Oracle® Solaris Cluster Data Service for Oracle Real Application Clusters Guide

Part No: 821–1262–10
September 2010, Revision A

ORACLE

110714@25097

Contents

Figures

Tables

Examples

Preface

Oracle Solaris Cluster Data Service for Oracle Real Application Clusters Guide explains how to install and configure Support for Oracle RAC.

Note – This Oracle Solaris Cluster release supports systems that use the SPARC and x86 families of processor architectures: UltraSPARC, SPARC64, AMD64, and Intel 64. In this document, x86 refers to the larger family of 64-bit x86 compatible products. Information in this document pertains to all platforms unless otherwise specified.

This document is intended for system administrators with extensive knowledge of Oracle software and hardware. Do not use this document as a planning or presales guide. Before reading this document, you should have already determined your system requirements and purchased the appropriate equipment and software.

The instructions in this book assume knowledge of the Oracle Solaris Operating System and expertise with the volume-manager software that is used with Oracle Solaris Cluster software.

Using UNIX Commands

This document contains information about commands that are specific to installing and configuring Oracle Solaris Cluster data services. The document does *not* contain comprehensive information about basic UNIX commands and procedures, such as shutting down the system, booting the system, and configuring devices. Information about basic UNIX commands and procedures is available from the following sources:

- Online documentation for the Oracle Solaris Operating System
- Oracle Solaris Operating System man pages
- Other software documentation that you received with your system

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P–1 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`** `Password:`
aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for shells that are included in the Oracle Solaris OS. Note that the default system prompt that is displayed in command examples varies, depending on the Oracle Solaris release.

TABLE P–2 Shell Prompts

Shell	Prompt
Bash shell, Korn shell, and Bourne shell	`$`
Bash shell, Korn shell, and Bourne shell for superuser	`#`
C shell	`machine_name%`
C shell for superuser	`machine_name#`

Related Documentation

Information about related Oracle Solaris Cluster topics is available in the documentation that is listed in the following table. All Oracle Solaris Cluster documentation is available at http://docs.sun.com.

Topic	Documentation
Data service administration	*Oracle Solaris Cluster Data Services Planning and Administration Guide* Individual data service guides
Concepts	*Oracle Solaris Cluster Concepts Guide*
Overview	*Oracle Solaris Cluster Overview*
Software installation	*Oracle Solaris Cluster Software Installation Guide*
System administration	*Oracle Solaris Cluster System Administration Guide*
Hardware administration	*Oracle Solaris Cluster 3.3 Hardware Administration Manual* Individual hardware administration guides
Data service development	*Oracle Solaris Cluster Data Services Developer's Guide*
Error messages	*Oracle Solaris Cluster Error Messages Guide*
Command and function reference	*Oracle Solaris Cluster Reference Manual*

For a complete list of Oracle Solaris Cluster documentation, see the release notes for your release of Oracle Solaris Cluster at http://docs.sun.com.

Related Third-Party Web Site References

Third-party URLs that are referenced in this document provide additional related information.

Note – Oracle is not responsible for the availability of third-party web sites mentioned in this document. Oracle does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Oracle will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation, Support, and Training

See the following web sites for additional resources:

- Documentation (`http://docs.sun.com`)
- Support (`http://www.oracle.com/us/support/systems/index.html`)
- Training (`http://education.oracle.com`) – Click the Sun link in the left navigation bar.

Oracle Welcomes Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of its documentation. If you find any errors or have any other suggestions for improvement, go to `http://docs.sun.com` and click Feedback. Indicate the title and part number of the documentation along with the chapter, section, and page number, if available. Please let us know if you want a reply.

Oracle Technology Network (`http://www.oracle.com/technetwork/index.html`) offers a range of resources related to Oracle software:

- Discuss technical problems and solutions on the Discussion Forums (`http://forums.oracle.com`).
- Get hands-on step-by-step tutorials with Oracle By Example (`http://www.oracle.com/technology/obe/start/index.html`).
- Download Sample Code (`http://www.oracle.com/technology/sample_code/index.html`).

Getting Help

If you have problems installing or using Oracle Solaris Cluster, contact your service provider and provide the following information:

- Your name and email address (if available)
- Your company name, address, and phone number
- The model number and serial number of your systems
- The release number of the Oracle Solaris Operating System (for example, Oracle Solaris 10)
- The release number of Oracle Solaris Cluster (for example, Oracle Solaris Cluster 3.3)

Use the following commands to gather information about each node on your system for your service provider.

Command	Function
`prtconf -v`	Displays the size of the system memory and reports information about peripheral devices
`psrinfo -v`	Displays information about processors
`showrev –p`	Reports which patches are installed
`prtdiag -v`	Displays system diagnostic information
`/usr/cluster/bin/clnode show-rev`	Displays Oracle Solaris Cluster release and package version information

Also have available the contents of the `/var/adm/messages` file.

◆ ◆ ◆ CHAPTER 1

Installing Support for Oracle RAC

This chapter explains how to install Support for Oracle RAC on your Oracle Solaris Cluster nodes.

Overview of the Installation Process for Support for Oracle RAC

The following table summarizes the installation tasks and provides cross-references to detailed instructions for performing the tasks.

Perform these tasks in the order in which they are listed in the table.

TABLE 1–1 Tasks for Installing Support for Oracle RAC

Task	Instructions
Plan your installation	"Preinstallation Considerations" on page 24
Prepare the Oracle Solaris Cluster nodes	"Preparing the Oracle Solaris Cluster Nodes" on page 31
Install data service packages	"Installing the Support for Oracle RAC Packages" on page 40
SPARC: Install the Oracle UNIX Distributed Lock Manager (Oracle UDLM)	"SPARC: Installing the Oracle UDLM" on page 42

Preinstallation Considerations

Oracle Real Application Clusters (Oracle RAC) is an application that can run on more than one machine concurrently. Oracle RAC can run either in the global-cluster voting nodes of the global cluster or in a zone cluster. An Oracle RAC installation is entirely contained within one cluster, either a global cluster or a specific zone cluster. Keeping the Oracle RAC installation in one cluster ensures the support of multiple independent Oracle RAC installations concurrently, where each Oracle RAC installation can be of a different version or use different options, such as storage. Support for Oracle RAC enables you to run Oracle RAC on Oracle Solaris Cluster nodes and to manage Oracle RAC by using Oracle Solaris Cluster commands.

Configuring this data service involves configuring resources for the following components of an Oracle RAC installation with Oracle Solaris Cluster software:

- **The RAC framework.** These resources enable Oracle RAC to run with Oracle Solaris Cluster software. The resources also enable reconfiguration parameters to be set by using Oracle Solaris Cluster commands. You *must* configure resources for the RAC framework. For more information, see "Registering and Configuring the RAC Framework Resource Group" on page 65.
- **Storage for Oracle files.** These resources provide fault monitoring and automatic fault recovery for volume managers and file systems that store Oracle files. Configuring storage resources for Oracle files is optional. For more information, see "Registering and Configuring Storage Resources for Oracle Files" on page 82.
- **Oracle RAC database instances.** The features that these resources provide depend on the version of Oracle that you are using:
  - **Oracle 10*g* release 2 and 11*g*.** These resource types enable Oracle Clusterware and Oracle Solaris Cluster software to interoperate. These resource types do *not* provide fault monitoring and automatic fault recovery for Oracle Solaris Cluster 10*g* release 2 or 11*g*. The Oracle Clusterware software provides this functionality. This statement is not applicable to Oracle 10*g* release 1.

    Note – Unless otherwise stated, instructions and information for Oracle 11*g* apply to both Oracle 11*g* release 1 and Oracle 11*g* release 2.

  - **Oracle 9*i*.** These resource types provide fault monitoring and automatic fault recovery for Oracle RAC. The automatic fault recovery that these resource types provide supplements the automatic fault recovery that the Oracle RAC software provides.

  Configuring resources, which enable Oracle Solaris Cluster software to administer Oracle RAC database instances, is optional. For more information, see "Configuring Resources for Oracle RAC Database Instances" on page 107.

Note – When you use Oracle RAC in a zone cluster, ensure that the zone cluster is not configured with the `/opt` directory as an inherited read-only directory. For an Oracle RAC configuration in a zone cluster, the `/opt` file system must be writable and unique to each zone. If the zone cluster that you intended to use for Oracle RAC is configured with an `inherit-pkg-dir` resource for the `/opt` directory, destroy and recreate the zone cluster or create a new zone cluster to meet this requirement.

The following sections contain additional preinstallation information:

Hardware and Software Requirements

Before you begin the installation, note the hardware and software requirements in the subsections that follow.

Oracle Solaris Cluster Framework Requirements

Support for Oracle RAC requires a functioning cluster with the initial cluster framework already installed. See *Oracle Solaris Cluster Software Installation Guide* for details about initial installation of cluster software.

Software License Requirements

Verify that you have obtained and installed the appropriate licenses for your software. If you install your licenses incorrectly or incompletely, the nodes might fail to boot correctly.

For example, if you are using VxVM with the cluster feature, verify that you have installed a valid license for the Volume Manager cluster feature by running the `vxlicrep` command.

Supported Topology Requirements

Check with a Sun Enterprise Services representative for the current supported topologies for Support for Oracle RAC, cluster interconnect, storage management scheme, and hardware configurations.

Patch Installation Requirements

Ensure that you have installed all the applicable software patches for the Solaris OS, Oracle Solaris Cluster, Oracle, and volume manager software. If you need to install any Support for Oracle RAC patches, you must apply these patches after you install the data service packages.

Storage Management Requirements for Oracle Files

This section provides the following information about storage management for Oracle RAC:

Storage Management Schemes for Oracle Files

Support for Oracle RAC enables you to use the storage management schemes for Oracle files that are listed in the following table. The table summarizes the types of Oracle files that each storage management scheme can store. Ensure that you choose a combination of storage management schemes that can store all types of Oracle files.

The meaning of each symbol in the table is as follows:

+ Indicates that the storage management scheme can store the type of Oracle file.

- Indicates that the storage management scheme *cannot* store the type of Oracle file.

TABLE 1–2 Storage Management Schemes for Oracle Files

Oracle File Type	Storage Management Scheme							
	Solaris Volume Manager for Sun Cluster	VxVM	Hardware RAID	Sun QFS Shared File System	Qualified NAS Devices	Oracle ASM	Cluster File System	Local Disks
RDBMS binary files	-	-	-	+	+	-	+	+
Oracle Clusterware binary files	-	-	-	+	+	-	+	+
Configuration files	-	-	-	+	+	-	+	+
System parameter file (`SPFILE`)	-	-	-	+	+	+	+	-
Alert files	-	-	-	+	+	-	+	+
Trace files	-	-	-	+	+	-	+	+
Data files	+	+	+	+	+	+	-	-
Control files	+	+	+	+	+	+	-	-
Online redo log files	+	+	+	+	+	+	-	-
Archived redo log files	-	-	-	+	+	+	+	-
Flashback log files	-	-	-	+	+	+	+	-
Recovery files	-	-	-	+	+	+	-	-
OCR files	+	+	+	+	+	-	+	-
Oracle Clusterware voting disk	+	+	+	+	+	-	+	-

Note – Some types of files are not included in all releases of Oracle RAC. For information about which types of file are included in the release that you are using, see your Oracle documentation.

Storage Management Requirements for the Oracle Clusterware

Oracle Clusterware disks such as Oracle cluster registry (OCR) and voting disks are supported on the following storage management schemes:

- Solaris Volume Manager for Sun Cluster

Note – Solaris Volume Manager for Sun Cluster is supported *only* with Oracle RAC.

- Veritas Volume Manager (VxVM)

Note – VxVM is supported *only* on the SPARC platform.

- Hardware redundant array of independent disks (RAID) support
- Sun QFS shared file systems with hardware RAID support or with Solaris Volume Manager for Sun Cluster
- Qualified network-attached storage (NAS) devices

Storage Management Requirements for the Oracle RAC Database

You can use the following storage management schemes for the Oracle RAC database:

- Solaris Volume Manager for Sun Cluster

Note – Solaris Volume Manager for Sun Cluster is supported *only* with Oracle RAC.

- Veritas Volume Manager (VxVM) with the cluster feature

Note – VxVM is supported *only* on the SPARC platform.

- Hardware redundant array of independent disks (RAID) support
- Sun QFS shared file system with hardware RAID support or with Solaris Volume Manager for Sun Cluster
- Qualified network-attached storage (NAS) devices
- Oracle Oracle Automatic Storage Management (Oracle ASM) (Oracle ASM)

Storage Management Requirements for Oracle Binary Files and Oracle Configuration Files

You can install the Oracle binary files and Oracle configuration files on one of the following locations.

- The local disks of each cluster node. See "Using Local Disks for Oracle Binary Files and Oracle Configuration Files" on page 29 for additional information.
- A shared file system from the following list:

- A Sun QFS shared file system
- A cluster file system
- A file system on a qualified NAS device

See "Using a Shared File System for Oracle Binary Files and Oracle Configuration Files" on page 29 for additional information.

Using Local Disks for Oracle Binary Files and Oracle Configuration Files

Placing the Oracle binary files and Oracle configuration files on the individual cluster nodes enables you to upgrade the Oracle application later without shutting down the data service.

Note – Some versions of Oracle require you to shut down the data service during an upgrade. To determine whether you can upgrade the Oracle application without shutting down the data service, see your Oracle documentation.

The disadvantage is that you then have several copies of the Oracle application binary files and Oracle configuration files to maintain and administer.

Using a Shared File System for Oracle Binary Files and Oracle Configuration Files

To simplify the maintenance of your Oracle installation, you can install the Oracle binary files and Oracle configuration files on a shared file system. The following shared file systems are supported:

- A Sun QFS shared file system
- A cluster file system

  If you use a cluster file system, decide which volume manager to use:

  - Solaris Volume Manager
  - VxVM *without* the cluster feature

    Note – VxVM is supported *only* on the SPARC platform.

- A file system on a qualified NAS device

If you put the Oracle binary files and Oracle configuration files on a shared file system, you have only one copy to maintain and manage. However, you must shut down the data service in the entire cluster to upgrade the Oracle application. If a short period of downtime for upgrades is acceptable, place a single copy of the Oracle binary files and Oracle configuration files on a shared file system.

Storage Management Schemes Supported by Zone Clusters

You can use the following storage management schemes for running Oracle RAC in a zone cluster, depending on the version of Oracle RAC you are running.

If you are using Oracle RAC 10*g* or Oracle RAC 11*g*, the following storage management schemes are supported:

- A Sun QFS shared file system with Solaris Volume Manager
- A Sun QFS shared file system with hardware RAID support
- Solaris Volume Manager for Sun Cluster
- A file system on a qualified NAS device with fencing

If you are using Oracle RAC 9*i*, the following storage management schemes are supported:

- A Sun QFS shared file system with Solaris Volume Manager
- A Sun QFS shared file system with hardware RAID support
- Solaris Volume Manager for Sun Cluster
- A file system on a qualified NAS device with fencing

SPARC: Processor Architecture Requirements for Oracle Components

Before you decide which architecture to use for the Oracle UDLM (SPARC) and the Oracle relational database management system (RDBMS), note the following points.

- The architecture of both Oracle components must match. For example, if you have 64-bit architecture for your Oracle UDLM, you must have 64-bit architecture for your RDBMS.
- If you have 32-bit architecture for your Oracle components, you can boot the node on which the components reside in either 32-bit mode or 64-bit mode. However, if you have 64-bit architecture for your Oracle components, you must boot the node on which the components reside in 64-bit mode.
- You must use the same architecture when you boot all the nodes. For example, if you boot one node to use 32-bit architecture, you must boot all the nodes to use 32-bit architecture.

Using Oracle Data Guard With Support for Oracle RAC

You can use Support for Oracle RAC with Oracle Data Guard. To configure Support for Oracle RAC with Oracle Data Guard, perform the tasks in this guide. The tasks for clusters that are to be used in an Oracle Data Guard configuration are identical to the tasks for a standalone cluster.

For information about the installation, administration, and operation of Oracle Data Guard, see your Oracle documentation.

Using the Oracle Real Application Clusters Guard Option With Oracle Solaris Cluster 3.3 Software

For information about the installation, administration, and operation of the Oracle Real Application Clusters Guard option, see your Oracle documentation. If you plan to use this product option with Oracle Solaris Cluster 3.3 software, note the points in the subsections that follow before you install Oracle Solaris Cluster 3.3 software.

This section describes the following restrictions when you use Oracle Real Application Clusters Guard in an Oracle Solaris Cluster configuration:

- "Hostname Restrictions" on page 31
- "Oracle Solaris Cluster Command Usage Restrictions" on page 31

Hostname Restrictions

If you use the Oracle Real Application Clusters Guard option with Oracle Solaris Cluster 3.3 software, the following restrictions apply to hostnames that you use in your cluster:

- Hostnames cannot contain special characters.

For more information about these restrictions and any other requirements, see the Oracle documentation.

Oracle Solaris Cluster Command Usage Restrictions

If you use the Oracle Real Application Clusters Guard option with Oracle Solaris Cluster 3.3 software, do *not* use Oracle Solaris Cluster commands to perform the following operations:

- Manipulating the state of resources that Oracle Real Application Clusters Guard installation. Using Oracle Solaris Cluster commands for this purpose might cause failures.
- Querying the state of the resources that Oracle Real Application Clusters Guard installation. This state might not reflect the actual state. To check the state of the Oracle Real Application Clusters Guard, use the commands that Oracle supplies.

Preparing the Oracle Solaris Cluster Nodes

Preparing the Oracle Solaris Cluster nodes modifies the configuration of the operating system to enable Oracle RAC to run on Oracle Solaris Cluster nodes. Preparing the Oracle Solaris Cluster nodes and disks involves the following tasks:

- Bypassing the NIS name service
- Creating the database administrator (DBA) group and the DBA user accounts
- Configuring shared memory for the Oracle RAC software

Caution – Perform these tasks on all nodes where Support for Oracle RAC can run. If you do not perform these tasks on all nodes, the Oracle installation is incomplete. An incomplete Oracle installation causes Support for Oracle RAC to fail during startup.

To enable the Oracle RAC to run in a zone cluster, you need to perform the following additional tasks:

- Configuring shared memory for Oracle RAC software in a zone cluster
- Setting necessary privileges for Oracle RAC software in a zone cluster
- Configuring logical hostname resources for Oracle RAC software in a zone cluster

This section contains the following information:

Before You Begin

Before you prepare the Oracle Solaris Cluster nodes, ensure that all preinstallation tasks for Oracle RAC are completed. For more information, see your Oracle RAC documentation.

▼ How to Bypass the NIS Name Service

Bypassing the NIS name service protects the Support for Oracle RAC data service against a failure of a cluster node's public network. A failure of a cluster node's public network might cause the NIS name service to become unavailable. If Support for Oracle RAC refers to the NIS name service, unavailability of the name service might cause the Support for Oracle RAC data service to fail.

Bypassing the NIS name service ensures that the Support for Oracle RAC data service does not refer to the NIS name service when the data service sets the user identifier (ID). The Support for Oracle RAC data service sets the user ID when the data service starts or stops the database.

1 Become superuser on all nodes where Support for Oracle RAC can run.

2 **On each node, include the following entries in the `/etc/nsswitch.conf` file.**

```
passwd:    files [NOTFOUND=return] nis [TRYAGAIN=0]
publickey: files [NOTFOUND=return] nis [TRYAGAIN=0]
project:   files [NOTFOUND=return] nis [TRYAGAIN=0]
group:     files [NOTFOUND=return] nis [TRYAGAIN=0]
```

For more information about the `/etc/nsswitch.conf` file, see the `nsswitch.conf(4)` man page.

Next Steps Go to "How to Create the DBA Group and the DBA User Accounts" on page 33.

How to Create the DBA Group and the DBA User Accounts

In installations of Oracle RAC with Oracle Solaris Cluster software, the DBA group is normally named `dba`. This group normally contains the `root` user and the `oracle` user.

Note – This configuration of users and groups differs from the configuration that is described in the Oracle documentation for a standalone installation of Oracle RAC. A standalone installation of Oracle RAC uses a primary DBA group that is named `oinstall` and a secondary group that is named `dba`. Some applications also require a secondary group that is named `oper`. For more information, see your Oracle documentation.

Perform this task on each cluster node.

1 **On the cluster node where you are performing this task, become superuser.**

2 **Add an entry for the DBA group and potential users in the group to the `/etc/group` file.**

```
# echo 'group-name:*:group-id:user-list' >> /etc/group
```

group-name Specifies the name of the group for which you are adding an entry. This group is normally named `dba`.

group-id Specifies the group's unique numerical ID (GID) within the system.

user-list Specifies a comma-separated list of the user names that are allowed in the group. Ensure that the list contains the names of all DBA users that are required. This group normally contains the `root` user and the `oracle` user.

Ensure that the entry is identical on each node that can run Support for Oracle RAC.

You can create the name service entries in a network name service, such as the Network Information Service (NIS) or NIS+, so that the information is available to the data service clients. You can also create entries in the local `/etc` files to eliminate dependency on the network name service.

3 Create the home directory of each potential user in the DBA group that you defined in Step 2.

You are not required to create a home directory for the `root` user.

For each potential user whose home directory you are creating, type the following command:

```
# mkdir -p user-home
```

user-home Specifies the full path of the home directory that you are creating.

4 Add each potential user in the DBA group that you defined in Step 2 to the system.

You are not required to add the `root` user.

Use the `useradd(1M)` command to add each user. Adding a user to the system adds an entry for the user to the following files:

- `/etc/passwd`
- `/etc/shadow`

```
# useradd -u user-id -g group-name -d user-home \
[ -s user-shell] user-name
```

`-u` *user-id* Specifies the user's unique numerical ID (UID) within the system.

`-g` *group-name* Specifies the name of the user group of which the user is a member. You must specify the DBA group that you defined in Step 2.

`-d` *user-home* Specifies the full path of the user's home directory. You must specify the home directory that you created for the user in Step 3.

`-s` *user-shell* Optionally specifies the full path name of the program that is to be used as the user's shell when the user logs in. If you omit the `-s` option, the system uses the `/bin/sh` program by default. If you specify the `-s` option, *user-shell* must specify a valid executable file.

user-name Specifies the user name of the user that you are adding. You must specify the name of a potential user in the DBA group that you defined in Step 2.

Ensure that each user is identical on each node that can run Support for Oracle RAC.

5 Set the password of each user that you added in Step 4.

Use the `passwd(1)` command to set the password of each user.

a. Type the following command:

```
# password user-name
```

user-name Specifies the user name of the user whose password you are setting. You must specify the name of a user in the DBA group that you added in Step 4.

The `passwd` command prompts you for the password.

b. In response to the prompt, type the password and press Return.

The `passwd` command prompts you to retype the password.

c. In response to the prompt, retype the password and press Return.

6 Change the ownership of each home directory that you created in Step 3 as follows:

- Owner: the user for which you created the home directory
- Group: the DBA group that you defined in Step 2

For each home directory for which you are changing ownership, type the following command:

```
# chown user-name:group-name user-home
```

user-name Specifies the user name of the user for whose home directory you are changing ownership. You must specify the name of a user in the DBA group that you added in Step 4.

group-name Specifies the name of the user group of which the user is a member. You must specify the DBA group that you defined in Step 2.

user-home Specifies the full path of the user's home directory. You must specify the home directory that you created for the user in Step 3.

