to access the server was not specified.**

**Cause:** Unable to access the Oracle Internet Directory server.

**Action:** Specify the correct Oracle Internet Directory information.

**15778, Error getting database information for SID {0} from host {1}. This instance will be excluded from the topology.xml file.**

**Cause:** Unable to get database information for the topology database.

**Action:** None.

**15779, Error getting instance information for instance {0} from host {1}. This instance will be excluded from the topology.xml file.**

**Cause:** Unable to get information for an instance.

**Action:** None.

**15780, Instance {0} cannot be found in the topology.**

**Cause:** The instance name does not exist in the topology file.

**Action:** Perform the asgctl discover topology command.

**15781, Warning: Unable to update topology file on host {0} in home {1}.**

**Cause:** The topology file cannot be written to the home.

**Action:** Make sure the OracleAS Guard server is running in that home.

**15782, Error: Instance {0} already exists in the topolgy.**

**Cause:** The instance already exists in the topology.

**Action:** Remove the instance using asgctl.

**15783, Error: OID host and port information needed to access the server was not specified.**

**Cause:** Unable to access the Oracle Internet Directory server.

**Action:** Specify the correct Oracle Internet Directory information.

# C

# Sync Operations Automated by OracleAS Disaster Recovery

The OracleAS Disaster Recovery solution uses Oracle Data Guard, OracleAS Recovery Manager, Oracle Application Server Guard, and other technologies to automate the synchronization of the OracleAS Disaster Recovery production and standby sites. This chapter describes the operations performed to keep the production and standby sites synchronized.

To keep the OracleAS Disaster Recovery production and standby sites synchronized, you must perform these steps:

1. Install and configure the OracleAS middle tier and Infrastructure instances, as well as any Oracle databases you want to include in your OracleAS Disaster Recovery topology at the production and standby sites. Refer to Chapter 1, "OracleAS Disaster Recovery Introduction" for more information about installing and configuring Oracle Application Server instances and Oracle databases for OracleAS Disaster Recovery.
2. Use the asgctl `discover topology` command at the production site to create the OracleAS Disaster Recovery topology file. Refer to discover topology for more information about the command.
3. Implement the appropriate backup and recovery strategy for each of the Oracle Application Server instances in your OracleAS Disaster Recovery topology using OracleAS Recovery Manager (previously known as the Backup and Recovery tool), using information provided in the *Oracle Application Server Administrator's Guide*. OracleAS Recovery Manager is installed by default as part of every Oracle Application Server instance installation.
4. Make sure that Oracle Data Guard is configured for the Oracle databases in your OracleAS Disaster Recovery topology. Read Section 1.1.1.1, "Configuring Oracle Data Guard for Databases in an OracleAS Disaster Recovery Topology" for more information about configuring Oracle Data Guard for the Oracle databases in your OracleAS Disaster Recovery topology. The asgctl `create standby database` command can be used to configure Oracle Data Guard for some databases.
5. Use the asgctl `instantiate topology` command to establish the initial relationship between the production site and standby site Application Server instances, mirror the configuration, and synchronize the standby site with the production site. Refer to instantiate topology for more information about the command.
6. Before the OracleAS Disaster Recovery production site is in use, perform an asgctl `sync topology` command to create a baseline snapshot of the production site, which can be used later to recover the production site configuration on the

standby site, if necessary. Refer to sync topology for more information about the command. The `sync topology` command automates these operations:

a. Validates the communication path from the production site to the standby site.

b. Ensures that the symmetric topology or asymmetric topology for the production site and standby site that is defined in the OracleAS topology file is maintained.

c. If a policy file is specified with the command, uses the policy file to perform the synchronization specified by the policy file.

d. Confirms that Oracle Data Guard ships OracleAS Infrastructure database archive logs from the production site to the standby site. Refer to Section C.1, "Shipping OracleAS Infrastructure Database Archive Logs to the Standby Site" for more information.