7 Create a subdirectory of the `/var/opt` directory for each user in the DBA group that you added in Step 4.

For each subdirectory that you are creating, type the following command:

```
# mkdir /var/opt/user-name
```

user-name Specifies the user name of the user whose subdirectory of the `/var/opt` directory you are creating. You must specify the name of a user in the DBA group that you added in Step 4.

8 Change the ownership of each directory that you created in Step 7 as follows:

- Owner: the user for which you created the directory
- Group: the DBA group that you defined in Step 2

For each directory for which you are changing ownership, type the following command:

```
# chown user-name:group-name /var/opt/user-name
```

user-name Specifies the user name of the user for whose home directory you are changing ownership. You must specify the name of a user in the DBA group that you added in Step 4.

group-name Specifies the name of the user group of which the user is a member. You must specify the DBA group that you defined in Step 2.

Example 1–1 Creating the DBA Group and the DBA User Accounts

This example shows the sequence of commands for creating the DBA group `dba`, which is to contain the users `root` and `oracle`.

The `dba` group and the `oracle` user are created as follows:

- The GID of the `dba` group is 520.
- The home directory of the `oracle` user is `/Oracle-home`.
- The UID of the `oracle` user is 120.
- The `oracle` user's login shell is the Korn shell.

```
# echo 'dba:*:520:root,oracle' >> /etc/group
# mkdir /Oracle-home
# useradd -u 120 -g dba -d /Oracle-home -s /bin/ksh oracle
# passwd oracle
New Password:oracle
Re-enter new Password:oracle
passwd: password successfully changed for oracle
# chown oracle:dba /Oracle-home
# mkdir /var/opt/oracle
# chown oracle:dba /var/opt/oracle
```

See Also The following man pages:

- `passwd`(1)
- `useradd`(1M)
- `group`(4)
- `passwd`(4)
- `shadow`(4)

Next Steps Go to "How to Configure Shared Memory for the Oracle RAC Software in the Global Cluster" on page 36.

▼ How to Configure Shared Memory for the Oracle RAC Software in the Global Cluster

To enable the Oracle RAC software to run correctly, you must ensure that sufficient shared memory is available on all the cluster nodes. Perform this task on each cluster node.

1 Become superuser on a cluster node.

2 Update the `/etc/system` file with the shared memory configuration information.

You must configure these parameters on the basis of the resources that are available in the cluster. However, the value of each parameter must be sufficient to enable the Oracle RAC

software to create a shared memory segment that conforms to its configuration requirements. For the minimum required value of each parameter, see your Oracle documentation.

The following example shows entries to configure in the `/etc/system` file.

```
*SHARED MEMORY/ORACLE
set shmsys:shminfo_shmmax=4294967295
```

3 **Shut down and reboot each node whose `/etc/system` file you updated in Step 2.**

Caution – Before you reboot, you must ensure that any volume manager software that is already installed is completely installed. An incomplete installation of volume manager software causes a panic. If you use VxVM, also ensure that you have installed the software and that the license for the VxVM cluster feature is valid. Otherwise, a panic occurs when you reboot. For information about how to recover from a node panic during installation, see "Node Panic During Initialization of Support for Oracle RAC" on page 166.

For detailed instructions, see "Shutting Down and Booting a Single Node in a Cluster" in *Oracle Solaris Cluster System Administration Guide*.

Next Steps Go to "Installing the Support for Oracle RAC Packages" on page 40.

▼ How to Configure Shared Memory for the Oracle RAC Software in a Zone Cluster

To configure shared memory for the Oracle RAC software in a zone cluster, perform the following steps:

1 **Perform the steps to configure the shared memory in a global cluster.**

For detailed instructions on configuring the shared memory in a global cluster, see "How to Configure Shared Memory for the Oracle RAC Software in the Global Cluster" on page 36.

2 **Perform the similar steps in each zone cluster.**

Note – These steps do not affect the actual shared memory control for the zone cluster. You perform these steps to help Oracle dbca utility enables you to set the database memory allocation. If the Oracle dbca utility is not used for RAC database creation, you can skip these steps in the zone cluster.

3 **Perform the following steps, if you want to limit the memory used for the zone cluster.**

a. **Become superuser on the global cluster node that hosts the zone cluster.**

b. Configure the `capped-memory` property attributes `physical`, `swap`, and `locked` by using the `clzonecluster` command.

```
#clzonecluster configure zcname
clzonecluster:zcname> add capped-memory
clzonecluster:cz1-2n:capped-memory> set physical=memsize
clzonecluster:cz1-2n:capped-memory> set swap=memsize
clzonecluster:cz1-2n:capped-memory> set locked=memsize
clzonecluster:cz1-2n:capped-memory> end
clzonecluster:cz1-2n>commit
```

`Physical=`*memsize*	Specifies the physical memory size.
`swap=`*memsize*	Specifies the swap memory size.
`locked=`*memsize*	Specifies the limit of the shared memory segment size that the Oracle RAC database processes can request to lock in memory.

Note – In addition to the `locked` attribute of the `capped-memory` property, you can use the `max-shm-memory` property to directly configure the limit of the shared memory segment in a zone cluster. See also the Solaris man page `zonecfg(1M)`.

c. Reboot the zone cluster.

```
#clzonecluster reboot zcname
```

Note – You can perform the step to configure the `capped-memory` property attributes as part of the zone cluster creation. If you configure the `capped-memory` property attributes as part of the zone cluster creation, the memory-related properties immediately take effect after the first zone cluster boot. See "How to Create a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

▼ How to Set the Necessary Privileges for Oracle RAC Software in a Zone Cluster

You should set the necessary privileges to enable Oracle RAC to run in a zone cluster configuration. You can use the `clzonecluster` command to include the necessary privileges in a zone cluster configuration by setting the `limitpriv` property. Perform the following steps to set the necessary privileges in a zone cluster to run Oracle RAC.

1 Become superuser on the global cluster node that hosts the zone cluster.

2 **Configure the `limitpriv` property by using the `clzonecluster` command.**

```
# clzonecluster configure zcname
clzonecluster:zcname>set limitpriv ="default,proc_priocntl,proc_clock_highres"
clzonecluster:zcname>commit
```

3 **Reboot the zone cluster.**

```
# clzonecluster reboot zcname
```

Note – You can perform the step to configure the `limitpriv` property as part of the zone cluster creation. For information on creating a zone cluster, see "How to Create a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

▼ How to Configure the Logical Hostname Resources or Virtual IP Addresses for Oracle RAC Software in a Zone Cluster

To support logical hostname resources for an Oracle RAC 9*i* configuration and Oracle Clusterware virtual IP resources in Oracle RAC 10*g* and 11*g* configurations in zone clusters, you should configure the failover-capable hostnames or IP addresses used by those resources in a given zone cluster using the `clzonecluster` command.

Perform the following steps to configure the virtual IP addresses in a zone cluster configuration for Oracle RAC 10*g* or 11*g*.

1 **Become superuser on the global cluster node that hosts the zone cluster.**

2 **Configure the virtual IP addresses using the `clzonecluster` command.**

```
# clzonecluster configure zcname
clzonecluster:zcname>add net
clzonecluster:zcname:net>set address=racnode1-vip
clzonecluster:zcname:net>end
clzonecluster:zcname>add net
clzonecluster:zcname:net>set address=racnode2-vip
clzonecluster:zcname:net>end
clzonecluster:zcname>commit
```

3 **Reboot the zone cluster.**

```
# clzonecluster reboot zcname
```

Installing the Support for Oracle RAC Packages

If you did not install the Support for Oracle RAC packages during your initial Oracle Solaris Cluster installation, perform this procedure to install the packages. To install the packages, use the `installer` program.

Note – You need to install the Support for Oracle RAC packages in the global cluster and not in the zone cluster.

▼ How to Install the Support for Oracle RAC Packages

Perform this procedure on each cluster node where you are installing the Support for Oracle RAC packages.

You can run the `installer` program with a command-line interface (CLI) or with a graphical user interface (GUI). The content and sequence of instructions in the CLI and the GUI are similar.

Before You Begin Ensure that you have the Oracle Solaris Cluster installation media.

If you intend to run the `installer` program with a GUI, ensure that your `DISPLAY` environment variable is set.

1. **On the cluster node where you are installing the data service packages, become superuser.**

2. **Load the Oracle Solaris Cluster installation media into the DVD-ROM drive.**

   If the Volume Management daemon `vold(1M)` is running and configured to manage DVD-ROM devices, the daemon automatically mounts the DVD-ROM on the `/cdrom` directory.

3. **Change to the installation wizard directory of the DVD-ROM.**

   - **If you are installing the data service packages on the SPARC platform, type the following command:**

     ```
     # cd /cdrom/cdrom0/Solaris_sparc
     ```

   - **If you are installing the data service packages on the x86 platform, type the following command:**

     ```
     # cd /cdrom/cdrom0/Solaris_x86
     ```

4. **Start the installation wizard.**

   ```
   # ./installer
   ```

5. **When you are prompted, accept the license agreement.**

6. **From the list of Oracle Solaris Cluster agents under Availability Services, select the data service for Oracle RAC.**

7. **If you require support for languages other than English, select the option to install multilingual packages.**

   English language support is always installed.

8. **When prompted whether to configure the data service now or later, choose Configure Later.**

   Choose Configure Later to perform the configuration after the installation.

9. **Follow the instructions on the screen to install the data service packages on the node.**

   The installation wizard displays the status of the installation. When the installation is complete, the wizard displays an installation summary and the installation logs.

10. **(GUI only) If you do not want to register the product and receive product updates, deselect the Product Registration option.**

    The Product Registration option is not available with the CLI. If you are running the installation wizard with the CLI, omit this step.

11. **Exit the installation wizard.**

12. **Unload the installation media from the DVD-ROM drive.**

    a. **To ensure that the DVD-ROM is not being used, change to a directory that does *not* reside on the DVD-ROM.**

    b. **Eject the DVD-ROM.**

       ```
       # eject cdrom
       ```

Next Steps The next step depends on the platform that you are using, as shown in the following table.

Platform	Next Step
SPARC	"SPARC: Installing the Oracle UDLM" on page 42
x86	Chapter 2, "Configuring Storage for Oracle Files"

SPARC: Installing the Oracle UDLM

For detailed instructions for installing the Oracle UDLM, see the Oracle RAC documentation.

Caution – Before you install the Oracle UDLM, ensure that you have created the DBA group and DBA user accounts. For more information, see "How to Create the DBA Group and the DBA User Accounts" on page 33.

▼ SPARC: How to Install the Oracle UDLM

If your Oracle RAC release requires Oracle UNIX Distributed Lock Manager (Oracle UDLM), you must install the Oracle UDLM software on the local disk of each node. To support Oracle RAC on a zone cluster node, install Oracle UDLM software on each zone cluster node.

Note – On Oracle RAC 11*g* release 2 only, to use native SKGXN instead of Oracle UDLM, do not install the Oracle UDLM software. Native SKGXN is automatically used if Oracle UDLM software is not installed in the cluster.

1. **Become superuser on a cluster node.**

2. **Install the Oracle UDLM software.**

   See the appropriate Oracle RAC installation documentation for instructions.

   Note – Ensure that you did not receive any error messages when you installed the Oracle UDLM packages. If an error occurred during package installation, correct the problem before you install the Oracle UDLM software.

Next Steps Go to Chapter 2, "Configuring Storage for Oracle Files."

◆ ◆ ◆ CHAPTER 2

Configuring Storage for Oracle Files

This chapter explains how to configure storage for Oracle files.

Summary of Configuration Tasks for Storage for Oracle Files

This section summarizes the following tasks for configuring each storage management scheme for Oracle files:

Tasks for Configuring the Sun QFS Shared File System for Oracle Files

The following tables summarizes the tasks for configuring the Sun QFS shared file system and provides cross-references to detailed instructions for performing the tasks. The first table provides information on Oracle RAC running in the global cluster and the second table provide information on Oracle RAC running in a zone cluster.

Perform these tasks in the order in which they are listed in the table.

TABLE 2–1 Tasks for Configuring the Sun QFS Shared File System for Oracle Files in the Global Cluster

Task	Instructions
Install and configure the Sun QFS shared file system	"Using the Sun QFS Shared File System" on page 55
Install and configure the other storage management scheme that you are using with the Sun QFS shared file system	If you are using Solaris Volume Manager for Sun Cluster, see "Using Solaris Volume Manager for Sun Cluster" on page 51. If you are using hardware RAID support, see "Using Hardware RAID Support" on page 53.
Register and configure the RAC framework resource group	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.
If you are using Solaris Volume Manager for Sun Cluster, create a multi-owner disk set in Solaris Volume Manager for Sun Cluster for the Oracle RAC database	"How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75
Register and configure storage resources for Oracle files	If you are using the `clsetup` utility for this task, see "Registering and Configuring Storage Resources for Oracle Files" on page 82. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

TABLE 2–2 Tasks for Configuring the Sun QFS Shared File System for Oracle Files in a Zone Cluster

Task	Instructions
Install and configure the Sun QFS shared file system in the global cluster	"Using the Sun QFS Shared File System" on page 55
Install and configure the other storage management scheme that you are using with the Sun QFS shared file system in the global cluster	If you are using Solaris Volume Manager for Sun Cluster, see "Using Solaris Volume Manager for Sun Cluster" on page 51. If you are using hardware RAID support, see "Using Hardware RAID Support" on page 53.
Register and configure the RAC framework resource group in the global cluster	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.
If you are using Solaris Volume Manager for Sun Cluster, create a multi-owner disk set in Solaris Volume Manager for Sun Cluster for the Oracle RAC database in the global cluster	"How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75
Configure Sun QFS shared file system for the zone cluster	See "How to Add a QFS Shared File System to a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*
Register and configure the storage resources for Oracle files in the zone cluster	If you are using the `clsetup` utility for this task, see "Registering and Configuring Storage Resources for Oracle Files" on page 82. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files

The following tables summarize the tasks for configuring Solaris Volume Manager for Sun Cluster and provides cross-references to detailed instructions for performing the tasks.

Perform these tasks in the order in which they are listed in the table.

TABLE 2–3 Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files in the Global Cluster

Task	Instructions
Configure Solaris Volume Manager for Sun Cluster	"Using Solaris Volume Manager for Sun Cluster" on page 51
Register and configure the RAC framework resource group	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.
Create a multi-owner disk set in Solaris Volume Manager for Sun Cluster for the Oracle RAC database	"How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75
Register and configure storage resources for Oracle files	If you are using the `clsetup` utility for this task, see "Registering and Configuring Storage Resources for Oracle Files" on page 82. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

TABLE 2–4 Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files in a Zone Cluster

Task	Instructions
Configure Solaris Volume Manager for Sun Cluster in the global cluster	"Using Solaris Volume Manager for Sun Cluster" on page 51
Register and configure the RAC framework resource group in the global cluster	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.

TABLE 2–4 Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files in a Zone Cluster *(Continued)*

Task	Instructions
Create a multi-owner disk set in Solaris Volume Manager for Sun Cluster for the Oracle RAC database in the global cluster	"How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75
Configure Solaris Volume Manager devices in a zone cluster	See "How to Add a Disk Set to a Zone Cluster (Solaris Volume Manager)" in *Oracle Solaris Cluster Software Installation Guide*
Register and configure storage resources for Oracle files in the zone cluster	If you are using the `clsetup` utility for this task, see "Registering and Configuring Storage Resources for Oracle Files" on page 82. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

Tasks for Configuring VxVM for Oracle Files

The following table summarizes the tasks for configuring VxVM and provides cross-references to detailed instructions for performing the tasks.

Perform these tasks in the order in which they are listed in the table.

TABLE 2–5 Tasks for Configuring VxVM for Oracle Files

Task	Instructions
Install and configure VxVM	"Using VxVM" on page 52
Register and configure the RAC framework resource group	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.
Create a VxVM shared-disk group for the Oracle RAC database	"How to Create a VxVM Shared-Disk Group for the Oracle RAC Database" on page 81
Register and configure storage resources for Oracle files	If you are using the `clsetup` utility for this task, see "Registering and Configuring Storage Resources for Oracle Files" on page 82. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

Note – VxVM devices are currently not supported by zone clusters.

Tasks for Configuring Hardware RAID Support for Oracle Files

The following table summarizes the tasks for configuring hardware RAID support and provides cross-references to detailed instructions for performing the tasks.

TABLE 2–6 Tasks for Configuring Hardware RAID Support for Oracle Files

Task	Instructions
Configure hardware RAID support	"Using Hardware RAID Support" on page 53

Note – For information configuring hardware RAID for a zone cluster, see "Adding Storage Devices to a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

Tasks for Configuring ASM for Oracle Files

The following table summarizes the tasks for configuring ASM and provides cross-references to detailed instructions for performing the tasks.

TABLE 2–7 Tasks for Configuring ASM for Oracle Files

Task	Instructions
Configure devices for ASM	"Using Oracle ASM" on page 60

Note – For information about configuring ASM for a zone cluster, see "Adding Storage Devices to a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

Tasks for Configuring Qualified NAS Devices for Oracle Files

The following table summarizes the tasks for configuring qualified NAS devices and provides cross-references to detailed instructions for performing the tasks. NAS devices are supported in both global and zone clusters.

Perform these tasks in the order in which they are listed in the table.

TABLE 2–8 Tasks for Configuring Qualified NAS Devices for Oracle Files

Task	Instructions
Install and configure the qualified NAS device	If you are using the Oracle Solaris Cluster maintenance commands for this task, see the*Oracle Solaris Cluster 3.3 With Network-Attached Storage Devices Manual*. If you are using the Oracle Solaris Cluster Manager for this task, see the online help.
Register and configure the RAC framework resource group in a global cluster or zone cluster	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.
Register and configure storage resources for Oracle files, including Oracle RAC to support NAS NFS	If you are using the `clsetup` utility for this task, see "Registering and Configuring Storage Resources for Oracle Files" on page 82. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

Tasks for Configuring a Cluster File System for Oracle Files

The following table summarizes the tasks for configuring the cluster file system and provides cross-references to detailed instructions for performing the tasks.

Perform these tasks in the order in which they are listed in the table.

TABLE 2–9 Tasks for Configuring a Cluster File System for Oracle Files

Task	Instructions
Install and configure the cluster file system	"Using a Cluster File System" on page 62
Register and configure the RAC framework resource group	If you are using the `clsetup` utility for this task, see "Registering and Configuring the RAC Framework Resource Group" on page 65. If you are using the Oracle Solaris Cluster maintenance commands for this task, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.

Note – A cluster file system is currently not supported for Oracle RAC in zone clusters.

Installing Storage Management Software With Support for Oracle RAC

Install the software for the storage management schemes that you are using for Oracle files. For more information, see "Storage Management Requirements for Oracle Files" on page 26.

Note – For information about how to install and configure qualified NAS devices with Support for Oracle RAC, see *Oracle Solaris Cluster 3.3 With Network-Attached Storage Devices Manual*.

This section contains the following information:

Using Solaris Volume Manager for Sun Cluster

Solaris Volume Manager for Sun Cluster is always installed in the global cluster, even when supporting zone clusters. The `clzc` command configures Solaris Volume Manager for Sun

Cluster devices from the global-cluster voting node into the zone cluster. All administration tasks for Solaris Volume Manager for Sun Cluster are performed in the global-cluster voting node, even when the Solaris Volume Manager for Sun Cluster volume is used in a zone cluster.

When an Oracle RAC installation inside a zone cluster uses a file system that exists on top of a Solaris Volume Manager for Sun Cluster volume, you should still configure the Solaris Volume Manager for Sun Cluster volume in the global cluster. In this case, the scalable device group resource belongs to this zone cluster.

When an Oracle RAC installation inside a zone cluster runs directly on the Solaris Volume Manager for Sun Cluster volume, you must first configure the Solaris Volume Manager for Sun Cluster in the global cluster and then configure the Solaris Volume Manager for Sun Cluster volume into the zone cluster. In this case, the scalable device group belongs to this zone cluster.

For information about the types of Oracle files that you can store by using Solaris Volume Manager for Sun Cluster, see "Storage Management Requirements for Oracle Files" on page 26.

▼ How to Use Solaris Volume Manager for Sun Cluster

To use the Solaris Volume Manager for Sun Cluster software with Support for Oracle RAC, perform the following tasks. Solaris Volume Manager for Sun Cluster is installed during the installation of the Solaris Operating System.

1. **Configure the Solaris Volume Manager for Sun Cluster software on the global-cluster nodes.**

   For information about configuring Solaris Volume Manager for Sun Cluster in the global cluster, see "Configuring Solaris Volume Manager Software" in *Oracle Solaris Cluster Software Installation Guide*.

2. **If you are using a zone cluster, configure the Solaris Volume Manager for Sun Cluster volume into the zone cluster.**

   For information on configuring Solaris Volume Manager for Sun Cluster volume into a zone cluster, see "How to Add a Disk Set to a Zone Cluster (Solaris Volume Manager)" in *Oracle Solaris Cluster Software Installation Guide*.

Next Steps Ensure that all other storage management schemes that you are using for Oracle files are installed.

After all storage management schemes that you are using for Oracle files are installed, go to Chapter 3, "Registering and Configuring the Resource Groups."

Using VxVM

For information about the types of Oracle files that you can store by using VxVM, see "Storage Management Requirements for Oracle Files" on page 26.

Note – Using VxVM for Oracle RAC in zone clusters is not supported in this release.

▼ SPARC: How to Use VxVM

To use the VxVM software with Support for Oracle RAC, perform the following tasks.

1. **If you are using VxVM with the cluster feature, obtain a license for the Volume Manager cluster feature in addition to the basic VxVM license.**

   See your VxVM documentation for more information about VxVM licensing requirements.

Caution – Failure to correctly install the license for the Volume Manager cluster feature might cause a panic when you install Oracle RAC support. Before you install the Oracle RAC packages, run the `vxlicense -p` or `vxlicrep` command to ensure that you have installed a valid license for the Volume Manager cluster feature.

2. **Install and configure the VxVM software on the cluster nodes.**

   See Chapter 5, "Installing and Configuring Veritas Volume Manager," in *Oracle Solaris Cluster Software Installation Guide* and the VxVM documentation for more information.

Next Steps Ensure that all other storage management schemes that you are using for Oracle files are installed.

After all storage management schemes that you are using for Oracle files are installed, go to Chapter 3, "Registering and Configuring the Resource Groups."

Using Hardware RAID Support

For information about the types of Oracle files that you can store by using hardware RAID support, see "Storage Management Requirements for Oracle Files" on page 26.

Oracle Solaris Cluster software provides hardware RAID support for several storage devices. To use this combination, configure raw device identities (`/dev/did/rdsk*`) on top of the disk arrays' logical unit numbers (LUNs). To set up the raw devices for Oracle RAC on a cluster that uses StorEdge SE9960 disk arrays with hardware RAID, perform the following task.

▼ How to Use Hardware RAID Support

1. **Create LUNs on the disk arrays.**

   See the Oracle Solaris Cluster hardware documentation for information about how to create LUNs.

2 **After you create the LUNs, run the `format(1M)` command to partition the disk arrays' LUNs into as many slices as you need.**

The following example lists output from the `format` command.

```
# format

0. c0t2d0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
   /sbus@3,0/SUNW,fas@3,8800000/sd@2,0
1. c0t3d0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
   /sbus@3,0/SUNW,fas@3,8800000/sd@3,0
2. c1t5d0 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@1/rdriver@5,0
3. c1t5d1 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@1/rdriver@5,1
4. c2t5d0 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@2/rdriver@5,0
5. c2t5d1 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@2/rdriver@5,1
6. c3t4d2 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@3/rdriver@4,2
```

Note – To prevent a loss of disk partition information, do not start the partition at cylinder 0 for any disk slice that is used for raw data. The disk partition table is stored in cylinder 0 of the disk.