e. Backs up OracleAS Infrastructure and middle tier configuration files at the production site. Refer to Section C.2, "Backing Up OracleAS Infrastructure and Middle Tier Configuration Files at the Production Site" for more information.

f. Restores OracleAS Infrastructure and middle tier configuration files at the standby site. Refer to Section C.3, "Restoring OracleAS Infrastructure and Middle Tier Configuration Files at the Standby Site" for more information.

g. Recovers OracleAS Infrastructure database archive logs at the standby site. Refer to Section C.4, "Restoring OracleAS Infrastructure Database Archive Logs at the Standby Site" for more information.

---

**Note:** When the `sync topology` command executes, it synchronizes configuration information between the production and standby sites by synchronizing the backup of OracleAS Infrastructure and middle-tier configuration files with the application of log information on the standby OracleAS Infrastructure database. These operations maintain logical consistency between the configuration of the Oracle Application Server instances and the database.

For Oracle Application Server, not all the configuration information is in the OracleAS Infrastructure database. The backup of the database files must be kept synchronized with the backup of the middle-tier and OracleAS Infrastructure configuration files. Due to this, log-apply services should not be enabled on the standby database. The log files from the production OracleAS Infrastructure are shipped to the standby OracleAS Infrastructure but are not applied.

---

7. Execute the `sync topology` command whenever there is any administration change in the production site (including changes to the OracleAS Infrastructure database and configuration files on the middle-tier and OracleAS Infrastructure nodes). Also, perform scheduled backup and restore operations (for example, on a daily or twice weekly basis). See the *Oracle Application Server Administrator's Guide* for more backup and restore procedures.

---

**Note:** Ensure that no configuration changes are being made or will be made to the Oracle Application Server system (underlying configuration files and OracleAS Infrastructure database) when you perform the `sync topology` command.

---

## C.1 Shipping OracleAS Infrastructure Database Archive Logs to the Standby Site

The sync topology command performs the following operations to initiate the transfer of the archive logs for the OracleAS Infrastructure database to the standby site host:

1. If log-apply services are not disabled already at the standby site host, the following SQL*Plus command is run on the standby site host to disable the log-apply services:

   ```
   SQL> alter database recover managed standby database cancel;
   ```

2. The following command is run to perform a log switch on the production OracleAS Infrastructure database. This ensures that Oracle Data Guard ships the latest log file to the standby OracleAS Infrastructure database

   ```
   SQL> alter system switch logfile;
   ```

3. In normal operation of the production site, the production database frequently ships log files to the standby database but they are not applied. At the standby site, the logs are applied to a consistency level. The following SQL statement discovers the System Change Number (SCN) level of the production OracleAS Infrastructure database,  encapsulates all the database changes into the latest log, and allows the Oracle Data Guard transport services to transport this log to the OracleAS Infrastructure in the standby site:

   ```
   SQL> select first_change# from v$log where status='CURRENT';
   ```

   The SCN returned by this statement essentially represents the timestamp of the transported log. This SCN is used as the sync point of the `sync topology` command.

4. This SCN is used later for the restoration of the production database changes on the standby site, which is described in Section C.4, "Restoring OracleAS Infrastructure Database Archive Logs at the Standby Site."

5. The previous steps are repeated for each database configured for Oracle Data Guard in the OracleAS Disaster Recovery topology.

## C.2 Backing Up OracleAS Infrastructure and Middle Tier Configuration Files at the Production Site

Next, OracleAS Recovery Manager backs up the OracleAS Infrastructure and middle tier configuration files. This step assumes you have installed and configured OracleAS Recovery Manager on each OracleAS installation (middle tier and OracleAS Infrastructure), as it must be customized for each installation. Refer to *Oracle Application Server Administrator's Guide* for more details about OracleAS Recovery Manager, including installation and configuration instructions.