3 **Determine the raw device identity (DID) that corresponds to the LUNs that you created in Step 1.**

Use the `cldevice(1CL)` command for this purpose.

The following example lists output from the `cldevice list -v` command.

```
# cldevice list -v

DID Device     Full Device Path
----------     ----------------
d1             phys-schost-1:/dev/rdsk/c0t2d0
d2             phys-schost-1:/dev/rdsk/c0t3d0
d3             phys-schost-2:/dev/rdsk/c4t4d0
d3             phys-schost-1:/dev/rdsk/c1t5d0
d4             phys-schost-2:/dev/rdsk/c3t5d0
d4             phys-schost-1:/dev/rdsk/c2t5d0
d5             phys-schost-2:/dev/rdsk/c4t4d1
d5             phys-schost-1:/dev/rdsk/c1t5d1
d6             phys-schost-2:/dev/rdsk/c3t5d1
d6             phys-schost-1:/dev/rdsk/c2t5d1
d7             phys-schost-2:/dev/rdsk/c0t2d0
d8             phys-schost-2:/dev/rdsk/c0t3d0
```

In this example, the `cldevice` output identifies that the raw DID that corresponds to the disk arrays' shared LUNs is `d4`.

4 **Obtain the full DID device name that corresponds to the DID device that you identified in Step 3.**

The following example shows the output from the `cldevice show` for the DID device that was identified in the example in Step 3. The command is run from node `phys-schost-1`.

```
# cldevice show d4
```

```
=== DID Device Instances ===

DID Device Name:                                /dev/did/rdsk/d4
  Full Device Path:                               phys-schost-1:/dev/rdsk/c2t5d0
  Replication:                                    none
  default_fencing:                                global
```

5 **If you are using a zone cluster configure the DID devices into the zone cluster. Otherwise, proceed to Step 6.**

For information about configuring DID devices into a zone cluster, see "How to Add a DID Device to a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

6 **Create or modify a slice on each DID device to contain the disk-space allocation for the raw device.**

Use the `format`(1M) command, `fmthard`(1M) command, or `prtvtoc`(1M) for this purpose. Specify the full device path from the node where you are running the command to create or modify the slice.

For example, if you choose to use slice `s0`, you might choose to allocate 100 GB of disk space in slice `s0`.

7 **Change the ownership and permissions of the raw devices that you are using to allow access to these devices.**

To specify the raw device, append `s`*N* to the DID device name that you obtained in Step 4, where *N* is the slice number.

For example, the `cldevice` output in Step 4 identifies that the raw DID that corresponds to the disk is `/dev/did/rdsk/d4`. If you choose to use slice `s0` on these devices, specify the raw device `/dev/did/rdsk/d4s0`.

Next Steps Ensure that all other storage management schemes that you are using for Oracle files are installed.

After all storage management schemes that you are using for Oracle files are installed, go to Chapter 3, "Registering and Configuring the Resource Groups."

Using the Sun QFS Shared File System

The Sun QFS shared file system is always installed in the global-cluster voting node, even when a file system is used by a zone cluster. You configure specific Sun QFS shared file system into a specific zone cluster using the `clzc` command. The scalable mount-point resource belongs to this zone cluster. The metadata server resource, `SUNW.qfs`, belongs to the global cluster.

You must use the Sun QFS shared file system with one storage management scheme from the following list:

- Hardware RAID support
- Solaris Volume Manager for Sun Cluster

Distributing Oracle Files Among Sun QFS Shared File Systems

You can store all the files that are associated with Oracle RAC on the Sun QFS shared file system.

Distribute these files among several file systems as explained in the subsections that follow.

- "Sun QFS File Systems for RDBMS Binary Files and Related Files" on page 56
- "Sun QFS File Systems for Database Files and Related Files" on page 56

Sun QFS File Systems for RDBMS Binary Files and Related Files

For RDBMS binary files and related files, create one file system in the cluster to store the files.

The RDBMS binary files and related files are as follows:

- Oracle relational database management system (RDBMS) binary files
- Oracle configuration files (for example, `init.ora`, `tnsnames.ora`, `listener.ora`, and `sqlnet.ora`)
- System parameter file (`SPFILE`)
- Alert files (for example, `alert_`*sid*`.log`)
- Trace files (`*.trc`)
- Oracle Clusterware binary files

Sun QFS File Systems for Database Files and Related Files

For database files and related files, determine whether you require one file system for each database or multiple file systems for each database.

- For simplicity of configuration and maintenance, create one file system to store these files for all Oracle RAC instances of the database.
- To facilitate future expansion, create multiple file systems to store these files for all Oracle RAC instances of the database.

Note – If you are adding storage for an existing database, you must create additional file systems for the storage that you are adding. In this situation, distribute the database files and related files among the file systems that you will use for the database.

Each file system that you create for database files and related files must have its own metadata server. For information about the resources that are required for the metadata servers, see "Resources for the Sun QFS Metadata Server" on page 318.

The database files and related files are as follows:

- Data files
- Control files
- Online redo log files
- Archived redo log files
- Flashback log files
- Recovery files
- Oracle cluster registry (OCR) files
- Oracle Clusterware voting disk

Optimizing the Performance of the Sun QFS Shared File System

For optimum performance with Solaris Volume Manager for Sun Cluster, configure the volume manager and the file system as follows:

- Use Solaris Volume Manager for Sun Cluster to mirror the logical unit numbers (LUNs) of your disk arrays.
- If you require striping, configure the striping by using the file system's stripe option.

Mirroring the LUNs of your disk arrays involves the following operations:

- Creating RAID-0 metadevices
- Using the RAID-0 metadevices or Solaris Volume Manager soft partitions of such metadevices as Sun QFS devices

The input/output (I/O) load on your system might be heavy. In this situation, ensure that the LUN for Solaris Volume Manager metadata or hardware RAID metadata maps to a different physical disk than the LUN for data. Mapping these LUNs to different physical disks ensures that contention is minimized.

▼ How to Install and Configure the Sun QFS Shared File System

Before You Begin You might use Solaris Volume Manager metadevices as devices for the shared file systems. In this situation, ensure that the metaset and its metadevices are created and available on all nodes before configuring the shared file systems.

1. **Ensure that the Sun QFS software is installed on all nodes of the global cluster where Support for Oracle RAC is to run.**

   For information about how to install Sun QFS, see Using SAM-QFS With Sun Cluster.

2. **Ensure that each Sun QFS shared file system is correctly created for use with Support for Oracle RAC.**

   For information about how to create a Sun QFS file system, see Using SAM-QFS With Sun Cluster.

For each Sun QFS shared file system, set the correct mount options for the types of Oracle files that the file system is to store.

- For the file system that contains binary files, configuration files, alert files, and trace files, use the default mount options.
- For the file systems that contain data files, control files, online redo log files, and archived redo log files, set the mount options as follows:
  - In the `/etc/vfstab` file set the `shared` option.
  - In the `/etc/opt/SUNWsamfs/samfs.cmd` file or the `/etc/vfstab` file, set the following options:

    `fs=`*fs-name*
    `stripe=`*width*
    `mh_write`
    `qwrite`
    `forcedirectio`
    `rdlease=300` *Set this value for optimum performance.*
    `wrlease=300` *Set this value for optimum performance.*
    `aplease=300` *Set this value for optimum performance.*

    fs-name Specifies the name that uniquely identifies the file system.

    width Specifies the required stripe width for devices in the file system. The required stripe width is a multiple of the file system's disk allocation unit (DAU). *width* must be an integer that is greater than or equal to 1.

Note – Ensure that settings in the `/etc/vfstab` file do not conflict with settings in the `/etc/opt/SUNWsamfs/samfs.cmd` file. Settings in the `/etc/vfstab` file override settings in the `/etc/opt/SUNWsamfs/samfs.cmd` file.

3 Mount each Sun QFS shared file system that you are using for Oracle files.

```
# mount mount-point
```

mount-point Specifies the mount point of the file system that you are mounting.

4 If you are using a zone cluster, configure the Sun QFS shared file system into the zone cluster. Otherwise, go to Step 5.

For information about configuring Sun QFS shared file system into a zone cluster, see "How to Add a QFS Shared File System to a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

5 Change the ownership of each file system that you are using for Oracle files.

Note – If you have configured Sun QFS shared file system for a zone cluster, perform this step in that zone cluster.

Change the file-system ownership as follows:

- Owner: the database administrator (DBA) user
- Group: the DBA group

The DBA user and the DBA group are created as explained in "How to Create the DBA Group and the DBA User Accounts" on page 33.

```
# chown user-name:group-name mount-point
```

user-name	Specifies the user name of the DBA user. This user is normally named `oracle`.
group-name	Specifies the name of the DBA group. This group is normally named `dba`.
mount-point	Specifies the mount point of the file system whose ownership you are changing.

6 Grant to the owner of each file system whose ownership you changed in Step 5 read access and write access to the file system.

Note – When Sun QFS shared file system is configured for a zone cluster, you need to perform this step in that zone cluster.

```
# chmod u+rw mount-point
```

mount-point	Specifies the mount point of the file system to whose owner you are granting read access and write access.

Next Steps Ensure that all other storage management schemes that you are using for Oracle files are installed.

After all storage management schemes that you are using for Oracle files are installed, go to Chapter 3, "Registering and Configuring the Resource Groups."

Using Oracle ASM

Use Oracle ASM with one storage management scheme from the following list:

- **Hardware RAID.** For more information, see "How to Use Oracle ASM With Hardware RAID" on page 60.
- **Solaris Volume Manager for Sun Cluster.** For more information, see "How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75.
- **VxVM.** For more information, see "How to Create a VxVM Shared-Disk Group for the Oracle RAC Database" on page 81.

For information about the types of Oracle files that you can store by using Oracle ASM, see "Storage Management Requirements for Oracle Files" on page 26.

Note – When an Oracle RAC installation in a zone cluster uses Oracle ASM, you must configure all the devices needed by that Oracle RAC installation into that zone cluster by using the `clzonecluster` command. When Oracle ASM runs inside a zone cluster, the administration of Oracle ASM occurs entirely within the same zone cluster.

▼ How to Use Oracle ASM With Hardware RAID

1 On a cluster member, log in as root or become superuser.

2 Determine the identities of device identity (DID) devices that correspond to shared disks that are available in the cluster.

Use the `cldevice(1CL)` command for this purpose.

The following example shows an extract from output from the `cldevice list -v` command.

```
# cldevice list -v
DID Device          Full Device Path
----------          ----------------
...
d5                  phys-schost-3:/dev/rdsk/c3t216000C0FF084E77d0
d5                  phys-schost-1:/dev/rdsk/c5t216000C0FF084E77d0
d5                  phys-schost-2:/dev/rdsk/c4t216000C0FF084E77d0
d5                  phys-schost-4:/dev/rdsk/c2t216000C0FF084E77d0
d6                  phys-schost-3:/dev/rdsk/c4t216000C0FF284E44d0
d6                  phys-schost-1:/dev/rdsk/c6t216000C0FF284E44d0
d6                  phys-schost-2:/dev/rdsk/c5t216000C0FF284E44d0
d6                  phys-schost-4:/dev/rdsk/c3t216000C0FF284E44d0
...
```

In this example, DID devices `d5` and `d6` correspond to shared disks that are available in the cluster.

3 Obtain the full DID device name for each DID device that you are using for the Oracle ASM disk group.

The following example shows the output from the `cldevice show` for the DID devices that were identified in the example in Step 2. The command is run from node `phys-schost-1`.

```
# cldevice show d5 d6

=== DID Device Instances ===

DID Device Name:                                /dev/did/rdsk/d5
  Full Device Path:                             phys-schost-1:/dev/rdsk/c5t216000C0FF084E77d0
  Replication:                                       none
  default_fencing:                                 global

DID Device Name:                                 /dev/did/rdsk/d6
  Full Device Path:                              phys-schost-1:/dev/rdsk/c6t216000C0FF284E44d0
  Replication:                                       none
  default_fencing:                                   global
```

4 If you are using a zone cluster, configure the DID devices into the zone cluster. Otherwise, proceed to Step 5.

For information about configuring DID devices in a zone cluster, see "How to Add a DID Device to a Zone Cluster" in *Oracle Solaris Cluster Software Installation Guide*.

5 Create or modify a slice on each DID device to contain the disk-space allocation for the Oracle ASM disk group.

Use the `format`(1M) command, `fmthard`(1M) command, or `prtvtoc`(1M) for this purpose. Specify the full device path from the node where you are running the command to create or modify the slice.

For example, if you choose to use slice `s0` for the Oracle ASM disk group, you might choose to allocate 100 Gbytes of disk space in slice `s0`.

6 Prepare the raw devices that you are using for Oracle ASM.

a. Change the ownership and permissions of each raw device that you are using for Oracle ASM, to allow access by Oracle ASM to these devices.

Note – If Oracle ASM on hardware RAID is configured for a zone cluster, perform this step in that zone cluster.

To specify the raw device, append s*X* to the DID device name that you obtained in Step 3, where *X* is the slice number.

```
# chown oraasm:oinstall /dev/did/rdsk/dNsX
# chmod 660 /dev/disk/rdsk/dNsX
# ls -lhL /dev/did/rdsk/dNsX
```

```
crw-rw----  1 oraasm  oinstall  239, 128 Jun 15 04:38 /dev/did/rdsk/dNsX
```

For more information about changing the ownership and permissions of raw devices for use by Oracle ASM, see your Oracle documentation.

b. Clean out the disk headers for each raw device that you are using for Oracle ASM.

```
# dd if=/dev/zero of=/dev/did/rdsk/dNsX bs=1024k count=200
2000+0 records in
2000+0 records out
```

7 Modify the `ASM_DISKSTRING` Oracle ASM instance-initialization parameter to specify the devices that you are using for the Oracle ASM disk group.

Note – If Oracle ASM on hardware RAID is configured for a zone cluster, perform this step in that zone cluster.

For example, to use the `/dev/did/` path for the Oracle ASM disk group, add the value `/dev/did/rdsk/d*` to the `ASM_DISKSTRING` parameter. If you are modifying this parameter by editing the Oracle initialization parameter file, edit the parameter as follows:

```
ASM_DISKSTRING = '/dev/did/rdsk/*'
```

For more information, see your Oracle documentation.

Next Steps Ensure that all other storage management schemes that you are using for Oracle files are installed.

After all storage management schemes that you are using for Oracle files are installed, go to Chapter 3, "Registering and Configuring the Resource Groups."

Using a Cluster File System

For general information about how to create and mount cluster file systems, see the following documentation:

- "Planning the Global Devices, Device Groups, and Cluster File Systems" in *Oracle Solaris Cluster Software Installation Guide*
- "Creating Cluster File Systems" in *Oracle Solaris Cluster Software Installation Guide*

For information that is specific to the use of the cluster file system with Support for Oracle RAC, see the subsections that follow.

- "Types of Oracle Files That You Can Store on a Cluster File System" on page 63
- "Optimizing Performance and Availability When Using a Cluster File System" on page 63

- "How to Use a Cluster File System" on page 64

Types of Oracle Files That You Can Store on a Cluster File System

You can store only these files that are associated with Oracle RAC on the cluster file system:

- Oracle RDBMS binary files
- Oracle Clusterware binary files
- Oracle configuration files (for example, `init.ora`, `tnsnames.ora`, `listener.ora`, and `sqlnet.ora`)
- System parameter file (`SPFILE`)
- Alert files (for example, `alert_`*sid*`.log`)
- Trace files (`*.trc`)
- Archived redo log files
- Flashback log files
- Oracle cluster registry (OCR) files
- Oracle Clusterware voting disk

Note – You *must not* store data files, control files, online redo log files, or Oracle recovery files on the cluster file system.

Optimizing Performance and Availability When Using a Cluster File System

The I/O performance during the writing of archived redo log files is affected by the location of the device group for archived redo log files. For optimum performance, ensure that the primary of the device group for archived redo log files is located on the same node as the Oracle RAC database instance. This device group contains the file system that holds archived redo log files of the database instance.

To improve the availability of your cluster, consider increasing the desired number of secondary nodes for device groups. However, increasing the desired number of secondary nodes for device groups might also impair performance. To increase the desired number of secondary nodes for device groups, change the `numsecondaries` property. For more information, see "Multiported Device Groups" in *Oracle Solaris Cluster Concepts Guide*.

▼ How to Use a Cluster File System

1 **Create and mount the cluster file system.**

See "Creating Cluster File Systems" in *Oracle Solaris Cluster Software Installation Guide* for information about how to create and mount the cluster file system.

2 **If you are using the UNIX file system (UFS), ensure that you specify the correct mount options for various types of Oracle files.**

For the correct options, see the table that follows. You set these options when you add an entry to the `/etc/vfstab` file for the mount point.

File Type	Options
Oracle RDBMS binary files	`global,logging`
Oracle Clusterware binary files	`global,logging`
Oracle configuration files	`global,logging`
System parameter file (`SPFILE`)	`global,logging`
Alert files	`global,logging`
Trace files	`global,logging`
Archived redo log files	`global,logging,forcedirectio`
Flashback log files	`global,logging,forcedirectio`
OCR files	`global,logging,forcedirectio`
Oracle Clusterware voting disk	`global,logging,forcedirectio`

Next Steps Ensure that all other storage management schemes that you are using for Oracle files are installed.

After all storage management schemes that you are using for Oracle files are installed, go to Chapter 3, "Registering and Configuring the Resource Groups."

◆ ◆ ◆ CHAPTER 3

Registering and Configuring the Resource Groups

This chapter explains how to register and configure the resource groups that are used in an Oracle RAC configuration.

Registering and Configuring the RAC Framework Resource Group

Registering and configuring the RAC framework resource group enables Oracle RAC to run with Oracle Solaris Cluster software.

Note – You *must* register and configure the RAC framework resource group. Otherwise, Oracle RAC cannot run with Oracle Solaris Cluster software.

The RAC framework resource in the global-cluster voting node can support any Oracle RAC installation running in the global cluster. The RAC framework resource in a zone cluster supports the Oracle RAC installation running in that specific zone cluster. Multiple RAC framework resource groups can exist in a single Oracle Solaris Cluster configuration.

This section contains the following information about registering the RAC framework resource group:

- "How to Register and Configure the RAC Framework Resource Group by Using `clsetup`" on page 66

Tools for Registering and Configuring the RAC Framework Resource Group

Oracle Solaris Cluster software provides the following tools for registering and configuring the RAC framework resource group in the global cluster or in a zone cluster:

- **The `clsetup` utility.** For more information, see "How to Register and Configure the RAC Framework Resource Group by Using `clsetup`" on page 66.
- **Oracle Solaris Cluster Manager.** For more information, see the Oracle Solaris Cluster Manager online help.
- **Oracle Solaris Cluster maintenance commands.** For more information, see Appendix D, "Command-Line Alternatives."

The `clsetup` utility and Oracle Solaris Cluster Manager each provide a wizard for configuring resources for the RAC framework resource group. The wizards reduce the possibility of configuration errors that might result from command syntax errors or omissions. These wizards also ensure that all required resources are created and that all required dependencies between resources are set.

Note – Oracle Solaris Cluster Manager and the `clsetup` utility run only in a voting node of the global cluster.

▼ How to Register and Configure the RAC Framework Resource Group by Using `clsetup`

When you register and configure the RAC framework resource group for a cluster, the RAC framework resource group is created.

Perform this procedure during your initial setup of Support for Oracle RAC. Perform this procedure from one node only.

Before You Begin Ensure that the following prerequisites are met:

- All preinstallation tasks for Oracle RAC are completed.
- The Oracle Solaris Cluster nodes are prepared.
- The data services packages are installed.
- If used, Oracle UDLM software is installed.

Alternatively on Oracle RAC 11*g* release 2, if you intend to use native SKGXN, ensure that Oracle UDLM software is *not* installed.

Ensure that you have the following information:

- The names of the nodes where you require Support for Oracle RAC to run.

1 Become superuser on any cluster node.

2 Start the `clsetup` utility.

```
# clsetup
```

The `clsetup` main menu is displayed.

3 Choose the menu item, Data Services.

The Data Services menu is displayed.

4 Choose the menu item, Oracle Real Application Clusters.

The `clsetup` utility displays information about Support for Oracle RAC.

5 Press Return to continue.

The `clsetup` utility prompts you to select whether you are performing the initial configuration of Support for Oracle RAC or administering an existing configuration.

Note – The `clsetup` utility currently allows ongoing administration of a RAC framework running only in the global cluster. For ongoing administration of a RAC framework configured in a zone cluster, you need to instead use the Oracle Solaris Cluster maintenance commands.

---

6 Choose the menu item, Oracle RAC Create Configuration.

The `clsetup` utility prompts you to select the Oracle RAC cluster location. This location can be the global cluster or a zone cluster.

7 Type the option number for the location of the Oracle RAC cluster and press Return.

- **If you select the global cluster option, the `clsetup` utility displays the list of components of Oracle RAC to configure. Skip to Step 9.**
- **If you select the zone cluster option, the `clsetup` utility prompts you to select the required zone cluster. Proceed to Step 8.**

8 Type the option number for the required zone cluster and press Return.

The `clsetup` utility displays a list of components of Oracle RAC to configure.

9. **Choose the menu item, RAC Framework Resource Group .**

   The `clsetup` utility displays the list of prerequisites for performing this task.

10. **Verify that the prerequisites are met, and press Return.**

    The `clsetup` utility displays a list of the cluster nodes on which the Support for Oracle RAC packages are installed.

11. **Select the nodes where you require Support for Oracle RAC to run.**

    - **To accept the default selection of all listed nodes in an arbitrary order, type `a` and press Return.**
    - **To select a subset of the listed nodes, type a comma-separated or space-separated list of the option numbers for the nodes you choose and press Return.**

      Ensure that the nodes are listed in the order in which the nodes are to appear in the RAC framework resource group's node list.
    - **To select all nodes in a particular order, type a comma-separated or space-separated ordered list of the option numbers for the nodes you choose and press Return.**

      Ensure that the nodes are listed in the order in which the nodes are to appear in the RAC framework resource group's node list.

12. **If needed to end the selection of nodes, type `d` and press Return.**

    The `clsetup` utility displays a list of vendor clusterware support choices for Oracle files.

13. **Type the option number for the vendor clusterware support to use and press Return.**

    The vendor clusterware support can be Oracle UNIX Distributed Lock Manager (Oracle UDLM) based or (Oracle 11*g* release 2 only) native SKGXN. If Oracle UDLM software is installed (SPARC only), Oracle UDLM is automatically chosen.

14. **To confirm your selection of storage management schemes, type `d` and press Return.**

    The `clsetup` utility displays the names of the Oracle Solaris Cluster objects that the utility will create.

15. **If you require a different name for any Oracle Solaris Cluster objects, change the name.**

    a. **Type the option number for the object name to change and press Return.**

       The `clsetup` utility displays a screen where you can specify the new name.

    b. **At the New Value prompt, type the new name and press Return.**

    The `clsetup` utility returns you to the list of the names of the Oracle Solaris Cluster objects that the utility will create.

16 To confirm your selection of Oracle Solaris Cluster object names, type `d` and press Return.

The `clsetup` utility displays information about the Oracle Solaris Cluster configuration that the utility will create.

17 To create the configuration, type `c` and Press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

18 Press Return to continue.

The `clsetup` utility returns you to the list of options for configuring Support for Oracle RAC.

19 (Optional) Type `q` and press Return repeatedly until you quit the `clsetup` utility.

If you prefer, you can leave the `clsetup` utility running while you perform other required tasks before using the utility again. If you choose to quit `clsetup`, the utility recognizes your existing RAC framework resource group when you restart the utility.

20 Determine whether the RAC framework resource group and its resources are online.

Use the `clresourcegroup(1CL)` utility for this purpose. By default, the `clsetup` utility assigns the name `rac-framework-rg` to the RAC framework resource group.

- **In the global cluster, type the following command.**

  ```
  # clresourcegroup status rac-framework-rg
  ```

- **In a zone cluster, type the following command.**

  ```
  # clresourcegroup status -Z zcname rac-framework-rg
  ```

21 If the RAC framework resource group and its resources are *not* online, bring them online.

- **In the global cluster, type the following command.**

  ```
  # clresourcegroup online -emM rac-framework-rg
  ```

- **In a zone cluster, type the following command.**

  ```
  # clresourcegroup online -emM -Z zcname rac-framework-rg
  ```

More Information Resource Configuration

The following table lists the default resource configuration that the `clsetup` utility creates when you complete this task.

Resource Name, Resource Type, and Resource Group	Dependencies	Description
Resource type: `SUNW.rac_framework` Resource name: `rac-framework-rs` Resource group: `rac-framework-rg`	None.	RAC framework resource.
SPARC: Resource type: `SUNW.rac_udlm` Resource name: `rac-udlm-rs` Resource group: `rac-framework-rg`	Strong dependency on the RAC framework resource.	Oracle UDLM resource.

Next Steps The next step depends on the volume manager that you are using, as shown in the following table.

Volume Manager	Next Step
Solaris Volume Manager for Sun Cluster or VxVM with the cluster feature	"Registering and Configuring the Multiple-Owner Volume-Manager Framework Resource Group" on page 70
None	"Registering and Configuring Storage Resources for Oracle Files" on page 82

Registering and Configuring the Multiple-Owner Volume-Manager Framework Resource Group

Registering and configuring the multiple-owner volume-manager framework resource group enables Oracle RAC to manage multiple-owner volume-manager resources with Oracle Solaris Cluster software.

The multiple-owner volume manager framework resource in the global-cluster voting node supports any volume manager used by Oracle RAC anywhere on the machine, including the global cluster and all zone clusters.

This section contains the following information about registering the multiple-owner volume-manager framework resource group:

Tools for Registering and Configuring the Multiple-Owner Volume-Manager Framework Resource Group

Oracle Solaris Cluster software provides the following tools for registering and configuring the multiple-owner volume manager framework resource group in the global cluster or in a zone cluster:

- **The `clsetup` utility.** For more information, see "How to Register and Configure the Multiple-Owner Volume-Manager Framework Resource Group by Using `clsetup`" on page 71.
- **Oracle Solaris Cluster Manager.** For more information, see the Oracle Solaris Cluster Manager online help.
- **Oracle Solaris Cluster maintenance commands.** For more information, see Appendix D, "Command-Line Alternatives."

The `clsetup` utility and Oracle Solaris Cluster Manager each provide a wizard for configuring resources for the multiple-owner volume manager framework resource group. The wizards reduce the possibility of configuration errors that might result from command syntax errors or omissions. These wizards also ensure that all required resources are created and that all required dependencies between resources are set.

Note – Oracle Solaris Cluster Manager and the `clsetup` utility run only in a voting node of the global cluster.

▼ How to Register and Configure the Multiple-Owner Volume-Manager Framework Resource Group by Using `clsetup`

When you register and configure the multiple-owner volume manager framework resource group for a cluster, the multiple-owner volume manager framework resource group is created.

Perform this procedure during your initial setup of Support for Oracle RAC. Perform this procedure from one node only.

Before You Begin

- Ensure that all storage management software that you intend to use is installed and configured on all nodes where Oracle RAC is to run.
- Ensure that you have the list of storage management schemes that you are using for Oracle files.

1 Become superuser on any cluster node.

2 **Start the `clsetup` utility.**

```
# clsetup
```

The `clsetup` main menu is displayed.

3 **Type the option number for Data Services and press Return.**

The Data Services menu is displayed.

4 **Type the option number for Oracle Real Application Clusters and press Return.**

The `clsetup` utility displays information about Support for Oracle RAC.

5 **Press Return to continue.**

The `clsetup` utility prompts you to select whether you are performing the initial configuration of Support for Oracle RAC or administering an existing configuration.

Note – The `clsetup` utility currently allows ongoing administration of a RAC framework running only in the global cluster. For ongoing administration of a RAC framework configured in a zone cluster, you need to instead use the Oracle Solaris Cluster maintenance commands.

---

6 **Type the option number for Oracle RAC Create Configuration and press Return.**

The `clsetup` utility prompts you to select the Oracle RAC cluster location. This location can be the global cluster or a zone cluster.

7 **Type the option number for Global Cluster and press Return.**

Note – Configure a multiple-owner volume-manager framework resource group only in the global cluster.

The `clsetup` utility displays the list of components of Oracle RAC to configure.

8 **Type the option number for Multiple-Owner Volume-Manager Framework Resource Group and press Return.**

The `clsetup` utility displays an overview about this task.

9 **Press Return to continue.**

The `clsetup` utility prompts you to select the multiple-owner volume managers to use. Only installed volume managers are listed.

10 **Type the option numbers for the multiple-owner volume managers to use and press Return.**

11 **To confirm your selection of multiple-owner volume managers, type `d` and press Return.**

The `clsetup` utility displays the names of the Oracle Solaris Cluster objects that the utility will create.

12 If you require a different name for any Oracle Solaris Cluster objects, change the name.

a. Type the option number for the name to change and press Return.

The `clsetup` utility displays a screen where you can specify the new name.

b. At the New Value prompt, type the new name and press Return.

The `clsetup` utility returns you to the list of the names of the Oracle Solaris Cluster objects that the utility will create.

Note – If, after you configure the multiple-owner volume manager framework resource group, you again run this wizard to configure another volume manager, the names of the multiple-owner volume manager framework resource group and its existing resources are not available to rename.

---

13 To confirm your selection of Oracle Solaris Cluster object names, type `d` and press Return.

The `clsetup` utility displays information about the Oracle Solaris Cluster configuration that the utility will create.

14 To create the configuration, type `c` and Press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

15 Press Return to continue.

The `clsetup` utility returns you to the list of options for configuring Support for Oracle RAC.

16 (Optional) Type `q` and press Return repeatedly until you quit the `clsetup` utility.

If you prefer, you can leave the `clsetup` utility running while you perform other required tasks before using the utility again. If you choose to quit `clsetup`, the utility recognizes your existing multiple-owner volume-manager framework resource group when you restart the utility.

17 Determine whether the multiple-owner volume-manager framework resource group and its resources are online.

Use the `clresourcegroup(1CL)` utility for this purpose. By default, the `clsetup` utility assigns the name `vucmm_framework_rg` to the multiple-owner volume-manager framework resource group.

```
# clresourcegroup status vucmm_framework_rg
```

18 If the multiple-owner volume-manager framework resource group and its resources are *not* online, bring them online.

```
# clresourcegroup online vucmm_framework_rg
```

More Information

Resource Configuration

The following table lists the default resource configuration that the `clsetup` utility creates when you complete this task.

Resource Name, Resource Type, and Resource Group	Dependencies	Description
Resource type: `SUNW.vucmm_framework` Resource name: `vucmm_framework_rs` Resource group: `vucmm_framework_rg`	None.	Multiple-owner volume-manager framework resource.
Resource type: `SUNW.vucmm_svm` Resource name: `vucmm_svm_rs` Resource group: `vucmm_framework_rg`	Strong dependency on the multiple-owner volume-manager framework resource.	Solaris Volume Manager for Sun Cluster resource. Created only if Solaris Volume Manager for Sun Cluster was selected.
SPARC: Resource type: `SUNW.vucmm_cvm` Resource name: `vucmm_cvm_rs` Resource group: `vucmm_framework_rg`	Strong dependency on the multiple-owner volume-manager framework resource.	VxVM resource. Created only if VxVM was selected.

Next Steps The next step depends on the volume manager that you are using, as shown in the following table.

Volume Manager	Next Step
Solaris Volume Manager for Sun Cluster	"How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75
VxVM with the cluster feature	"How to Create a VxVM Shared-Disk Group for the Oracle RAC Database" on page 81
None	"Registering and Configuring Storage Resources for Oracle Files" on page 82

Creating a Global Device Group for the Oracle RAC Database

If you are using a volume manager for Oracle database files, the volume manager requires a global device group for the Oracle RAC database to use.

The type of global device group to create depends on the volume manager that you are using:

- If you are using Solaris Volume Manager for Sun Cluster, create a Solaris Volume Manager for Sun Cluster multi-owner disk set. See "How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75.
- If you are using VxVM, create a VxVM shared-disk group. See "How to Create a VxVM Shared-Disk Group for the Oracle RAC Database" on page 81.

How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database

Note – Perform this task only if you are using Solaris Volume Manager for Sun Cluster.

If you are using Solaris Volume Manager for Sun Cluster, Solaris Volume Manager requires a multi-owner disk set for the Oracle RAC database, the Sun QFS shared file system, or Oracle ASM to use. For information about Solaris Volume Manager for Sun Cluster multi–owner disk sets, see "Multi-Owner Disk Set Concepts" in *Solaris Volume Manager Administration Guide*.

Before You Begin Note the following points.

- Ensure that the required Support for Oracle RAC software packages are installed on each node. For more information, see "Installing the Support for Oracle RAC Packages" on page 40.
- Unless you are using the Sun QFS shared file system, do not create any file systems in the multi-owner disk set. In configurations without the Sun QFS shared file system, only the raw data file uses this disk set.
- Disk devices that you add to the multi-owner disk set must be directly attached to all the cluster nodes.

1 Create a multi-owner disk set.

Use the `metaset(1M)` command for this purpose.

```
# metaset -s setname -M -a -h nodelist
```

`-s` *setname*	Specifies the name of the disk set that you are creating.
`-M`	Specifies that the disk set that you are creating is a multi-owner disk set.
`-a`	Specifies that the nodes that the `-h` option specifies are to be added to the disk set.

`-h` *nodelist* — Specifies a space-separated list of nodes that are to be added to the disk set. The Support for Oracle RAC software packages *must* be installed on each node in the list.

2 Add global devices to the disk set that you created in Step 1.

```
# metaset -s setname -a devicelist
```

`-s` *setname* — Specifies that you are modifying the disk set that you created in Step 1.

`-a` — Specifies that the devices that *devicelist* specifies are to be added to the disk set.

devicelist — Specifies a space-separated list of full device ID path names for the global devices that are to be added to the disk set. To enable consistent access to each device from any node in the cluster, ensure that each device ID path name is of the form `/dev/did/dsk/d`*N*, where *N* is the device number.

3 For the disk set that you created in Step 1, create the volumes that the Oracle RAC database or Sun QFS shared file system will use.

Tip – If you are creating many volumes for Oracle data files, you can simplify this step by using soft partitions. However, if you are using the Sun QFS shared file system and the I/O load on your system is heavy, use separate partitions for data and metadata. Otherwise, the performance of your system might be impaired. For information about soft partitions, see Chapter 12, "Soft Partitions (Overview)," in *Solaris Volume Manager Administration Guide* and Chapter 13, "Soft Partitions (Tasks)," in *Solaris Volume Manager Administration Guide*.

Create each volume by concatenating slices on global devices that you added in Step 2. Use the `metainit(1M)` command for this purpose.

```
# metainit -s setname volume-abbrev numstripes width slicelist
```

`-s` *setname* — Specifies that you are creating a volume for the disk set that you created in Step 1.

volume-abbrev — Specifies the abbreviated name of the volume that you are creating. An abbreviated volume name has the format `d`*V*, where *V* is the volume number.

numstripes — Specifies the number of stripes in the volume.

width — Specifies the number of slices in each stripe. If you set *width* to greater than 1, the slices are striped.

slicelist — Specifies a space-separated list of slices that the volume contains. Each slice must reside on a global device that you added in Step 2.

4 If you are using mirrored devices, create the mirrors by using volumes that you created in Step 3 as submirrors.

If you are not using mirrored devices, omit this step.

Use the `metainit` command to create each mirror as follows:

```
# metainit -s setname mirror -m submirror-list
```

`-s` *setname* — Specifies that you are creating a mirror for the disk set that you created in Step 1.

mirror — Specifies the name of the mirror that you are creating in the form of an abbreviated volume name. An abbreviated volume name has the format `d`*V*, where *V* is the volume number.

submirror-list — Specifies a space-separated list of submirrors that the mirror is to contain. Each submirror must be a volume that you created in Step 3. Specify the name of each submirror in the form of an abbreviated volume name.

Note – For information on configuring a Solaris Volume Manager disk set in a zone cluster, see "How to Add a Disk Set to a Zone Cluster (Solaris Volume Manager)" in *Oracle Solaris Cluster Software Installation Guide*.

---

5 Verify that each node is correctly added to the multi-owner disk set.

Use the `metaset` command for this purpose.

```
# metaset -s setname
```

`-s` *setname* — Specifies that you are verifying the disk set that you created in Step 1.

This command displays a table that contains the following information for each node that is correctly added to the disk set:

- The `Host` column contains the node name.
- The `Owner` column contains the text `multi-owner`.
- The `Member` column contains the text `Yes`.

6 Verify that the multi-owner disk set is correctly configured.

```
# cldevicegroup show setname
```

setname — Specifies that configuration information only for the disk set that you created in Step 1 is displayed.

This command displays the device group information for the disk set. For a multi-owner disk set, the device group type is `Multi-owner_SVM`.

7 Verify the online status of the multi-owner disk set.

```
# cldevicegroup status setname
```

This command displays the status of the multi-owner disk set on each node in the multi-owner disk set.

8 (Configurations *without* the Sun QFS shared file system only) On each node that can own the disk set, change the ownership of each volume that you created in Step 3.

If you are using the Sun QFS shared file system, omit this step.

Note – For a zone cluster, perform this step in the zone cluster.

Change the volume ownership as follows:

- Owner: the DBA user
- Group: the DBA group

The DBA user and the DBA group are created as explained in "How to Create the DBA Group and the DBA User Accounts" on page 33.

Ensure that you change ownership only of volumes that the Oracle RAC database will use.

```
# chown user-name:group-name volume-list
```

user-name Specifies the user name of the DBA user. This user is normally named `oracle`.

group-name Specifies the name of the DBA group. This group is normally named `dba`.

volume-list Specifies a space-separated list of the logical names of the volumes that you created for the disk set. The format of these names depends on the type of device where the volume resides, as follows:

- For block devices: `/dev/md/`*setname*`/dsk/d`*V*
- For raw devices: `/dev/md/`*setname*`/rdsk/d`*V*

The replaceable items in these names are as follows:

setname Specifies the name of the multi-owner disk set that you created in Step 1.

V Specifies the volume number of a volume that you created in Step 3.

Ensure that this list specifies each volume that you created in Step 3.

9 (Configurations *without* the Sun QFS shared file system only) Grant read access and write access to the owner of each volume whose ownership you changed in Step 8 .

If you are using the Sun QFS shared file system, omit this step.

Note – For a zone cluster, perform this step in the zone cluster.

Grant access to the volume on each node that can own the disk set. Ensure that you change access permissions only of volumes that the Oracle RAC database will use.

```
# chmod u+rw volume-list
```

volume-list Specifies a space-separated list of the logical names of the volumes to whose owners you are granting read access and write access. Ensure that this list contains the volumes that you specified in Step 8.

10 If you are using Oracle ASM, specify the raw devices that you are using for the Oracle ASM disk group.

To specify the devices, modify the `ASM_DISKSTRING` Oracle ASM instance-initialization parameter.

For example, to use the `/dev/md/`*setname*`/rdsk/d` path for the Oracle ASM disk group, add the value `/dev/md/*/rdsk/d*` to the `ASM_DISKSTRING` parameter. If you are modifying this parameter by editing the Oracle initialization parameter file, edit the parameter as follows:

```
ASM_DISKSTRING = '/dev/md/*/rdsk/d*'
```

If you are using mirrored devices, specify external redundancy in the Oracle ASM configuration.

For more information, see your Oracle documentation.

Example 3–1 Creating a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster

This example shows the sequence of operations that is required to create a multi-owner disk set in Solaris Volume Manager for Sun Cluster for a four-node cluster. The disk set uses mirrored devices.

The disk set is to be used with the Sun QFS shared file system. This example does not show the creation of the Sun QFS shared file system on the devices that are added to the disk set.

1. To create the multi-owner disk set, the following command is run:

   ```
   # metaset -s oradg -M -a -h  pclus1 pclus2 pclus3 pclus4
   ```

   The multi-owner disk set is named `oradg`. The nodes `pclus1`, `pclus2`, `pclus3`, and `pclus4` are added to this disk set.

2. To add global devices to the disk set, the following command is run:

   ```
   # metaset -s oradg -a  /dev/did/dsk/d8  /dev/did/dsk/d9 /dev/did/dsk/d15 \
   /dev/did/dsk/d16
   ```

   The preceding command adds the following global devices to the disk set:

   - `/dev/did/dsk/d8`
   - `/dev/did/dsk/d9`

- /dev/did/dsk/d15
- /dev/did/dsk/d16

3. To create volumes for the disk set, the following commands are run:

```
# metainit -s oradg d10 1 1 /dev/did/dsk/d9s0
# metainit -s oradg d11 1 1 /dev/did/dsk/d16s0
# metainit -s oradg d20 1 1 /dev/did/dsk/d8s0
# metainit -s oradg d21 1 1 /dev/did/dsk/d15s0
```

Each volume is created by a one-on-one concatenation of a slice as shown in the following table. The slices are *not* striped.

Volume	Slice
d10	/dev/did/dsk/d9s0
d11	/dev/did/dsk/d16s0
d20	/dev/did/dsk/d8s0
d21	/dev/did/dsk/d15s0

4. To create mirrors for the disk set, the following commands are run:

```
# metainit -s oradg d1 -m d10 d11
# metainit -s oradg d2 -m d20 d21
```

The preceding commands create a mirror that is named d1 from volumes d10 and d11, and a mirror that is named d2 from volumes d20 and d21.

5. To verify that each node is correctly added to the multi-owner disk set, the following command is run:

```
# metaset -s oradgMulti-owner Set name = oradg, Set number = 1, Master = pclus2

Host                Owner          Member
  pclus1             multi-owner   Yes
  pclus2             multi-owner   Yes
  pclus3             multi-owner   Yes
  pclus4             multi-owner   Yes

Drive Dbase

d8    Yes

d9    Yes

d15   Yes

d16   Yes
```

6. To verify that the multi-owner disk set is correctly configured, the following command is run:

```
# cldevicegroup show oradg
=== Device Groups ===

Device Group Name:                                oradg
  Type:                                              Multi-owner_SVM
  failback:                                          false
  Node List:                                         pclus1, pclus2, pclus3, pclus4
  preferenced:                                       false
  numsecondaries:                                    0
  diskset name:                                      oradg
```

7. To verify the online status of the multi-owner disk set, the following command is run:

```
# cldevicegroup status oradg

=== Cluster Device Groups ===

--- Device Group Status ---

Device Group Name     Primary     Secondary     Status
-----------------     -------     ---------     ------


--- Multi-owner Device Group Status ---

Device Group Name           Node Name           Status
-----------------           ---------           ------
oradg                       pclus1              Online
                            pclus2              Online
                            pclus3              Online
                            pclus4              Online
```

Next Steps Go to "Registering and Configuring Storage Resources for Oracle Files" on page 82.

▼ How to Create a VxVM Shared-Disk Group for the Oracle RAC Database

Note – Perform this task only if you are using VxVM with the cluster feature.

If you are using VxVM with the cluster feature, VxVM requires a shared-disk group for the Oracle RAC database or Oracle ASM to use.

Before You Begin Note the following points.

- Ensure that the required Support for Oracle RAC software packages are installed on each node. For more information, see "Installing the Support for Oracle RAC Packages" on page 40.
- Do not register the shared-disk group as a cluster device group with the cluster.

- Do not create any file systems in the shared-disk group because only the raw data file uses this disk group.
- Create volumes as the gen use type.
- Disks that you add to the shared-disk group must be directly attached to all the cluster nodes.
- Ensure that your VxVM license is current. If your license expires, the node panics.

1 **Use Veritas commands that are provided for creating a VxVM shared-disk group.**

For information about VxVM shared-disk groups, see your VxVM documentation.

2 **If you are using Oracle ASM, specify the raw devices that you are using for the Oracle ASM disk group.**

To specify the devices, modify the `ASM_DISKSTRING` Oracle ASM instance-initialization parameter.

For example, to use the `/dev/md/`*setname*`/rdsk/d` path for the Oracle ASM disk group, add the value `/dev/md/*/rdsk/d*` to the `ASM_DISKSTRING` parameter. If you are modifying this parameter by editing the Oracle initialization parameter file, edit the parameter as follows:

```
ASM_DISKSTRING = '/dev/md/*/rdsk/d*'
```

If you are using mirrored devices, specify external redundancy in the Oracle ASM configuration.

For more information, see your Oracle documentation.

Next Steps Go to "Registering and Configuring Storage Resources for Oracle Files" on page 82.

Registering and Configuring Storage Resources for Oracle Files

Storage resources provide fault monitoring and automatic fault recovery for global device groups and file systems.

If you are using global device groups or shared file systems for Oracle files, configure storage resources to manage the availability of the storage on which the Oracle software depends.

Configure the following types of storage resources:

- Global device groups:
  - Solaris Volume Manager for Sun Cluster multi-owner disk sets
  - VxVM shared-disk groups
- Shared file systems:

- A Sun QFS shared file system with Solaris Volume Manager for Sun Cluster multi-owner disk sets
- A Sun QFS shared file system with hardware RAID
- Hardware RAID without a volume manager
- A file system on a qualified NAS device with Oracle RAC:
  - Oracle's Sun NAS device
  - Oracle's Sun Storage 7000 Unified Storage Systems NAS device
  - Network Appliance NAS device

Note – You can configure a NAS NFS in a zone cluster with the existing tools. See "Tasks for Configuring Qualified NAS Devices for Oracle Files" on page 49.

This section contains the following information about registering and configuring storage resources for Oracle files:

Tools for Registering and Configuring Storage Resources for Oracle Files

Oracle Solaris Cluster provides the following tools for registering and configuring storage resources for Oracle files in a global cluster or a zone cluster:

- **The `clsetup`(1CL) utility.** For more information, see "How to Register and Configure Storage Resources for Oracle Files by Using `clsetup`" on page 84.
- **Oracle Solaris Cluster Manager.** For more information, see the Oracle Solaris Cluster Manager online help.
- **Oracle Solaris Cluster maintenance commands.** For more information, see "Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands" on page 317.

The `clsetup` utility and Oracle Solaris Cluster Manager each provide a wizard for configuring storage resources for Oracle files. The wizards reduce the possibility of configuration errors that might result from command syntax errors or omissions. These wizards also ensure that all required resources are created and that all required dependencies between resources are set.

▼ How to Register and Configure Storage Resources for Oracle Files by Using `clsetup`

Perform this procedure from only one node of the cluster.