**Note:** After performing the installation and configuration steps for the OracleAS Recovery Manager, which are described in the *Oracle Application Server Administrator's Guide*, the variables `oracle_home`, `log_path`, and `config_backup_path` in the OracleAS Recovery manager's configuration file, `config.inp`, have the appropriate values. Also, the following command for the OracleAS Recovery Manager should have been run to complete the configuration:

```
perl bkp_restore.pl -m configure_nodb
```

For each middle-tier and OracleAS Infrastructure installation, the following steps are performed (the same steps are used for the OracleAS Infrastructure and middle tier configuration files):

1. The following command is executed to back up the configuration files from the current installation:

   ```
   perl bkp_restore.pl -v -m backup_config
   ```

   This command creates a directory in the location specified by the `config_backup_path` variable specified in the `config.inp` file. The directory name includes the time of the backup. For example: `config_bkp_2003-09-10_13-21`.
2. A log of the backup is also generated in the location specified by the `log_path` variable in the `config.inp` file. Any errors that occur during the backup process are written to the log files.
3. The OracleAS Recovery Manager's directory structure and contents from the current host are copied to the equivalent directories on the standby site host, which ensures that the path structure on the standby host is identical to that on the production site host.
4. The backup directory (as defined by `config_backup_path`) is copied from the current host to the equivalent directory on the standby site host, which ensures that the path structure on the standby host is identical to that on the production site host.
5. The previous steps are repeated each Oracle Application Server installation in the production site (OracleAS Infrastructure and middle tier).

## C.3 Restoring OracleAS Infrastructure and Middle Tier Configuration Files at the Standby Site

The OracleAS Recovery Manager used to back up the configuration files at the production site restores the backed up files at the standby site.

For each middle-tier and OracleAS Infrastructure installation in the standby site, the following steps are performed (the same steps are used for the OracleAS Infrastructure and middle tier configuration files):

1. A check is performed to confirm that the OracleAS Recovery Manager directory structure and the backup directory are the same on the production site host and standby site host.

2. The following OPMN command is used to stop the Oracle Application Server instances and their processes so that no modification of configuration files can occur during the restoration process:

   In UNIX:

   ```
   <ORACLE_HOME>/opmn/bin/opmnctl stopall
   ```

   In Windows:

   ```
   <ORACLE_HOME>\opmn\bin\opmnctl stopall
   ```

3. OracleAS Recovery Manager executes the following command to determine the valid configuration backup locations:

   ```
   perl bkp_restore.pl -v -m restore_config
   ```

4. OracleAS Recovery Manager executes the following command to restore the configuration files:

   ```
   perl bkp_restore.pl -v -m restore_config -t <backup_directory>
   ```

   where <*backup_directory*> is the name of the directory with the backup files that was copied from the production site. For example, this could be `config_bkp_2003-09-10_13-21`.

5. Any errors that occur during the restoration process are written to the log file specified in `config.inp`.

6. The previous steps are repeated for each Oracle Application Server installation in the production site (OracleAS Infrastructure and middle tier).

## C.4 Restoring OracleAS Infrastructure Database Archive Logs at the Standby Site

During the backup phase described in Section C.1, "Shipping OracleAS Infrastructure Database Archive Logs to the Standby Site," several commands shipped the database log files from the production site to the standby site up to the latest SCN number. A SQL*Plus statement similar to the following statement restores the OracleAS Infrastructure database up to the SCN number at the standby site host by applying the log files to the standby database:

```
SQL> alter database recover managed standby database from '/private/oracle/oracleas/standby/'
standby database until change <SCN>;
```

Each database in the OracleAS Disaster Recovery topology that is configured with Oracle Data Guard is restored up to the appropriate SCN number using a similar SQL*Plus statement.

# Glossary

### clusterware

A software that manages the operations of the members of a cluster as a system. It allows one to define a set of resources and services to monitor through a heartbeat mechanism between cluster members and to move these resources and services to a different member in the cluster as efficiently and transparently as possible.