Before You Begin Ensure that the following prerequisites are met:

- The RAC framework resource group is created and is online. For more information, see "Registering and Configuring the RAC Framework Resource Group" on page 65.
- The multiple-owner volume-manager framework resource group is created and is online. For more information, see "Registering and Configuring the Multiple-Owner Volume-Manager Framework Resource Group" on page 70.
- Required volumes, global device groups, and file systems are created. For more information, see the following sections:
  - "Installing Storage Management Software With Support for Oracle RAC" on page 51
  - "Creating a Global Device Group for the Oracle RAC Database" on page 74
- Required file systems are mounted.

Ensure that you have the following information:

- The name of each scalable device group that you are using for Oracle files, if any
- The mount point of each shared file system that you are using for Oracle files, if any

1 On one node of the cluster, become superuser.

2 Start the `clsetup` utility.

```
# clsetup
```

The `clsetup` main menu is displayed.

3 Choose the menu item, Data Services.

The Data Services menu is displayed.

4 Choose the menu item, Oracle Real Application Clusters.

The `clsetup` utility displays information about Support for Oracle RAC.

5 Press Return to continue.

The `clsetup` utility prompts you to select whether you are performing the initial configuration of Support for Oracle RAC or administering an existing configuration.

6 Choose the menu item, Oracle RAC Create Configuration.

The `clsetup` utility prompts you to select the Oracle RAC cluster location. This location can be the global cluster or a zone cluster.

7 **Type the option number for the location of the Oracle RAC cluster and press Return.**

- **If you select the global cluster option, the `clsetup` utility displays the list of components to configure. Skip to Step 9.**
- **If you select the zone cluster option, the `clsetup` utility prompts you to select the required zone cluster. Proceed to Step 8.**

8 **Type the option number for the required zone cluster and press Return.**

The `clsetup` utility displays the list of components of Oracle RAC to configure.

9 **Choose the menu item, Storage Resources for Oracle Files.**

The `clsetup` utility displays the list of prerequisites for performing this task.

10 **Verify that the prerequisites are met, and press Return.**

If you are prompted for resources for scalable device groups, omit this step.

11 **If you are prompted to select a Storage Management Scheme for your Oracle files, select the appropriate scheme.**

- Sun StorEdge QFS With Solaris Volume Manager for Oracle Solaris Cluster
- Sun StorEdge QFS With Hardware RAID
- NAS Device
- Hardware RAID Without a Volume Manager

12 **If no suitable resources exist, or if no resource exists for a device group that you are using, add a resource to the list.**

The `clsetup` utility displays a list of the resources for scalable device groups that are configured on the cluster. If no suitable resources exist, this list is empty.

If resources exist for all the device groups that you are using, omit this step.

For each resource that you are adding, perform the following steps:

a. **Press Return.**

The `clsetup` utility displays a list of the scalable device groups that are configured on the cluster.

b. **Type the option number for the device group to use and press Return.**

Once you select the device group, you can either select the entire disk group or choose to specify logical devices, or disks, in the disk group.

c. Choose whether you want to specify logical devices.

- **To specify logical devices, type `yes`. Proceed to Step d.**
- **To select the entire disk group, type `no`. Skip to Step e.**

d. Type a comma-separated list of numbers that corresponds to the logical devices or disks you choose or type `a` for all.

The `clsetup` utility returns you to the list of resources for scalable device groups that are configured on the cluster.

e. To confirm your selection of device groups, type `d` and press Return.

The `clsetup` utility returns you to the list of the resources for scalable device groups that are configured on the cluster. The resource that you are creating is added to the list.

13 Type the numbers that correspond to the resources that you require, if not already selected.

You can select existing resources, resources that are not yet created, or a combination of existing resources and new resources. If you select more than one existing resource, the selected resources must be in the same resource group.

14 To confirm your selection of resources for device groups, type `d` and press Return.

The `clsetup` utility displays a list of the resources for shared file-system mount points that are configured on the cluster. If no suitable resources exist, this list is empty.

15 If no suitable resources exist, or if no resource exists for a file-system mount point that you are using, add a resource to the list.

If resources exist for all the file-system mount points that you are using, omit this step.

For each resource that you are adding, perform the following steps:

a. Press Return.

The `clsetup` utility displays a list of the shared file systems that are configured on the cluster.

b. Type a comma-separated or space-separated list of numbers that correspond to the file systems that you are using for Oracle files and press Return.

c. To confirm your selection of file systems, type `d` and press Return.

The `clsetup` utility returns you to the list of the resources for file-system mount points that are configured on the cluster. The resource that you are creating is added to the list.

16 Type the option numbers for the resources that you require, if not already selected.

You can select existing resources, resources that are not yet created, or a combination of existing resources and new resources. If you select more than one existing resource, the selected resources must be in the same resource group.

17 To confirm your selection of resources for file-system mount points, type `d` and press Return.

The `clsetup` utility displays the names of the Oracle Solaris Cluster objects that the utility will create or add to your configuration.

18 If you need to modify an Oracle Solaris Cluster object that the utility will create, modify the object.

a. Type the option number for the Oracle Solaris Cluster object that you are modifying and press Return.

The `clsetup` utility displays a list of properties that are set for the object.

b. Modify each property that you are changing as follows:

i. Type the option number for the property that you are changing and press Return.

The `clsetup` utility prompts you for the new value.

ii. At the prompt, type the new value and press Return.

The `clsetup` utility returns you to the list of properties that are set for the object.

c. When you have modified all the properties that you need to change, type `d` and press Return.

The `clsetup` utility returns you to the list of the names of the Oracle Solaris Cluster objects that the utility will create or add to your configuration.

19 When you have modified all the Oracle Solaris Cluster objects that you need to change, type `d` and press Return.

The `clsetup` utility displays information about the RAC framework resource group for which storage resources will be configured.

20 To create the configuration, type `c` and press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

21 Press Return to continue.

The `clsetup` utility returns you to the list of options for configuring Support for Oracle RAC.

22 (Optional) Type `q` and press Return repeatedly until you quit the `clsetup` utility.

If you prefer, you can leave the `clsetup` utility running while you perform other required tasks before using the utility again. If you choose to quit `clsetup`, the utility recognizes your existing RAC framework resource group when you restart the utility.

23 Determine whether the resource groups that the wizard created are online.

```
# clresourcegroup status
```

24 If a resource group that the wizard created is *not* online, bring the resource group online.

For each resource group that you are bringing online, type the following command:

```
# clresourcegroup online -emM rac-storage-rg
```

rac-storage-rg Specifies the name of the resource group that you are bringing online.

More Information

Resource Configuration

The following table lists the default resource configuration that the `clsetup` utility creates when you complete this task.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: `SUNW.ScalDeviceGroup` Resource name: `scal`*dg-name*`-rs`, where *dg-name* is the name of the device group that the resource represents Resource group: `scaldg-rg`	Strong dependency on the resource in the multiple-owner volume-manager framework resource group for the volume manager that is associated with the device group: either the Solaris Volume Manager for Sun Cluster resource or the VxVM resource.	Scalable device-group resource. One resource is created for each scalable device group that you are using for Oracle files.
Resource type: `SUNW.qfs` Resource name: `qfs-`*mp-dir*`-rs`, where *mp-dir* is the mount point of the file system, with `/` replaced by – Resource group: `qfsmds-rg`	Strong dependency on the scalable `wait_zc_boot` resource and scalable device-group resource, if any. If you are using Sun QFS without a volume manager, this resource does not depend on any other resources.	Resource for the Sun QFS metadata server. One resource is created for each Sun QFS shared file system that you are using for Oracle files.
Resource type: `SUNW.ScalMountPoint` Resource name: `scal-`*mp-dir*`-rs`, where *mp-dir* is the mount point of the file system, with `/` replaced by – Resource group: `scalmnt-rg`	Strong dependency on the resource for the Sun QFS metadata server, if any. Offline-restart dependency on the scalable device-group resource, if any. If you are using a file system on a qualified NAS device without a volume manager, this resource does not depend on any other resources.	Scalable file system mount point resource. One resource is created for each shared file system that you are using for Oracle files.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: `SUNW.wait_zc_boot` Resource name: `wait-zc-rs`, where *zc* is the zone cluster name. Resource group: `scalmnt-rg`	None	Resource to ensure that the Sun QFS shared file system configured to the zone cluster is mounted only after the zone cluster is booted.

Note – For detailed information for the resource configuration for zone clusters, see the figures in Appendix A, "Sample Configurations of This Data Service."

Next Steps If you are using Oracle Automatic Storage Management (Oracle ASM), go to "Registering and Configuring the Oracle ASM Resource Group" on page 89.

Otherwise, go to Chapter 4, "Enabling Oracle RAC to Run in a Cluster."

Registering and Configuring the Oracle ASM Resource Group

Oracle Automatic Storage Management (Oracle ASM) manages the storage that used by the Oracle database. This wizard creates an Oracle ASM instance resource for the Oracle database.

This section contains the following information about registering the Oracle ASM resource group:

Tools for Registering and Configuring the Oracle ASM Resource Group

Oracle Solaris Cluster software provides the following tools for registering and configuring the Oracle ASM resource group in the global cluster or in a zone cluster:

- **The `clsetup` utility.** For more information, see "How to Register and Configure the Oracle ASM Resource Group by Using `clsetup`" on page 90.
- **Oracle Solaris Cluster Manager.** For more information, see the Oracle Solaris Cluster Manager online help.
- **Oracle Solaris Cluster maintenance commands.** For more information, see Appendix D, "Command-Line Alternatives."

The `clsetup` utility and Oracle Solaris Cluster Manager each provide a wizard for configuring resources for the Oracle ASM resource group. The wizards reduce the possibility of configuration errors that might result from command syntax errors or omissions. These wizards also ensure that all required resources are created and that all required dependencies between resources are set.

Note – Oracle Solaris Cluster Manager and the `clsetup` utility run only in a voting node of the global cluster.

▼ How to Register and Configure the Oracle ASM Resource Group by Using `clsetup`

When you register and configure the Oracle ASM resource group for a cluster, the Oracle ASM resource group is created.

Perform this procedure from one node only.

Before You Begin

- Ensure that Oracle ASM disk groups are configured. For more information, see "Using Oracle ASM" on page 60.

Ensure that you have the following information:

- The name of the Oracle ASM home directory.
- The list of Oracle ASM systems identifiers (SID).
- The name of the file-system mount point to use.
- The names of the Oracle ASM disk groups to use.

1 Become superuser on any cluster node.

2 Start the `clsetup` utility.

```
# clsetup
```

The `clsetup` main menu is displayed.

3 Type the option number for Data Services and press Return.

The Data Services menu is displayed.

4 Type the option number for Oracle Real Application Clusters and press Return.

The `clsetup` utility displays information about Support for Oracle RAC.

5 Press Return to continue.

The `clsetup` utility prompts you to select whether you are performing the initial configuration of Support for Oracle RAC or administering an existing configuration.

Note – The `clsetup` utility currently allows ongoing administration of a RAC framework running only in the global cluster. For ongoing administration of a RAC framework configured in a zone cluster, use instead the Oracle Solaris Cluster maintenance commands.

6 **Type the option number for Oracle RAC Create Configuration and press Return.**

The `clsetup` utility prompts you to select the Oracle RAC cluster location. This location can be the global cluster or a zone cluster.

7 **Type the option number for the location of the Oracle RAC cluster and press Return.**

- **If you select the global cluster option, the `clsetup` utility displays the list of components of Oracle RAC to configure. Skip to Step 9.**
- **If you select the zone cluster option, the `clsetup` utility prompts you to select the required zone cluster. Proceed to Step 8.**

8 **Type the option number for the required zone cluster and press Return.**

The `clsetup` utility displays a list of components of Oracle RAC to configure.

9 **Type the option number for Automatic Storage Management (ASM) and press Return.**

The `clsetup` utility displays a list of the cluster nodes to select.

10 **Select the nodes where you require Oracle ASM to run.**

- **To accept the default selection of all listed nodes in an arbitrary order, type `a` and press Return.**
- **To select a subset of the listed nodes, type a comma-separated or space-separated list of the option numbers for the nodes you choose and press Return.**

  Ensure that the nodes are listed in the order in which the nodes are to appear in the RAC framework resource group's node list.
- **To select all nodes in a particular order, type a comma-separated or space-separated ordered list of the option numbers for the nodes you choose and press Return.**

  Ensure that the nodes are listed in the order in which the nodes are to appear in the RAC framework resource group's node list.

11 **If needed to end the selection of nodes, type `d` and press Return.**

The `clsetup` utility displays a list of Oracle ASM instance resources.

12 Type the option number that corresponds to the Oracle ASM instance resource to use.

If no Oracle ASM instance resource is available and you are prompted to create a resource, press Return. Skip to Step 14.

13 To confirm your selection of an Oracle ASM instance resource, type d and press Return.

The `clsetup` utility displays the selection screen for the Oracle ASM home directory.

14 Type the option number to choose a listed directory or to specify a directory explicitly and press Return.

The `clsetup` utility displays a list of Oracle ASM system identifiers (SID) that are discovered in the cluster.

15 Review the list of SIDs.

- **If the list is correct, type d and press Return.**
- **If the list is not correct, type the option number for the SID to change and press Return.**

The `clsetup` utility displays a list of discovered storage resources.

16 Review the list of storage resources to manage the file-system mount point where the Oracle ASM home is installed.

- **If the list is correct, type d and press Return.**
- **If no storage resource is listed, type d and press Return.**

  The `clsetup` utility will create a new resource when you complete Oracle ASM configuration.
- **If the list is not correct, type the option number for the incorrect storage resource and press Return.**

The `clsetup` utility displays a list of Oracle ASM disk-group resources that manage Oracle ASM disk groups.

17 If no suitable disk-group resources exist, or if no resource exists for an Oracle ASM disk group that you are using, add a resource to the list.

a. Type y and press Return.

The `clsetup` utility discovers Oracle ASM disk groups.

b. Type a comma-separated or space-separated list of the option numbers for the Oracle ASM disk groups to use and press Return.

c. To confirm your selection of disk groups, type d and press Return.

The `clsetup` utility returns you to the list of Oracle ASM disk-group resources. The resource that you are creating is added to the list.

18 Type the option numbers for the resources that you require, if not already selected.

You can select existing resources, resources that are not yet created, or a combination of existing resources and new resources. If you select more than one existing resource, the selected resources must be in the same resource group.

19 To confirm your selection of resources for Oracle ASM disk groups, type d and press Return.

The `clsetup` utility displays a list of discovered underlying disk sets or disk groups.

20 Type a comma-separated list of the option numbers for the global-device paths to use and press Return.

21 To confirm your selection of global-device paths, type d and press Return.

The `clsetup` utility displays the names of the Oracle Solaris Cluster objects for Oracle ASM that the utility will create or add to your configuration.

22 If you need to modify an Oracle Solaris Cluster object, modify the object.

a. Type the option number for the object to modify and press Return.

The `clsetup` utility prompts you for the new value.

b. At the prompt, type the new value and press Return.

The `clsetup` utility returns you to the list of properties that are set for the object.

23 When you have modified all the Oracle Solaris Cluster objects that you need to change, type d and press Return.

The `clsetup` utility displays information about the Oracle Solaris Cluster configuration that the utility will create.

24 To create the configuration, type c and Press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

25 Press Return to continue.

The `clsetup` utility returns you to the list of options for configuring Support for Oracle RAC.

26 (Optional) Type `q` and press Return repeatedly until you quit the `clsetup` utility.

If you prefer, you can leave the `clsetup` utility running while you perform other required tasks before using the utility again.

More Information

Resource Configuration

The following table lists the default resource configuration that the `clsetup` utility creates when you complete this task.

Resource Name, Resource Type, and Resource Group	Dependencies	Description
Resource type: `SUNW.scalable_rac_server_proxy` Resource name: `rac_server_proxy-rs` Resource group: `rac_server_proxy-rg`	Strong dependency on the Oracle RAC framework resource. Offline-restart dependency on the clustered Oracle ASM disk-group resource. Offline-restart dependency on the Oracle Clusterware framework resource. When configured with Solaris Volume Manager for Sun Cluster, strong positive affinity by the Oracle RAC instance proxy resource group for the Oracle RAC framework resource group. When configured with hardware RAID, strong positive affinity by the Oracle RAC instance proxy resource group for the clustered Oracle ASM disk-group resource group.	Oracle RAC instance proxy resource
Resource type: `SUNW.oracle_asm_diskgroup` Resource name: `asm-dg-rs` Resource group: `asm-dg-rg`	Strong positive affinity by the clustered Oracle ASM disk-group resource group for the clustered Oracle ASM instance resource group. When configured with Solaris Volume Manager for Sun Cluster: ■ Strong dependency on the clustered Oracle ASM instance resource. ■ Offline-restart dependency on the scalable device-group resource for database files. When configured with hardware RAID, offline-restart dependency on the clustered Oracle ASM disk-group resource.	Clustered Oracle ASM disk-group resource

Resource Name, Resource Type, and Resource Group	Dependencies	Description
SPARC: Resource type: `SUNW.scalable_oracle_asm_instance_proxy` Resource name: `asm-inst-rs` Resource group: `asm-inst-rg`	Offline-restart dependency on the Oracle Clusterware resource. When configured with hardware RAID, strong positive affinity by the clustered Oracle ASM instance resource group for the Oracle Clusterware framework resource group.	Clustered Oracle ASM instance resource
SPARC: Resource type: `SUNW.sqfs` Resource name: `asm-home-sqfs-rs` Resource group: `asm-home-sqfs-rg`	Strong dependency on the scalable `wait_zc_boot` resource and scalable device-group resource, if any. If you are using Sun QFS without a volume manager, this resource does not depend on any other resources.	Resource for the Oracle ASM home on a Sun QFS metadata server. One resource is created for each Sun QFS shared file system that you are using for Oracle files.
Resource type: `SUNW.ScalMountPoint` Resource name: `asm-mp-rs` Resource group: `asm-mp-rg`	Strong dependency on the resource for the Sun QFS metadata server, if any. Offline-restart dependency on the scalable device-group resource, if any. If you are using a file system on a qualified NAS device without a volume manager, this resource does not depend on any other resources.	Resource for the Oracle ASM home on a scalable file system mount point. One resource is created for each shared file system that you are using for Oracle files.
Resource type: `SUNW.ScalDeviceGroup` Resource name: `scal`*dg-name*`-rs`, where *dg-name* is the name of the device group that the resource represents Resource group: `scaldg-rg`	Strong dependency on the resource in the multiple-owner volume-manager framework resource group for the volume manager that is associated with the device group: either the Solaris Volume Manager for Sun Cluster resource or the VxVM resource.	Scalable device-group resource. One resource is created for each scalable device group that you are using for Oracle files.

Next Steps Go to Chapter 4, "Enabling Oracle RAC to Run in a Cluster."

◆ ◆ ◆ CHAPTER 4

Enabling Oracle RAC to Run in a Cluster

This chapter explains how to enable Oracle RAC to run on your Oracle Solaris Cluster nodes.

Overview of Tasks for Enabling Oracle RAC to Run in a Cluster

Table 4–1 summarizes the tasks for enabling Oracle RAC to run in a cluster.

Perform these tasks in the order in which they are listed in the table.

TABLE 4–1 Tasks for Enabling Oracle RAC to Run in a Cluster

Task	Instructions
Install the Oracle RAC software	"Installing Oracle RAC Software" on page 98
Create node-specific files and directories that the Support for Oracle RAC software requires	"Creating Node-Specific Files and Directories for a Shared File System" on page 99
Verify the installation of the Oracle RAC software	"Verifying the Installation of Oracle RAC" on page 105
Create an Oracle Oracle ASM instance	"Creating an Oracle ASM Instance and Disk Groups" on page 105
Create your Oracle database	"Creating an Oracle Database" on page 106

TABLE 4–1 Tasks for Enabling Oracle RAC to Run in a Cluster *(Continued)*

Task	Instructions
(Not required for Oracle 10*g* release 1) Configure resources for Oracle RAC database instances	"Configuring Resources for Oracle RAC Database Instances" on page 107
Verify the Support for Oracle RAC installation and configuration	"Verifying the Installation and Configuration of Support for Oracle RAC" on page 120

Installing Oracle RAC Software

This section contains the following information:

For detailed instructions for installing Oracle RAC, see your Oracle documentation.

Installing Binary Files and Configuration Files on a Shared File System

To simplify the maintenance of your Oracle installation, you can install the Oracle binary files and Oracle configuration files on a shared file system. The following shared file systems are supported:

- The Sun QFS shared file system
- The cluster file system
- A file system on a qualified NAS device

If you are installing the Oracle binary files and Oracle configuration files on a shared file system, specify the absolute paths to the file system when the Oracle installation tool requests this information. Do *not* use a symbolic link whose target is the shared file system.

Overriding Networking Defaults for Oracle 10*g* or 11*g* Oracle Clusterware

By default, the network interfaces panel of the Oracle 10*g* or 11*g* Universal Installer shows all interfaces as private. If you are installing Oracle 10*g* or 11*g* RAC for use with Oracle Solaris Cluster software, override these defaults as follows:

- Ensure that `clprivnet0` is the only private interface.

- Set the public network interfaces to public.
- Ensure that all other interfaces are not used. These interfaces represent the underlying network interfaces for the cluster interconnects.

Installing Oracle Clusterware on a Subset of Oracle Solaris Cluster Nodes

By default, the Oracle installer installs Oracle Clusterware on all nodes in a cluster. Instructions for installing Oracle Clusterware on a subset of Oracle Solaris Cluster nodes are available at the Oracle MetaLink web site (`http://metalink.oracle.com/`). See Oracle MetaLink note 280589.1 *How to install Oracle 10g CRS on a cluster where one or more nodes are not to be configured to run CRS*.

Next Steps

The next steps depend on the location of your Oracle binary files and Oracle configuration files. See the following table.

File Location	Next Step
Shared file system	"Creating Node-Specific Files and Directories for a Shared File System" on page 99
Local disks of each node	"Verifying the Installation of Oracle RAC" on page 105

Creating Node-Specific Files and Directories for a Shared File System

When Oracle software is installed on a shared file system, all the files in the directory that the `ORACLE_HOME` environment variable specifies are accessible by all cluster nodes. However, some Oracle files and directories must maintain node-specific information.

If you install Oracle software on a shared file system, you must create local copies of files and directories that must maintain node-specific information. To ensure that these files and directories are accessible by all cluster nodes, use a symbolic link whose target is a file or a directory on a file system that is local to a node. Such a file system is not part of the shared file system.

To use a symbolic link for this purpose, you must allocate an area on a local file system. To enable Oracle applications to create symbolic links to files in this area, the applications must be

able to access files in this area. Because the symbolic links reside on the shared file system, all references to the links from all nodes are the same. Therefore, all nodes must have the same namespace for the area on the local file system.

The administrator creates the file system on a global-cluster voting node, even when a zone cluster uses the file system. The administrator configures the node-specific file system into the zone cluster node using the `zonecfg` command.

This section contains the following procedures:

- "How to Create a Node-Specific Directory for a Shared File System" on page 100
- "How to Create a Node-Specific File for a Shared File System" on page 103

How to Create a Node-Specific Directory for a Shared File System

Perform this procedure for each directory that is to maintain node-specific information. The following directories are typically required to maintain node-specific information:

- `$ORACLE_HOME/rdbms/audit`
- `$ORACLE_HOME/rdbms/log`
- `$ORACLE_HOME/network/agent`
- `$ORACLE_HOME/network/log`
- `$ORACLE_HOME/network/trace`
- `$ORACLE_HOME/srvm/log`
- `$ORACLE_HOME/apache`

For information about other directories that might be required to maintain node-specific information, see your Oracle documentation.

1 On each cluster node, create the local directory that is to maintain node-specific information.

Ensure that the local directory structure that you create matches the global directory structure that contains the node-specific information. For example, the global directory `/global/oracle/network/agent` might contain node-specific information that you require to be stored locally under the `/local` directory. In this situation, you would create a directory that is named `/local/oracle/network/agent`.

```
# mkdir -p local-dir
```

`-p`	Specifies that all nonexistent parent directories are created first.
local-dir	Specifies the full path name of the directory that you are creating.

2 On each cluster node, make a local copy of the global directory that is to maintain node-specific information.

Ensure that the local copy of the node-specific information is contained in the local directory that you created in Step 1.

```
# cp -pr global-dir local-dir-parent
```

`-p`	Specifies that the owner, group, permissions modes, modification time, access time, and access control lists are preserved.
`-r`	Specifies that the directory and all its files, including any subdirectories and their files, are copied.
global-dir	Specifies the full path of the global directory that you are copying. This directory resides on the shared file system under the directory that the `ORACLE_HOME` environment variable specifies.
local-dir-parent	Specifies the directory on the local node that is to contain the local copy. This directory is the parent directory of the directory that you created in Step 1.

3 Replace the global directory that you copied in Step 2 with a symbolic link to the local copy of the global directory.

a. From any cluster node, remove the global directory that you copied in Step 2.

```
# rm -r global-dir
```

`-r`	Specifies that the directory and all its files, including any subdirectories and their files, are removed.
global-dir	Specifies the file name and full path of the global directory that you are removing. This directory is the global directory that you copied in Step 2.

b. From any cluster node, create a symbolic link from the local copy of the directory to the global directory that you removed in Step a.

```
# ln -s local-dir global-dir
```

`-s`	Specifies that the link is a symbolic link.
local-dir	Specifies that the local directory that you created in Step 1 is the source of the link.
global-dir	Specifies that the global directory that you removed in Step a is the target of the link.

Example 4–1 Creating Node-Specific Directories

This example shows the sequence of operations that is required to create node-specific directories on a two-node cluster. This cluster is configured as follows:

- The `ORACLE_HOME` environment variable specifies the `/global/oracle` directory.
- The local file system on each node is located under the `/local` directory.

The following operations are performed on each node:

1. To create the required directories on the local file system, the following commands are run:

```
# mkdir -p /local/oracle/network/agent
# mkdir -p /local/oracle/network/log
# mkdir -p /local/oracle/network/trace
# mkdir -p /local/oracle/srvm/log
# mkdir -p /local/oracle/apache
```

2. To make local copies of the global directories that are to maintain node-specific information, the following commands are run:

```
# cp -pr $ORACLE_HOME/network/agent /local/oracle/network/.
# cp -pr $ORACLE_HOME/network/log /local/oracle/network/.
# cp -pr $ORACLE_HOME/network/trace /local/oracle/network/.
# cp -pr $ORACLE_HOME/srvm/log /local/oracle/srvm/.
# cp -pr $ORACLE_HOME/apache /local/oracle/.
```

The following operations are performed on only one node:

1. To remove the global directories, the following commands are run:

```
# rm -r $ORACLE_HOME/network/agent
# rm -r $ORACLE_HOME/network/log
# rm -r $ORACLE_HOME/network/trace
# rm -r $ORACLE_HOME/srvm/log
# rm -r $ORACLE_HOME/apache
```

2. To create symbolic links from the local directories to their corresponding global directories, the following commands are run:

```
# ln -s /local/oracle/network/agent $ORACLE_HOME/network/agent
# ln -s /local/oracle/network/log $ORACLE_HOME/network/log
# ln -s /local/oracle/network/trace $ORACLE_HOME/network/trace
# ln -s /local/oracle/srvm/log $ORACLE_HOME/srvm/log
# ln -s /local/oracle/apache $ORACLE_HOME/apache
```

How to Create a Node-Specific File for a Shared File System

Perform this procedure for each file that is to maintain node-specific information. The following files are typically required to maintain node-specific information:

- `$ORACLE_HOME/network/admin/snmp_ro.ora`
- `$ORACLE_HOME/network/admin/snmp_rw.ora`

For information about other files that might be required to maintain node-specific information, see your Oracle documentation.

1 On each cluster node, create the local directory that will contain the file that is to maintain node-specific information.

`# mkdir -p` *local-dir*

`-p` Specifies that all nonexistent parent directories are created first.

local-dir Specifies the full path name of the directory that you are creating.

2 On each cluster node, make a local copy of the global file that is to maintain node-specific information.

`# cp -p` *global-file local-dir*

`-p` Specifies that the owner, group, permissions modes, modification time, access time, and access control lists are preserved.

global-file Specifies the file name and full path of the global file that you are copying. This file was installed on the shared file system under the directory that the `ORACLE_HOME` environment variable specifies.

local-dir Specifies the directory that is to contain the local copy of the file. This directory is the directory that you created in Step 1.

3 Replace the global file that you copied in Step 2 with a symbolic link to the local copy of the file.

a. From any cluster node, remove the global file that you copied in Step 2.

`# rm` *global-file*

global-file Specifies the file name and full path of the global file that you are removing. This file is the global file that you copied in Step 2.

b. From any cluster node, create a symbolic link from the local copy of the file to the global file that you removed in Step a.

`# ln -s` *local-file global-file*

`-s` Specifies that the link is a symbolic link.

local-file Specifies that the file that you copied in Step 2 is the source of the link.

global-file Specifies that the global version of the file that you removed in Step a is the target of the link.

Example 4–2 Creating Node-Specific Files

This example shows the sequence of operations that is required to create node-specific files on a two-node cluster. This cluster is configured as follows:

- The `ORACLE_HOME` environment variable specifies the `/global/oracle` directory.
- The local file system on each node is located under the `/local` directory.

The following operations are performed on each node:

1. To create the local directory that will contain the files that are to maintain node-specific information, the following command is run:

```
# mkdir -p /local/oracle/network/admin
```

2. To make a local copy of the global files that are to maintain node-specific information, the following commands are run:

```
# cp -p $ORACLE_HOME/network/admin/snmp_ro.ora \
  /local/oracle/network/admin/.

# cp -p $ORACLE_HOME/network/admin/snmp_rw.ora \
  /local/oracle/network/admin/.
```

The following operations are performed on only one node:

1. To remove the global files, the following commands are run:

```
# rm $ORACLE_HOME/network/admin/snmp_ro.ora

# rm $ORACLE_HOME/network/admin/snmp_rw.ora
```

2. To create symbolic links from the local copies of the files to their corresponding global files, the following commands are run:

```
# ln -s /local/oracle/network/admin/snmp_ro.ora \
  $ORACLE_HOME/network/admin/snmp_rw.ora

# ln -s /local/oracle/network/admin/snmp_rw.ora \
  $ORACLE_HOME/network/admin/snmp_rw.ora
```

Next Steps Go to "Verifying the Installation of Oracle RAC" on page 105.

Verifying the Installation of Oracle RAC

After you have installed Oracle RAC, verify that the installation is correct. Perform this verification before you attempt to create your Oracle database. This verification does *not* verify that Oracle RAC database instances can be started and stopped automatically.

This section contains the following procedures:

▼ How to Verify the Installation of Oracle 10*g* or 11*g* RAC

- **Confirm that the tests that the Oracle installer runs to verify the cluster were passed.**

  If the results of these tests are no longer available for review, run the Oracle utility `cluvfy` to repeat the tests.

  For more information, see your Oracle documentation.

▼ How to Verify the Installation of Oracle 9*i* RAC

1. **Confirm that the owner, group, and mode of the `$ORACLE_HOME/bin/oracle` file are as follows:**

   - Owner: `oracle`
   - Group: `dba`
   - Mode: `-rwsr-s--x`

   ```
   # ls -l $ORACLE_HOME/bin/oracle
   ```

2. **Confirm that the binary files for the Oracle listener exist in the `$ORACLE_HOME/bin` directory.**

Next Steps Go to "Creating an Oracle Database" on page 106.

Creating an Oracle ASM Instance and Disk Groups

Oracle ASM installation consists of installing and creating an Oracle ASM instance and configuring the required Oracle ASM disk groups. An Oracle ASM disk group is a collection of disk devices to store data files that Oracle ASM instances manage as a unit. Oracle ASM instances mount disk groups to make Oracle ASM files available to database instances.

▼ How to Create an Oracle Oracle ASM Instance and Disk Groups

Before You Begin

- Ensure that the Oracle Clusterware software is already installed.
- Ensure that the RAC framework is running on all nodes where the Oracle ASM instance is to be created.
- Ensure that the file system where the Oracle ASM `$ORACLE_HOME` directory will reside is already created.

1 Become superuser on a cluster node.

2 Install and configure an Oracle ASM instance and create disk groups.

See the appropriate Oracle RAC installation documentation for instructions.

When you run Oracle Universal Installer, you have the option to configure and install Oracle ASM and create disk groups. For detailed information about installing and configuring Oracle ASM, see the Oracle documentation corresponding to the Oracle database version you are using.

Creating an Oracle Database

Perform this task to configure and create the initial Oracle database in an Oracle Solaris Cluster environment. If you create and configure additional databases, you do not need to repeat this task.

To create the database, use one command from the following list:

- The Oracle `dbca` command
- The Oracle `sqlplus` command

For detailed instructions for creating an Oracle database, see your Oracle documentation.

How to specify to the `dbca` command the location of data files on a shared file system depends on your release of Oracle.

- "How to Specify the Location of Data Files on a Shared File System for Oracle 10*g* or 11*g*" on page 107
- "How to Specify the Location of Data Files on a Shared File System for Oracle 9*i*" on page 107

▼ How to Specify the Location of Data Files on a Shared File System for Oracle 10*g* or 11*g*

1. **When `dbca` prompts you for a storage option, select Cluster File System.**
2. **When `dbca` prompts you for the location, select the option for common location.**
3. **After selecting the option for a common location, type the name of the directory on the shared file system and press Return.**

Next Steps After you have created the Oracle database, go to "Configuring Resources for Oracle RAC Database Instances" on page 107.

▼ How to Specify the Location of Data Files on a Shared File System for Oracle 9*i*

- **Specify the `-datafileDestination` *path* option of the `dbca` command.**

  path specifies the path to a directory on a shared file system where the data files are to be stored.

Next Steps After you have created the Oracle database, go to "Configuring Resources for Oracle RAC Database Instances" on page 107.

Configuring Resources for Oracle RAC Database Instances

Resources for Oracle RAC database instances enable the database instances to be administered from Oracle Solaris Cluster.

If you are using Oracle 10*g* release 2 or 11*g* RAC, these resources also enable Oracle Solaris Cluster and Oracle Clusterware to interoperate.

The software architectures of Oracle 9*i*, Oracle 10*g* release 1, and Oracle 10*g* release 2 or 11*g* are different. As a result of these differences, the resources for Oracle RAC database instances that Oracle Solaris Cluster requires depend on the version of Oracle that you are using.

Note – If you are using Oracle 10*g* release 1, omit this task. In Oracle 10*g* release 1, Oracle Clusterware and Oracle Solaris Cluster cannot interoperate. Instead, Oracle Clusterware start and shut down Oracle RAC database instances.

This section contains the following information about configuring resources for Oracle RAC database instances:

Tools for Registering and Configuring Resources for Oracle RAC Database Instances

Oracle Solaris Cluster provides the following tools for registering and configuring resources for Oracle RAC database instances in the global cluster or in a zone cluster:

- **The `clsetup`(1CL) utility.** For more information, see the following sections:
  - "How to Enable Oracle Solaris Cluster and Oracle 10*g* Release 2 or 11*g* Oracle Clusterware to Interoperate" on page 109
  - "How to Automate the Startup and Shutdown of Oracle 9*i* RAC Database Instances" on page 115
- **Oracle Solaris Cluster Manager.** For more information, see the Oracle Solaris Cluster Manager online help.
- **Oracle Solaris Cluster maintenance commands.** For more information, see the following sections:
  - "Creating Resources for Interoperation With Oracle 10*g* or 11*g* by Using Oracle Solaris Cluster Maintenance Commands" on page 327
  - "Registering and Configuring Oracle Solaris Cluster Resources for Interoperation With Oracle 9*i* by Using Oracle Solaris Cluster Maintenance Commands" on page 339

The `clsetup` utility and Oracle Solaris Cluster Manager each provide a wizard for configuring resources for Oracle RAC database instances. The wizards reduce the possibility of configuration errors that might result from command syntax errors or omissions. These wizards also ensure that all required resources are created and that all required dependencies between resources are set.

▼ How to Enable Oracle Solaris Cluster and Oracle 10*g* Release 2 or 11*g* Oracle Clusterware to Interoperate

When you perform this task, the `clsetup` utility creates the following resources for interoperation with Oracle 10*g* Release 2 or 11*g*:

- A Oracle Solaris Cluster resource to act as a proxy for the Oracle RAC database
- A Oracle Solaris Cluster resource to represent the Oracle Clusterware framework
- Oracle Clusterware resources to represent scalable device groups
- Oracle Clusterware resources to represent scalable file-system mount points

Before You Begin Ensure that the following prerequisites are met:

- The RAC framework resource group is created and is online.
- Storage resources for Oracle files are configured.
- The Oracle RAC software is installed.
- Node-specific files and directories for a shared file system are created.
- The installation of the Oracle RAC software is verified.
- The Oracle database is created.
- The Oracle database is running.

To determine whether the Oracle database is running, use the Oracle `srvctl` command as follows:

```
$ srvctl status database -d db-name
```

db-name specifies the name of the Oracle database.

Ensure that you have the following information:

- The full path to the Oracle Clusterware home directory
- The name of the Oracle RAC database for which you are configuring resources
- The full path to the Oracle home directory
- The Oracle system identifier (SID) for each instance of the database for which you are configuring resources
- The names of all Oracle Solaris Cluster storage resources for Oracle files that you are using

1 On one node of the cluster, become superuser.

2 Start the `clsetup` utility.

```
# clsetup
```

The `clsetup` main menu is displayed.

3 **Choose the menu item, Data Services.**

The Data Services menu is displayed.

4 **Choose the menu item, Oracle Real Application Clusters.**

The `clsetup` utility displays information about Support for Oracle RAC.

5 **Press Return to continue.**

The `clsetup` utility prompts you to select whether you are performing the initial configuration of Support for Oracle RAC or administering an existing configuration.

6 **Choose the menu item, Oracle RAC Create Configuration.**

The `clsetup` utility prompts you to select the Oracle RAC cluster location. This location can be the global cluster or a zone cluster.

7 **Type the option number for the location of the Oracle RAC cluster and press Return.**

- **If you select the global cluster option, the `clsetup` utility displays the list of components of Oracle RAC to configure. Skip to Step 9.**
- **If you select the zone cluster option, the `clsetup` utility prompts you to select the required zone cluster. Proceed to Step 8.**

8 **Type the option number for the required zone cluster and press Return.**

The `clsetup` utility displays the list of components of Oracle RAC to configure.

9 **Choose the menu item, Resources for Oracle Real Application Clusters Database Instances.**

The `clsetup` utility displays the list of prerequisites for performing in this task.

10 **Verify that the prerequisites are met, and press Return.**

The `clsetup` utility displays a list of versions of Oracle that you can configure by using this utility.

11 **Choose the menu item, Oracle 10*g* R2 or 11*g*.**

The `clsetup` utility displays a list of Oracle Clusterware home directories that exist on the cluster.

12 **Specify the Oracle Clusterware home directory for your installation of the Oracle Clusterware software.**

- **If the directory is listed, type the option number for the directory that you are selecting and press Return.**

- **If the directory is not listed, type the full path to the Oracle Clusterware home directory and press Return.**

The `clsetup` utility displays a list of names of Oracle RAC databases that are configured on the cluster.

13 Specify the name of the Oracle RAC database that you are configuring.

- **If the name is listed, type the number that corresponds to the name that you are selecting and press Return.**

  The `clsetup` utility displays a list of Oracle home directories that exist on the cluster.

- **If the name is not listed, specify the name explicitly:**

  a. **Type e and press Return.**

     The `clsetup` utility prompts you for the name of the Oracle RACdatabase that you are configuring.

  b. **Type the name of the Oracle RAC database that you are configuring and press Return.**

     The `clsetup` utility displays a list of Oracle home directories that exist on the cluster.

14 Specify the Oracle home directory for your installation of the Oracle RAC software.

- **If the directory is listed, type the number that corresponds to the directory that you are selecting and press Return.**

  The `clsetup` utility displays a list of Oracle system identifiers that are configured on the cluster. The utility also prompts you to specify the system identifier for the first node in the list of nodes where Oracle RAC is to run.

- **If the directory is not listed, specify the directory explicitly:**

  a. **Type e and press Return.**

     The `clsetup` utility prompts you for the Oracle home directory.

  b. **Type the full path to the Oracle home directory and press Return.**

     The `clsetup` utility displays a list of Oracle system identifiers that are configured on the cluster. The utility also prompts you to specify the system identifier for the first node in the list of nodes where Oracle RAC is to run.

15 Specify the Oracle SID of the Oracle RAC database instance for the node that you are configuring.

Perform this step for each node in the list of nodes where Oracle RAC is to run.

Note – You must specify a unique Oracle SID for each node. If you specify a duplicate SID, the `clsetup` utility warns you that the SID is a duplicate and prompts you to specify the SID again.

- **If the SID is listed, type the number that corresponds to the SID that you are selecting and press Return.**

  The response of the `clsetup` utility depends on whether you have specified the Oracle SID for all nodes where Oracle RAC is to run.

  - If you have *not* specified the Oracle SID for all nodes, the `clsetup` utility prompts you for the SID of the next node in the node list.
  - If you have specified the Oracle SID for all nodes, the `clsetup` utility displays a list of Oracle Solaris Cluster storage resources for Oracle files that are configured. For information about these resources, see "Registering and Configuring Storage Resources for Oracle Files" on page 82.

- **If the SID is not listed, specify the SID explicitly.**

  a. **Type e and press Return.**

     The `clsetup` utility prompts you for the SID.

  b. **Type the SID and press Return.**

     The response of the `clsetup` utility depends on whether you have specified the Oracle SID for all nodes where Oracle RAC is to run.

     - If you have *not* specified the Oracle SID for all nodes, the `clsetup` utility prompts you for the SID of the next node in the node list.
     - If you have specified the Oracle SID for all nodes, the `clsetup` utility displays a list of Oracle Solaris Cluster storage resources for Oracle files that are configured. For information about these resources, see "Registering and Configuring Storage Resources for Oracle Files" on page 82.

16 **Type a comma-separated or space-separated list of the numbers that correspond to the storage resources that you are using, and press Return.**

The `clsetup` utility displays the names of the Oracle Solaris Cluster objects that the utility will create.

17 **If you require a different name for any Oracle Solaris Cluster objects, change the name.**

a. **Type the option number for the name to change and press Return.**

   The `clsetup` utility displays a screen where you can specify the new name.

b. At the New Value prompt, type the new name and press Return.

The `clsetup` utility returns you to the list of the names of the Oracle Solaris Cluster objects that the utility will create.

18 To confirm your selection of Oracle Solaris Cluster object names, press Return.

The `clsetup` utility displays information about the Oracle Solaris Cluster configuration that the utility will create.

19 To create the configuration, type c and Press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

20 Press Return to continue.

The `clsetup` utility returns you to the list of options for configuring Support for Oracle RAC.

21 (Optional) Type q and press Return repeatedly until you quit the `clsetup` utility.

More Information

Resource Configuration

The following table lists the default configuration of Oracle Solaris Cluster resources that the `clsetup` utility creates when you complete this task.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: `SUNW.crs_framework` Resource name: `crs_framework-rs` Resource group: `rac-framework-rg`	Strong dependency on the RAC framework resource. Offline-restart dependencies on all scalable device-group resources for Oracle files, if any. If you are using a shared file system without a volume manager, this resource has an offline-restart dependency on all scalable file-system mount-point resources for Oracle files. If you are not using a shared file system or a volume manager, this resource has no offline-restart dependencies on any other resources.	Oracle Clusterware framework resource.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: `SUNW.scalable_rac_server_proxy` Resource name: `rac_server_proxy-rs` Resource group: `rac_server_proxy-rg`	Strong dependency on the RAC framework resource. Offline-restart dependency on the Oracle Clusterware framework resource.	Proxy resource for the Oracle RAC database server.

Note – For detailed information for the resource configuration for zone clusters, see the figures in Appendix A, "Sample Configurations of This Data Service."

The `clsetup` utility also creates an Oracle Clusterware resource for each Oracle Solaris Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend.

The name of each Oracle Clusterware resource that the `clsetup` utility creates is as follows:

`sun.`*node*`.`*sc-rs*

The replaceable items in this name are as follows:

- *node* specifies the name of the node where the Oracle Clusterware resource is to run.
- *sc-rs* specifies the name of the Oracle Solaris Cluster resource that the Oracle Clusterware resource represents. This resource is created when the task in "Registering and Configuring Storage Resources for Oracle Files" on page 82 is performed.

The Oracle Clusterware resource for the database instance on *node* depends on the Oracle Clusterware resource that the `clsetup` utility creates for that node. Oracle Clusterware resources for database instances are created during the installation and configuration of the Oracle RAC

The name of the Oracle Clusterware resource for the database instance on a node is as follows:

`ora.`*dbname*`.`*sid*`.inst`

The replaceable items in this name are as follows:

- *dbname* specifies the database name of the database instance that the Oracle Clusterware resource represents.
- *sid* specifies the Oracle SID of the database instance that the Oracle Clusterware resource represents.

Next Steps Go to "Verifying the Installation and Configuration of Support for Oracle RAC" on page 120.

▼ How to Automate the Startup and Shutdown of Oracle 9*i* RAC Database Instances

Before You Begin Ensure that the following prerequisites are met:

- The RAC framework resource group is created and is online.
- Storage resources for Oracle files are configured.
- The Oracle RAC software is installed.
- Node-specific files and directories for a shared file system are created.
- The installation of the Oracle RAC software is verified.
- The Oracle database is created.

Ensure that you have the following information:

- The full path to the Oracle home directory
- The Oracle system identifier (SID) for each instance of the database for which you are configuring resources
- The names of all Oracle Solaris Cluster storage resources for Oracle files that you are using
- The logical hostnames, or the names of the logical hostname resources, that you are using for Oracle listeners

1 On one node of the cluster, become superuser.

2 Start the `clsetup` utility.

```
# clsetup
```

The `clsetup` main menu is displayed.

3 Choose the menu item, Data Services.

The Data Services menu is displayed.

4 Choose the menu item, Oracle Real Application Clusters.

The `clsetup` utility displays information about Support for Oracle RAC.

5 Press Return to continue.

The `clsetup` utility prompts you to select whether you are performing the initial configuration of Support for Oracle RAC or administering an existing configuration.

6 Choose the menu item, Oracle RAC Create Configuration.

The `clsetup` utility prompts you to choose the global cluster or a zone cluster.

7 **Type the option number for the location of the Oracle RAC cluster and press Return.**

- **If you select the global cluster option, the `clsetup` utility displays the list of components of Oracle RAC to configure. Skip to Step 9.**
- **If you select the zone cluster option, the `clsetup` utility prompts you to select the required zone cluster. Proceed to Step 8.**

8 **Type the option number for the required zone cluster, and press Return.**

The `clsetup` utility displays the list of components of Oracle RAC to configure.