### hardware cluster

A hardware cluster is a collection of computers that provides a single view of network services (for example: an IP address) or application services (for example: databases, Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

A hardware cluster achieves high availability and scalability through the use of specialized hardware (cluster interconnect, shared storage) and software (health monitors, resource monitors). (The cluster interconnect is a private link used by the hardware cluster for heartbeat information to detect node death.) Due to the need for specialized hardware and software, hardware clusters are commonly provided by hardware vendors such as Sun, HP, IBM, and Dell. While the number of nodes that can be configured in a hardware cluster is vendor dependent, only two nodes are required for Oracle Application Server high availability. Hence, this document assumes a two-node hardware cluster for high availability solutions employing a hardware cluster.

### network hostname

Network hostname is a name assigned to an IP address either through the `/etc/hosts` file (on UNIX), `C:\WINDOWS\system32\drivers\etc\hosts` file (on Windows), or through DNS resolution. This name is visible in the network that the system which it refers to is connected. A system may have the same network hostname and physical hostname. However, although a system has only one physical hostname, it may have multiple network hostnames. Thus, a system’s network hostname may not always be its physical hostname.

### physical hostname

This guide differentiates between the terms physical hostname and network hostname. This guide uses physical hostname to refer to the "internal name" of the current system. The physical hostname is the name returned by the `hostname` command.

The physical hostname is used by Oracle Application Server to reference the local host. During installation, the installer automatically retrieves the physical hostname from the current system and stores it in the Oracle Application Server configuration metadata on disk.

### shared storage

Although each node in a hardware cluster is a standalone server that runs its own set of processes, the storage subsystem required for any cluster-aware purpose is usually shared. Shared storage refers to the ability of the cluster to be able to access the same storage, usually disks, from either node in the hardware cluster. While the nodes have equal access to the storage, only one node, the primary node, has active access to the storage at any given time. The hardware cluster's software grants the secondary node access to this storage if the primary node fails.

In OracleAS Cold Failover Cluster (Middle-Tier) environments, you can install the Oracle home directory on a shared storage system or on a local storage of each node in the hardware cluster.

### virtual hostname

Virtual hostname is a network addressable hostname that maps to one or more physical systems through a load balancer or a hardware cluster. For load balancers, the name "virtual server name" is used interchangeably with virtual hostname in this book. A load balancer can hold a virtual hostname on behalf of a set of servers, and clients communicate indirectly with the systems using the virtual hostname. A virtual hostname in a hardware cluster is a network hostname assigned to a cluster virtual IP. Because the cluster virtual IP is not permanently attached to any particular node of a cluster, the virtual hostname is not permanently attached to any particular node either.

Note that when the term "virtual hostname" is used in this document, it is assumed to be associated with a virtual IP address. In cases where just the IP address is needed or used, it will be explicitly stated.

### virtual IP

Also, cluster virtual IP and load balancer virtual IP. Generally, a virtual IP can be assigned to a hardware cluster or load balancer. To present a single system view of a cluster to network clients, a virtual IP serves as an entry point IP address to the group of servers which are members of the cluster. A virtual IP can be assigned to a server load balancer or a hardware cluster.

A hardware cluster uses a cluster virtual IP to present to the outside world the entry point into the cluster (it can also be set up on a standalone system). The hardware cluster's software manages the movement of this IP address between the two physical nodes of the cluster while clients connect to this IP address without the need to know which physical node this IP address is currently active on. In a typical two-node hardware cluster configuration, each system has its own physical IP address and physical hostname, while there could be several cluster IP addresses. These cluster IP addresses float or migrate between the two nodes. The node with current ownership of a cluster IP address is active for that address.

A load balancer also uses a virtual IP as the entry point to a set of servers. These servers tend to be active at the same time. This virtual IP address is not assigned to any individual server but to the load balancer which acts as a proxy between servers and their clients.

# Index