9 **Choose the menu item, Resources for Oracle Real Application Clusters Database Instances.**

The `clsetup` utility displays the list of prerequisites for performing in this task.

10 **Verify that the prerequisites are met and press Return.**

The `clsetup` utility displays a list of versions of Oracle that you can configure by using this utility.

11 **Choose the menu item, Oracle 9i.**

12 **Type the option number to configure both the listener and the server.**

The `clsetup` utility displays a list of Oracle home directories that exist on the cluster.

13 **Specify the Oracle home directory for your installation of the Oracle RAC software.**

- **If the directory is listed, type the number that corresponds to the directory that you are selecting and press Return.**

  The `clsetup` utility displays a list of Oracle system identifiers that are configured on the cluster. The utility also prompts you to specify the system identifier for the first node in the list of nodes where Oracle RAC is to run.

- **If the directory is not listed, specify the directory explicitly.**

  a. **Type e and press Return.**

  The `clsetup` utility prompts you for the Oracle home directory.

  b. **Type the full path to the Oracle home directory and press Return.**

  The `clsetup` utility displays a list of Oracle system identifiers that are configured on the cluster. The utility also prompts you to specify the system identifier for the first node in the list of nodes where Oracle RAC is to run.

14 Specify the Oracle SID of the Oracle RAC database instance for the node that you are configuring.

The wizard prompts you to perform this step for each node in the list of nodes where Oracle RAC is to run.

Note – You must specify a unique Oracle SID for each node. If you specify a duplicate SID, the `clsetup` utility warns you that the SID is a duplicate and prompts you to specify the SID again.

- **If the SID is listed, type the number that corresponds to the SID that you are selecting and press Return.**

  The `clsetup` utility displays the values of properties for the node of the Oracle Solaris Cluster resources that the utility will create.

- **If the SID is not listed, specify the SID explicitly.**

  a. **Type e and press Return.**

  The `clsetup` utility prompts you for the SID.

  b. **Type the SID and press Return.**

  The `clsetup` utility displays the values of properties for the node of the Oracle Solaris Cluster resources that the utility will create.

15 If you require a different value for any Oracle Solaris Cluster resource properties, change the property.

The wizard prompts you to perform this step for each node in the list of nodes where Oracle RAC is to run.

a. **Type the option number for the property to change and press Return.**

The `clsetup` utility displays a screen where you can specify the new value.

b. **At the New Value prompt, type the new value and press Return.**

The `clsetup` utility returns you to the values of properties for the node.

c. **To confirm your selection of Oracle Solaris Cluster resource properties, press Return.**

The response of the `clsetup` utility depends on whether you have specified the Oracle Solaris Cluster resource properties for all nodes where Oracle RAC is to run.

- If you have *not* specified Oracle Solaris Cluster resource properties for all nodes, the `clsetup` utility prompts you to specify the system identifier for the next node in the list of nodes.
- If you have specified the Oracle Solaris Cluster resource properties for all nodes, the `clsetup` utility displays a list of logical hostname resources that are configured.

16 Configure a logical hostname resource for each node where Oracle RAC is to run.

The Oracle listener uses the logical hostname resource for a node to connect to the Oracle RAC database instance on the node.

The wizard prompts you to perform this step for each node in the list of nodes where Oracle RAC is to run.

- **If the logical hostname resource is listed, type the option number for the resource to select and press Return.**

  The response of the `clsetup` utility depends on whether you have configured a logical hostname resource for all nodes where Oracle RAC is to run.

  - If you have *not* configured a resource for all nodes, the `clsetup` utility prompts you to configure a resource for the next node.
  - If you have configured a resource for all nodes, the `clsetup` utility displays a list of Oracle Solaris Cluster storage resources for Oracle files that are configured. For information about these resources, see "Registering and Configuring Storage Resources for Oracle Files" on page 82.

- **If the logical hostname resource is not listed, create the resource as follows:**

  a. **Type c and press Return.**

     The `clsetup` utility prompts you for the logical hostname that the resource is to make available.

- **Type the logical hostname and press Return.**

  The `clsetup` utility returns you to the list of logical hostname resources. An entry for the logical hostname resource that the wizard will create is added to the list.

- **Type the option number for the resource to create and press Return.**

  The response of the `clsetup` utility depends on whether you have configured a logical hostname resource for all nodes where Oracle RAC is to run.

  - If you have *not* configured a resource for all nodes, the `clsetup` utility prompts you to configure a resource for the next node.
  - If you have configured a resource for all nodes, the `clsetup` utility displays a list of Oracle Solaris Cluster storage resources for Oracle files that are configured. For information about these resources, see "Registering and Configuring Storage Resources for Oracle Files" on page 82.

17 Type a comma-separated or space-separated list of the numbers that correspond to the storage resources that you are using and press Return.

The `clsetup` utility displays the names of the Oracle Solaris Cluster objects that the utility will create.

18 If you require a different name for any Oracle Solaris Cluster objects, change the name.

a. Type the option number for the name to change and press Return.

The `clsetup` utility displays a screen where you can specify the new name.

b. At the New Value prompt, type the new name and press Return.

The `clsetup` utility returns you to the list of the names of the Oracle Solaris Cluster objects that the utility will create.

19 To confirm your selection of Oracle Solaris Cluster object names, press Return.

The `clsetup` utility displays information about the Oracle Solaris Cluster configuration that the utility will create.

20 To create the configuration, type `c` and Press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

21 Press Return to continue.

The `clsetup` utility returns you to the list of options for configuring Support for Oracle RAC.

22 (Optional) Type `q` and press Return repeatedly until you quit the `clsetup` utility.

More Information

Resource Configuration

The following table lists the default resource configuration that the `clsetup` utility creates when you complete this task.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: `SUNW.LogicalHostname` Resource name: *lh-name*, where *lh-name* is the logical hostname that you specified when you created the resource Resource group: `rac-lh`*N*`-rg`, where *N* is an integer in the range 1 to the number of nodes where Oracle RAC is to run. These integers are assigned in the order in which the resource groups are created.	None.	Logical hostname resource. One logical hostname resource is created for each node where Oracle RAC is to run.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: `SUNW.scalable_rac_listener` Resource name: `rac-listener-rs` Resource group: *ora-sid*`-rg`, where *ora-sid* is the SID on the primary node without any numbers in the SID	Strong dependency on each logical hostname resource.	Scalable RAC listener resource.
Resource type: `SUNW.scalable_rac_server` Resource name: *ora-sid*`-rs`, where *ora-sid* - is the SID on the primary node without any numbers in the SID Resource group: *ora-sid*`-rg`, where *ora-sid* is the SID on the primary node without any numbers in the SID	Strong dependency on the RAC framework resource. Weak dependency on the scalable RAC listener resource. Offline-restart dependencies on all scalable device-group resources for Oracle files, if any. Offline-restart dependencies on all scalable file-system mount-point resources for Oracle files. If you are not using a shared file system or a volume manager, this resource has no offline-restart dependencies on any other resources.	Scalable RAC server resource.

Note – For detailed information for the resource configuration for zone clusters, see the figures in Appendix A, "Sample Configurations of This Data Service."

Next Steps Go to "Verifying the Installation and Configuration of Support for Oracle RAC" on page 120.

Verifying the Installation and Configuration of Support for Oracle RAC

After you install, register, and configure Support for Oracle RAC, verify the installation and configuration. Verifying the Support for Oracle RAC installation and configuration determines if resources and resource groups for Oracle RAC behave as required.

Offline-restart dependencies between resources ensure that if the independent resource is about to be taken offline, the dependent resource is taken offline first. The dependent resource remains offline until the independent resource is restarted. The procedures in this section explain how to verify that these dependencies are correctly set. For detailed information about offline-restart dependencies, see the description of the

`resource_dependencies_offline_restart` resource property in "Resource Properties" in *Oracle Solaris Cluster Data Services Planning and Administration Guide*.

Verifying the installation and configuration of Support for Oracle RAC involves the following tasks:

1. Verifying the configuration of the RAC framework resource group and, if used, verifying the configuration of the multiple-owner volume-manager framework resource group
2. Verifying the configuration of storage resources for Oracle files
3. (Oracle 10*g* release 2, Oracle 11*g*, and Oracle 9*i* only) Verifying the configuration of resources for Oracle RAC database instances

   Note – If you are using Oracle 10*g* release 1, no resources for Oracle RAC database instances are configured.

   The task to perform depends on the version of Oracle RAC that you are using:

   - **Oracle 10*g* release 2 or 11*g***. Perform the task in "How to Verify the Configuration of Resources for Oracle 10*g* Release 2 or 11*g* RAC Database Instances" on page 124.
   - **Oracle 9*i***. Perform the task in "How to Verify the Configuration of Resources for Oracle 9*i* RAC Database Instances" on page 127.
4. Verifying the correct behavior for shutdown and booting of the cluster

How to Verify the Configuration of the RAC Framework Resource Group

The RAC framework resource group is created when the task in "Registering and Configuring the RAC Framework Resource Group" on page 65 is performed.

1 Become superuser on a cluster node or assume a role that provides `solaris.cluster.read` and `solaris.cluster.admin` RBAC authorizations.

2 Verify that the RAC framework resource group is correctly configured.

```
# clresourcegroup show rac-fmk-rg
```

rac-fmk-rg Specifies the name of the RAC framework resource group.

3 If the RAC framework resource group is not already online, bring online the RAC framework resource group.

```
# clresourcegroup online rac-fmwk-rg
```

rac-fmk-rg Specifies the name of the RAC framework resource group.

4 **Verify that the RAC framework resource group is online.**

```
# clresourcegroup status
```

How to Verify the Configuration of the Multiple-Owner Volume-Manager Framework Resource Group

Perform this procedure if you use a multiple-owner volume-manager framework resource group to contain the volume manager resources that are used by your RAC configuration.

1 **Become superuser on a cluster node or assume a role that provides `solaris.cluster.read` and `solaris.cluster.admin` RBAC authorizations.**

2 **Verify that the multiple-owner volume-manager framework resource group is correctly configured.**

```
# clresourcegroup show vucmm-fmk-rg
```

vucmm-fmk-rg Specifies the name of the multiple-owner volume-manager framework resource group.

3 **If the multiple-owner volume-manager framework resource group is not already online, bring online the multiple-owner volume-manager framework resource group.**

```
# clresourcegroup online vucmm-fmwk-rg
```

vucmm-fmk-rg Specifies the name of the multiple-owner volume-manager framework resource group.

4 **Verify that the RAC resource group is online.**

```
# clresourcegroup status
```

How to Verify the Configuration of Storage Resources for Oracle Files

Perform this task only if your configuration of storage management schemes requires storage resources for Oracle files.

1 **Become superuser on a cluster node.**

2 **Verify that all resource groups that contain storage resources for Oracle files are correctly configured.**

```
# clresourcegroup show rac-storage-rg-list
```

rac-storage-rg-list Specifies a comma-separated list of resource groups that contain storage resources for Oracle files.

3 **Take offline the resource group that contains the resources on which the RAC database resource depends.**

```
# clresourcegroup offline rac-storage-rg
```

rac-storage-rg Specifies the name of the resource group that contains the resources on which the RAC database resource depends. The resource group to take offline depends on your choice of storage management schemes for Oracle files.

- If you are using a shared file system for Oracle files, take offline the resource group that contains scalable file-system mount-point resources.
- If you are using a volume manager without a file system for Oracle files, take offline the resource group that contains scalable device-group resources.

This step might require several minutes to complete.

4 **Verify that the resource groups for Oracle RAC behave as follows:**

- The status of the resource group that you took offline in Step 3 is offline.
- The status of the RAC database resource group is offline.
- The status of the RAC framework resource group depends on the version of Oracle that you are using:
  - If you are using Oracle 10*g* release 2 or 11*g*, the status of the RAC framework resource group is pending online blocked.
  - If you are using Oracle 10*g* release 1 or Oracle 9*i*, the status of the RAC framework resource group remains online.

```
# clresourcegroup status
```

5 **(Oracle 10*g* release 2 or 11*g* only) Verify that the Oracle Clusterware resources are offline.**

To verify that the Oracle Clusterware resources are offline, consult the system messages file to find messages to indicate that Oracle Clusterware resources are being shut down.

6 **Bring online the resource group that you took offline in Step 3.**

```
# clresourcegroup online rac-storage-rg
```

rac-storage-rg Specifies the name of the resource group that you took offline in Step 3.

This step might require several minutes to complete.

7 (Oracle 10*g* release 2 or 11*g* only) On each node, restart the Oracle Clusterware.

```
# /etc/init.d/init.crs start
Startup will be queued to init within 30 seconds.
```

This step might require several minutes to complete.

8 (Oracle 10*g* release 2 or 11*g* only) On each node, verify that the Oracle Clusterware resources are online.

Use the Oracle command `crstat` for this purpose.

```
# crs-home/bin/crs_stat
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

Note – The startup of Oracle Clusterware on all nodes might require several minutes to complete. If you perform this step before the Oracle Clusterware resources are online on all nodes, the status on some nodes might be offline. In this situation, repeat this step until the status of the Oracle Clusterware is online on all nodes.

---

9 Verify that all resource groups for Oracle RAC are online.

```
# clresourcegroup status
```

Note – Bringing online the resource group that contains the resources on which the RAC database resource depends also brings online the RAC database resource group. If you perform this step before the RAC database resource group is online on all nodes, the status on some nodes might be online faulted. In this situation, repeat this step until the status of the RAC database resource group is online on all nodes.

▼ How to Verify the Configuration of Resources for Oracle 10*g* Release 2 or 11*g* RAC Database Instances

Resources for Oracle 10*g* Release 2 or 11*g* RAC database instances are created when the tasks in one of the following sections are performed:

- "How to Enable Oracle Solaris Cluster and Oracle 10*g* Release 2 or 11*g* Oracle Clusterware to Interoperate" on page 109
- "Creating Resources for Interoperation With Oracle 10*g* or 11*g* by Using Oracle Solaris Cluster Maintenance Commands" on page 327

1 Become superuser on a cluster node.

2 Verify that the RAC database resource group is correctly configured.

```
# clresourcegroup show rac-db-rg
```

rac-db-rg Specifies the name of the RAC database resource group.

3 If required, verify that the Oracle Clusterware resources to represent Oracle Solaris Cluster storage resources are correctly configured.

Perform this step only if your configuration of storage management schemes requires storage resources for Oracle files.

a. Confirm that the Oracle Clusterware resources to represent Oracle Solaris Cluster storage resources exist.

Use the Oracle command `crstat` for this purpose.

```
# crs-home/bin/crs_stat | grep NAME=sun.
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

b. For each Oracle Clusterware resource that represents an Oracle RAC database instance, confirm that the required dependencies are set.

Each Oracle Clusterware resource that represents an Oracle RAC database instance requires a dependency on a resource that is listed in Step a.

```
# crs-home/bin/crs_stat crs-resource | grep REQUIRED_RESOURCE
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

crs-resource Specifies the name of an Oracle Clusterware resource that represents an Oracle RAC database instance. The format of this name is `ora.`*dbname*`.`*sid*`.inst`. The replaceable items in this format are as follows:

dbname Specifies the database name of the instance.

sid Specifies the Oracle SID of the instance.

4 Disable the Oracle Clusterware framework resource.

```
# clresource disable -t SUNW.crs_framework +
```

This step might require several minutes to complete.

5 Verify that the Oracle Clusterware resources are offline.

a. Attempt to obtain the status of Oracle Clusterware resources.

Use the Oracle command `crstat` for this purpose.

```
# crs-home/bin/crs_stat -t
CRS-0184: Cannot communicate with the CRS daemon.
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

b. Consult the system messages file to find messages to indicate that Oracle Clusterware resources are being shut down.

6 Verify that the following resources are offline on all nodes:

- The Oracle Clusterware framework resource
- The RAC database resource

```
# clresource status -t SUNW.crs_framework,SUNW.scalable_rac_server_proxy +
```

7 On each node, restart the Oracle Clusterware.

```
# /etc/init.d/init.crs start
Startup will be queued to init within 30 seconds.
```

This step might require several minutes to complete.

8 On each node, verify that the Oracle Clusterware resources are online.

Use the Oracle command `crstat` for this purpose.

```
# crs-home/bin/crs_stat
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

Note – The startup of Oracle Clusterware on all nodes might require several minutes to complete. If you perform this step before the Oracle Clusterware resources are online on all nodes, the status on some nodes might be offline. In this situation, repeat this step until the status of the Oracle Clusterware is online on all nodes.

---

9 Enable the Oracle Clusterware framework resource.

```
# clresource enable -t SUNW.crs_framework +
```

This step might require several minutes to complete.

10 Verify that the following resources are online on all nodes:

- The Oracle Clusterware framework resource

- The RAC database resource

```
# clresource status -t SUNW.crs_framework,SUNW.scalable_rac_server_proxy +
```

Note – Enabling the Oracle Clusterware framework resource also enable the RAC database resource. If you perform this step before the RAC database resource is enabled on all nodes, the status on some nodes might be offline. In this situation, repeat this step until the status of the RAC database resource is online on all nodes.

How to Verify the Configuration of Resources for Oracle 9*i* RAC Database Instances

Resources for Oracle 9*i* RAC database instances are created when the tasks in one of the following sections are performed:

- "How to Automate the Startup and Shutdown of Oracle 9*i* RAC Database Instances" on page 115
- "Registering and Configuring Oracle Solaris Cluster Resources for Interoperation With Oracle 9*i* by Using Oracle Solaris Cluster Maintenance Commands" on page 339

1. **Become superuser on a cluster node, or assume a role that provides `solaris.cluster.read` and `solaris.cluster.admin` RBAC authorizations.**

2. **Verify that the RAC database resource group and the resource groups for logical hostname resources are correctly configured.**

   ```
   # clresourcegroup show rac-db-rg,lhrg-list
   ```

   rac-db-rg Specifies the name of the RAC database resource group.

   lhrg-list Specifies a comma-separated list of resource groups for logical hostname resources.

3. **Take offline the RAC database resource group.**

   ```
   # clresourcegroup offline rac-db-rg
   ```

   rac-db-rg Specifies the name of the RAC database resource group.

4. **Verify that the RAC database resource group is offline.**

   ```
   # clresourcegroup status rac-db-rg
   ```

   rac-db-rg Specifies the name of the RAC database resource group.

5. **Verify that the resources in the RAC database resource group are offline on all nodes.**

The RAC database resource group contains the following resources:

- The scalable RAC server resource
- The scalable RAC listener resource

```
# clresource status -t SUNW.scalable_rac_server,SUNW.scalable_rac_listener +
```

6 Bring online again the RAC database resource group.

```
# clresourcegroup online rac-db-rg
```

rac-db-rg Specifies the name of the RAC database resource group.

7 Verify that the RAC database resource group is online.

```
# clresourcegroup status rac-db-rg
```

rac-db-rg Specifies the name of the RAC database resource group.

8 Verify that the resources in the RAC database resource group are online on all nodes.

The RAC database resource group contains the following resources:

- The scalable RAC server resource
- The scalable RAC listener resource

```
# clresource status -t SUNW.scalable_rac_server,SUNW.scalable_rac_listener +
```

▼ How to Verify the Correct Behavior for Shutdown and Booting of the Cluster

If your configuration of Support for Oracle RAC is correct, Oracle Solaris Cluster ensures that Oracle RAC is stopped and started correctly when you shut down and boot the cluster.

Caution – This task requires downtime. If you are verifying the installation and configuration of Support for Oracle RAC on a production cluster where other data services are running, omit this task.

1 Shut down the cluster.

For instructions for performing this task, see "How to Shut Down a Cluster" in *Oracle Solaris Cluster System Administration Guide*.

2 Confirm that the cluster shuts down correctly.

3 **Boot the cluster.**

For instructions for performing this task, see "How to Boot a Cluster" in *Oracle Solaris Cluster System Administration Guide*.

4 **Confirm that the cluster starts correctly.**

5 **Verify that all Oracle RAC resource groups and their resources are online.**

```
# clresourcegroup status
```

Note – The bringing online of all Oracle RAC resource groups might require several minutes to complete. If you perform this step before the resource groups are enabled on all nodes, the status on some nodes might be offline. In this situation, repeat this step until the status of all Oracle RAC resource groups is online on all nodes.

◆ ◆ ◆ CHAPTER 5

Administering Support for Oracle RAC

This chapter explains how to administer Support for Oracle RAC on your Oracle Solaris Cluster nodes.

Overview of Administration Tasks for Support for Oracle RAC

Table 5–1 summarizes the administration tasks for Support for Oracle RAC.

Perform these tasks whenever they are required.

TABLE 5–1 Administration Tasks for Support for Oracle RAC

Task	Instructions
AdministerOracle RAC databases from Oracle Solaris Cluster	"Administering Oracle RAC Databases From Oracle Solaris Cluster Software" on page 133
Tune Support for Oracle RAC extension properties	"Tuning Support for Oracle RAC" on page 136
Tune Support for Oracle RAC fault monitors	"Tuning the Support for Oracle RAC Fault Monitors" on page 139
Customize the Oracle 9i RAC server fault monitor	"Customizing the Oracle 9i RAC Server Fault Monitor" on page 145
Troubleshoot Support for Oracle RAC	Chapter 6, "Troubleshooting Support for Oracle RAC"

Automatically Generated Names for Oracle Solaris Cluster Objects

When the `clsetup` utility or Oracle Solaris Cluster Manager is used to create resources, these tools assign preset names to the resources. If you are administering resources that were created by using the `clsetup` utility or Oracle Solaris Cluster Manager, see the following table for these names.

Resource Type	Resource Name
`SUNW.rac_svm`	`rac-svm-rs`
SPARC: `SUNW.rac_cvm`	`rac-cvm-rs`
SPARC: `SUNW.rac_udlm`	`rac-udlm-rs`
`SUNW.rac_framework`	`rac-framework-rs`
`SUNW.scalable_rac_server`	*ora-sid*`-rs`, where *ora-sid* is the SID on the primary node without any numbers in the SID
`SUNW.scalable_rac_listener`	`rac-listener-rs`
`SUNW.scalable_rac_server_proxy`	`rac_server_proxy-rs`
`SUNW.crs_framework`	`crs_framework-rs`
`SUNW.ScalDeviceGroup`	`scal`*dg-name*`-rs`, where *dg-name* is the name of the device group that the resource represents
`SUNW.ScalMountPoint`	`scal-`*mp-dir*`-rs`, where *mp-dir* is the mount point of the file system, with `/` replaced by `–` `asm-mp-rs`
`SUNW.qfs`	`qfs-`*mp-dir*`-rs`, where *mp-dir* is the mount point of the file system, with `/` replaced by `–` `asm-home-sqfs-rs`
`SUNW.scalable_rac_server_proxy`	`rac_server_proxy_rs`
`SUNW.oracle_asm_diskgroup`	`asm-dg-rs`
`SUNW.scalable_oracle_asm_instance_proxy`	`asm-inst-rs`
`SUNW.LogicalHostname`	*lh-name*, where *lh-name* is the logical hostname that you specified when you created the resource

Administering Oracle RAC Databases From Oracle Solaris Cluster Software

Administering Oracle RAC databases from Oracle Solaris Cluster software involves using Oracle Solaris Cluster administration tools to modify the states of Oracle Solaris Cluster resources for Oracle RAC database instances. For information about how to create these resources, see "Configuring Resources for Oracle RAC Database Instances" on page 107.

The software architectures of Oracle 9*i*, Oracle 10*g* release 1, and Oracle 10*g* release 2 or 11*g* are different. As a result of these differences, the resources for Oracle RAC database instances that Oracle Solaris Cluster software requires depend on the version of Oracle that you are using. Consequently, the administration of Oracle RAC databases from Oracle Solaris Cluster software also depends on the version of Oracle that you are using.

Note – If you are using Oracle 10*g* release 1, you cannot administer Oracle RAC databases from Oracle Solaris Cluster software. Instead, use Oracle Clusterware utilities to start and shut down Oracle RAC database instances.

Each cluster, such as the global cluster or a specific zone cluster, constitutes an independent name space for resource groups and resources. Thus, there is no name collision between names in these different name spaces. You register resource types independently for each cluster.

You can administer the resource groups and resources belonging to the global cluster only from the global-cluster voting node. You can administer the resource groups and resources belonging to a zone cluster from that zone cluster. In the global-cluster voting node, you can also administer the resource groups and resources in a zone cluster by using the `-Z` option to specify the zone cluster. You can set up inter-cluster resource dependencies or affinities only from the global-cluster voting node. The examples in the following sections typically explain the situations when the command is issued in the cluster where the resource group or resource resides.

The effects of changes to the states of Oracle Solaris Cluster resources on Oracle database components are explained in the subsections that follow:

- "Effects of State Changes to Oracle Solaris Cluster Resources for Oracle 10*g* Release 2 or 11*g* RAC Database Instances" on page 134
- "Effects of State Changes to Oracle Solaris Cluster Resources for Oracle 9*i* RAC Database Instances" on page 136

Effects of State Changes to Oracle Solaris Cluster Resources for Oracle 10*g* Release 2 or 11*g* RAC Database Instances

In Oracle 10*g* or 11*g*, Oracle Clusterware manages the startup and shutdown of Oracle database instances, listeners, and other components that are configured in the Oracle Clusterware. Oracle Clusterware is a mandatory component of Oracle 10*g* or 11*g*. Oracle Clusterware also monitors components that are started by Oracle Clusterware and, if failures are detected, perform actions to recover from failures.

Because Oracle Clusterware manages the startup and shutdown of Oracle database components, these components cannot be stopped and started exclusively under the control of the Oracle Solaris Cluster RGM. Instead, Oracle Clusterware and the RGM interoperate so that when Oracle RAC database instances are started and stopped by Oracle Clusterware, the state of the database instances is propagated to Oracle Solaris Cluster resources.

TABLE 5–2 Propagation of State Changes Between Oracle Solaris Cluster Resources and Oracle Clusterware Resources

	Initial State		Resulting State	
Trigger	**Oracle Solaris Cluster Resource**	**Oracle Clusterware Resource**	**Oracle Solaris Cluster Resource**	**Oracle Clusterware Resource**
Oracle Solaris Cluster command to take offline a resource	Enabled and online	Enabled and online	Enabled and offline	Enabled and offline
Oracle Clusterware command to stop a resource	Enabled and online	Enabled and online	Enabled and offline	Enabled and offline
Oracle Solaris Cluster command to bring online a resource	Enabled and offline	Enabled and offline	Enabled and online	Enabled and online
Oracle Clusterware command to start a resource	Enabled and offline	Enabled and offline	Enabled and online	Enabled and online
Oracle Solaris Cluster command to disable a resource	Enabled and online	Enabled and online	Disabled and offline	Disabled and offline
Oracle Clusterware command to disable a resource	Enabled and online	Enabled and online	Enabled and online	Disabled and online

TABLE 5–2 Propagation of State Changes Between Oracle Solaris Cluster Resources and Oracle Clusterware Resources *(Continued)*

	Initial State		Resulting State	
Trigger	**Oracle Solaris Cluster Resource**	**Oracle Clusterware Resource**	**Oracle Solaris Cluster Resource**	**Oracle Clusterware Resource**
Oracle `SQLPLUS` command to shut down the database	Enabled and online	Enabled and online	Enabled and offline	Enabled and offline
Oracle Solaris Cluster command to enable a resource	Disabled and offline	Disabled and offline	Enabled and online or offline	Enabled and online or offline
Oracle Clusterware command to enable a resource	Disabled and offline	Disabled and offline	Disabled and offline	Enabled and offline

The names of the states of Oracle Solaris Cluster resources and Oracle Clusterware resources are identical. However, the meaning of each state name is different for Oracle Solaris Cluster resources and Oracle Clusterware resources. For more information, see the following table.

TABLE 5–3 Comparisons of States for Oracle Solaris Cluster Resources and Oracle Clusterware Resources

State	Meaning for Oracle Solaris Cluster Resources	Meaning for Oracle Clusterware Resources
Enabled	The resource is available to the Oracle Solaris Cluster RGM for automatic startup, failover, or restart. A resource that is enabled can also be in either the online state or the offline state.	The resource is available to run under Oracle Clusterware for automatic startup, failover, or restart. A resource that is enabled can also be in either the online state or the offline state.
Disabled	The resource is unavailable to the Oracle Solaris Cluster RGM for automatic startup, failover, or restart. A resource that is disabled is also offline.	The resource is unavailable to run under the Oracle Clusterware for automatic startup, failover, or restart. A resource that is disabled can also be in either the online state or the offline state.
Online	The resource is running and providing service.	The resource is running and providing service. A resource that is online must also be enabled.
Offline	The resource is stopped and not providing service.	The resource is stopped and not providing service. A resource that is offline can also be in either the disabled state or the enabled state.

For detailed information about the state of Oracle Solaris Cluster resources, see "Resource and Resource Group States and Settings" in *Oracle Solaris Cluster Concepts Guide*.

For detailed information about the state of Oracle Clusterware resources, see your Oracle documentation.

Effects of State Changes to Oracle Solaris Cluster Resources for Oracle 9*i* RAC Database Instances

In Oracle 9*i*, Oracle database components can be stopped and started exclusively under the control of the Oracle Solaris Cluster RGM. The effects of state changes to Oracle Solaris Cluster resources for Oracle 9*i* RAC database instances are as follows:

- Bringing online a resource for an Oracle 9*i* RAC database component starts the component on the nodes where the resource is brought online.
- Taking offline a resource for an Oracle 9*i* RAC database component stops the component on the nodes where the resource is taken offline.

Tuning Support for Oracle RAC

To tune the Support for Oracle RAC data service, you modify the extension properties of the resources for this data service. For details about these extension properties, see Appendix C, "Support for Oracle RAC Extension Properties." Typically, you use the option **-p** *property*=*value* of the `clresource`(1CL) command to set extension properties of Support for Oracle RAC resources. You can also use the procedures in Chapter 2, "Administering Data Service Resources," in *Oracle Solaris Cluster Data Services Planning and Administration Guide* to configure the resources later.

This section contains the following information about tuning the Support for Oracle RAC data service:

- "Guidelines for Setting Timeouts" on page 136
- "SPARC: Guidelines for Setting the Communications Port Range for the Oracle UDLM" on page 138
- "How to Modify an Extension Property That Is Tunable Only When a Resource Is Disabled" on page 138

Guidelines for Setting Timeouts

Many of the extension properties for Support for Oracle RAC specify timeouts for steps in reconfiguration processes. The optimum values for most of these timeouts are independent of your cluster configuration. Therefore, you should not need to change the timeouts from their default values.

Timeouts that depend on your cluster configuration are described in the subsections that follow. If timeouts occur during reconfiguration processes, increase the values of these timeout properties to accommodate your cluster configuration.

This section describes the following timeouts:

SPARC: VxVM Component Reconfiguration Step 4 Timeout

The time that is required for step 4 of a reconfiguration of the VxVM component of Support for Oracle RAC is affected by the size and complexity of your Veritas shared-disk group configuration. If your Veritas shared-disk group configuration is large or complex and the reconfiguration of the VxVM component times out, increase the timeout for step 4 of a reconfiguration of the VxVM component.

To increase the timeout for step 4 of a reconfiguration of the VxVM component, increase the value of the `Cvm_step4_timeout` extension property of the `SUNW.rac_cvm` resource.

For more information, see "SPARC: `SUNW.rac_cvm` Extension Properties" on page 265.

EXAMPLE 5–1 Setting the VxVM Component Reconfiguration Step 4 Timeout

```
# clresource set -p cvm_step4_timeout=1200 rac-cvm-rs
```

This example sets the timeout for step 4 of a reconfiguration of the VxVM component to 1200 seconds. This example assumes that the VxVM component is represented by an instance of the `SUNW.rac_cvm` resource type that is named `rac-cvm-rs`.

Reservation Step Timeout

The time that is required for reservation commands to run is affected by the following factors:

- The number of shared physical disks in the cluster
- The load on the cluster

If the number of shared physical disks in the cluster is large, or if your cluster is heavily loaded, the reconfiguration of Support for Oracle RAC might time out. If such a timeout occurs, increase the reservation step timeout.

To increase the reservation step timeout, increase the `Reservation_timeout` extension property of the `SUNW.rac_framework` resource.

For more information, see "`SUNW.rac_framework` Extension Properties" on page 268.

EXAMPLE 5–2 Setting the Reservation Step Timeout

```
# clresource set -p reservation_timeout=350 rac-framework-rs
```

EXAMPLE 5–2 Setting the Reservation Step Timeout *(Continued)*

This example sets the timeout for the reservation step of a reconfiguration of Support for Oracle RAC to 350 seconds. This example assumes that RAC framework component is represented by an instance of the `SUNW.rac_framework` resource type that is named `rac-frameowrk-rs`.

SPARC: Guidelines for Setting the Communications Port Range for the Oracle UDLM

An application other than the Oracle UDLM on a cluster node might use a range of communications ports that conflicts with the range for the Oracle UDLM. If such a conflict occurs, modify the range of communications ports that the Oracle UDLM uses.

The range of communications ports that the Oracle UDLM uses is determined by the values of the following extension properties of the `SUNW.rac_udlm` resource type:

- `Port`. Specifies the communications port number that the Oracle UDLM uses. The first number in the range of communications port numbers that the Oracle UDLM uses is the value of `Port`.
- `Num_ports`. Specifies the number of communications ports that the Oracle UDLM uses. The last number in the range of communications port numbers that the Oracle UDLM uses is the sum of the values of `Port` and `Num_ports`.

For more information, see "SPARC: `SUNW.rac_udlm` Extension Properties" on page 270.

EXAMPLE 5–3 Setting the Communications Port Number for the Oracle UDLM

```
# clresource set -p port=7000 rac-udlm-rs
```

This example sets the communications port number that the Oracle UDLM uses to 7000. The following assumptions apply to this example:

- The Oracle UDLM component is represented by an instance of the `SUNW.rac_udlm` resource type that is named `rac-udlm-rs`.
- The command in this example is run as part of the procedure for modifying an extension property that is tunable only when disabled. For more information, see "How to Modify an Extension Property That Is Tunable Only When a Resource Is Disabled" on page 138.

▼ How to Modify an Extension Property That Is Tunable Only When a Resource Is Disabled

Restrictions apply to the circumstances in which you can modify an extension property that is tunable only when a resource is disabled. Those circumstances depend on the resource type as follows:

- SPARC: `SUNW.rac_udlm` – Only when the Oracle UDLM is *not* running on any cluster node
- SPARC: `SUNW.rac_cvm` – Only when VxVM is *not* running in cluster mode on any cluster node

1. **Disable each resource that the RAC framework resource group contains and bring the RAC framework resource group into the `UNMANAGED` state.**

   Disable the instance of the `SUNW.rac_framework` resource only after you have disabled all other resources that the RAC framework resource group contains. The other resources in the RAC framework resource group depend on the `SUNW.rac_framework` resource.

   For detailed instructions, see "Disabling Resources and Moving Their Resource Group Into the UNMANAGED State" in *Oracle Solaris Cluster Data Services Planning and Administration Guide*.

2. **Reboot all the nodes that are in the node list of the RAC framework resource group.**

3. **Use the `clresource` command to set the property to its new value.**

   ```
   # clresource set -p property=value resource
   ```

   property Specifies the name of the property that you are changing.

   value The new value of the property.

   resource Specifies the name of the resource for which you are modifying an extension property. If this resource was created by using the `clsetup` utility, the name depends on the resource type, as shown in "Automatically Generated Names for Oracle Solaris Cluster Objects" on page 132.

4. **Bring the RAC framework resource group and its resources online.**

   ```
   # clresourcegroup online resource-group
   ```

   resource-group Specifies the name of the RAC framework resource group that is to be moved to the `MANAGED` state and brought online. If this resource group was created by using the `clsetup` utility, the name of the resource group is `rac-framework-rg`.

Tuning the Support for Oracle RAC Fault Monitors

Fault monitoring for the Support for Oracle RAC data service is provided by fault monitors for the following resources:

- Scalable device group resource
- Scalable file-system mount-point resource
- Oracle 9*i* RAC server resource
- Oracle 9*i* RAC listener resource

Each fault monitor is contained in a resource whose resource type is shown in the following table.

TABLE 5–4 Resource Types for Support for Oracle RAC Fault Monitors

Fault Monitor	Resource Type
Scalable device group	`SUNW.ScalDeviceGroup`
Scalable file-system mount point	`SUNW.ScalMountPoint`
Oracle 9*i* RAC server	`SUNW.scalable_rac_server`
Oracle 9*i*RAC listener	`SUNW.scalable_rac_listener`

System properties and extension properties of these resources control the behavior of the fault monitors. The default values of these properties determine the preset behavior of the fault monitors. The preset behavior should be suitable for most Oracle Solaris Cluster installations. Therefore, you should tune the Support for Oracle RAC fault monitors *only* if you need to modify this preset behavior.

Tuning the Support for Oracle RAC fault monitors involves the following tasks:

- Setting the interval between fault monitor probes
- Setting the timeout for fault monitor probes
- Defining the criteria for persistent faults
- Specifying the failover behavior of a resource

For more information, see "Tuning Fault Monitors for Oracle Solaris Cluster Data Services" in *Oracle Solaris Cluster Data Services Planning and Administration Guide*. Information about the Support for Oracle RAC fault monitors that you need to perform these tasks is provided in the subsections that follow:

- "Operation of the Fault Monitor for a Scalable Device Group" on page 140
- "Operation of the Fault Monitor for Scalable File-System Mount Points" on page 141
- "Operation of the Oracle 9i RAC Server Fault Monitor" on page 142
- "Operation of the Oracle 9i RAC Listener Fault Monitor" on page 144
- "Obtaining Core Files for Troubleshooting DBMS Timeouts" on page 144

Operation of the Fault Monitor for a Scalable Device Group

By default, the fault monitor monitors all logical volumes in the device group that the resource represents. If you require only a subset of the logical volumes in a device group to be monitored, set the `LogicalDeviceList` extension property.

The status of the device group is derived from the statuses of the individual logical volumes that are monitored. If all monitored logical volumes are healthy, the device group is healthy. If any monitored logical volume is faulty, the device group is faulty. If a device group is discovered to be faulty, monitoring of the resource that represents the group is stopped and the resource is put into the disabled state.

The status of an individual logical volume is obtained by querying the volume's volume manager. If the status of a Solaris Volume Manager for Sun Cluster volume cannot be determined from a query, the fault monitor performs file input/output (I/O) operations to determine the status.

Note – For mirrored disks, if one submirror is faulty, the device group is still considered to be healthy.

If reconfiguration of userland cluster membership causes an I/O error, the monitoring of device group resources by fault monitors is suspended while userland cluster membership monitor (UCMM) reconfigurations are in progress.

Operation of the Fault Monitor for Scalable File-System Mount Points

To determine if the mounted file system is available, the fault monitor performs I/O operations such as opening, reading, and writing to a test file on the file system. If an I/O operation is not completed within the timeout period, the fault monitor reports an error. To specify the timeout for I/O operations, set the `IOTimeout` extension property.

The response to an error depends on the type of the file system, as follows:

- If the file system is an NFS file system on a qualified NAS device, the response is as follows:
  - Monitoring of the resource is stopped on the current node.
  - The resource is placed into the disabled state on the current node, causing the file system to be unmounted from that node.
- If the file system is a Sun QFS shared file system, the response is as follows:
  - If the node on which the error occurred is hosting the metadata server resource, the metadata server resource is failed over to another node.
  - The file system is unmounted.

  If the failover attempt fails, the file system remains unmounted and a warning is given.

Operation of the Oracle 9i RAC Server Fault Monitor

The fault monitor for the Oracle 9i RAC server uses a request to the server to query the health of the server.

The server fault monitor is started through `pmfadm` to make the monitor highly available. If the monitor is killed for any reason, the Process Monitor Facility (PMF) automatically restarts the monitor.

The server fault monitor consists of the following processes.

- A main fault monitor process
- A database client fault probe

This section contains the following information about the server fault monitor:

- "Operation of the Main Fault Monitor" on page 142
- "Operation of the Database Client Fault Probe" on page 142
- "Actions by the Server Fault Monitor in Response to a Database Transaction Failure" on page 143
- "Scanning of Logged Alerts by the Server Fault Monitor" on page 143

Operation of the Main Fault Monitor

The main fault monitor determines that an operation is successful if the database is online and no errors are returned during the transaction.

Operation of the Database Client Fault Probe

The database client fault probe performs the following operations:

1. Monitoring the partition for archived redo logs. See "Operations to Monitor the Partition for Archived Redo Logs" on page 142.
2. If the partition is healthy, determining whether the database is operational. See "Operations to Determine Whether the Database is Operational" on page 143.

The probe uses the timeout value that is set in the resource property `Probe_timeout` to determine how much time to allocate to successfully probe Oracle.

Operations to Monitor the Partition for Archived Redo Logs

The database client fault probe queries the dynamic performance view `v$archive_dest` to determine all possible destinations for archived redo logs. For every active destination, the probe determines whether the destination is healthy and has sufficient free space for storing archived redo logs.

- If the destination is healthy, the probe determines the amount of free space in the destination's file system. If the amount of free space is less than 10% of the file system's capacity and is less than 20 Mbytes, the probe prints a message to `syslog`.

- If the destination is in `ERROR` status, the probe prints a message to `syslog` and disables operations to determine whether the database is operational. The operations remain disabled until the error condition is cleared .

Operations to Determine Whether the Database is Operational

If the partition for archived redo logs is healthy, the database client fault probe queries the dynamic performance view `v$sysstat` to obtain database performance statistics. Changes to these statistics indicate that the database is operational. If these statistics remain unchanged between consecutive queries, the fault probe performs database transactions to determine if the database is operational. These transactions involve the creation, updating, and dropping of a table in the user table space.

The database client fault probe performs all its transactions as the Oracle user. The ID of this user is specified during the preparation of the nodes or zones as explained in "How to Create the DBA Group and the DBA User Accounts" on page 33.

Actions by the Server Fault Monitor in Response to a Database Transaction Failure

If a database transaction fails, the server fault monitor performs an action that is determined by the error that caused the failure. To change the action that the server fault monitor performs, customize the server fault monitor as explained in "Customizing the Oracle 9i RAC Server Fault Monitor" on page 145.

If the action requires an external program to be run, the program is run as a separate process in the background.

Possible actions are as follows:

- **Ignore.** The server fault monitor ignores the error.
- **Stop monitoring.** The server fault monitor is stopped without shutting down the database.
- **Restart.** The server fault monitor stops and restarts the Oracle 9i RAC server resource.

Scanning of Logged Alerts by the Server Fault Monitor

The Oracle software logs alerts in an alert log file. The absolute path of this file is specified by the `alert_log_file` extension property of the `SUNW.scalable_rac_server` resource. The server fault monitor scans the alert log file for new alerts at the following times:

- When the server fault monitor is started
- Each time that the server fault monitor queries the health of the server

If an action is defined for a logged alert that the server fault monitor detects, the server fault monitor performs the action in response to the alert.

Preset actions for logged alerts are listed in Table B–2. To change the action that the server fault monitor performs, customize the server fault monitor as explained in "Customizing the Oracle 9i RAC Server Fault Monitor" on page 145.

Operation of the Oracle 9i RAC Listener Fault Monitor

The Oracle 9i RAC listener fault monitor checks the status of an Oracle listener.

If the listener is running, the Oracle 9i RAC listener fault monitor considers a probe successful. If the fault monitor detects an error, the listener is restarted.

Note – The listener resource does not provide a mechanism for setting the listener password. If Oracle listener security is enabled, a probe by the listener fault monitor might return Oracle error TNS-01169. Because the listener is able to respond, the listener fault monitor treats the probe as a success. This action does not cause a failure of the listener to remain undetected. A failure of the listener returns a different error, or causes the probe to time out.

The listener probe is started through `pmfadm` to make the probe highly available. If the probe is killed, PMF automatically restarts the probe.

If a problem occurs with the listener during a probe, the probe tries to restart the listener. The value that is set for the resource property `retry_count` determines the maximum number of times that the probe attempts the restart. If, after trying for the maximum number of times, the probe is still unsuccessful, the probe stops the fault monitor.

Obtaining Core Files for Troubleshooting DBMS Timeouts

To facilitate troubleshooting of unexplained DBMS timeouts, you can enable the fault monitor to create a core file when a probe timeout occurs. The contents of the core file relate to the fault monitor process. The fault monitor creates the core file in the `/` directory. To enable the fault monitor to create a core file, use the `coreadm` command to enable set-id core dumps. For more information, see the `coreadm(1M)` man page.

Customizing the Oracle 9i RAC Server Fault Monitor

Customizing the Oracle 9i RAC server fault monitor enables you to modify the behavior of the server fault monitor as follows:

- Overriding the preset action for an error
- Specifying an action for an error for which no action is preset

Caution – Before you customize the Oracle 9i RAC server fault monitor, consider the effects of your customizations, especially if you change an action from restart or switch over to ignore or stop monitoring. If errors remain uncorrected for long periods, the errors might cause problems with the database. If you encounter problems with the database after customizing the Oracle 9i RAC server fault monitor, revert to using the preset actions. Reverting to the preset actions enables you to determine if the problem is caused by your customizations.

Customizing the Oracle 9i RAC server fault monitor involves the following activities:

1. Defining custom behavior for errors
2. Propagating a custom action file to all nodes in a cluster
3. Specifying the custom action file that a server fault monitor should use

Defining Custom Behavior for Errors

The Oracle 9i RAC server fault monitor detects the following types of errors:

- DBMS errors that occur during a probe of the database by the server fault monitor
- Alerts that Oracle logs in the alert log file
- Timeouts that result from a failure to receive a response within the time that is set by the `Probe_timeout` extension property

To define custom behavior for these types of errors, create a custom action file. This section contains the following information about custom action files:

- "Custom Action File Format" on page 145
- "Changing the Response to a DBMS Error" on page 148
- "Changing the Response to Logged Alerts" on page 150
- "Changing the Maximum Number of Consecutive Timed-Out Probes" on page 151

Custom Action File Format

A custom action file is a plain text file. The file contains one or more entries that define the custom behavior of the Oracle 9i RAC server fault monitor. Each entry defines the custom behavior for a single DBMS error, a single timeout error, or several logged alerts. A maximum of 1024 entries is allowed in a custom action file.

Note – Each entry in a custom action file overrides the preset action for an error, or specifies an action for an error for which no action is preset. Create entries in a custom action file *only* for the preset actions that you are overriding or for errors for which no action is preset. Do *not* create entries for actions that you are not changing.

An entry in a custom action file consists of a sequence of keyword-value pairs that are separated by semicolons. Each entry is enclosed in braces.

The format of an entry in a custom action file is as follows:

```
{
[ERROR_TYPE=DBMS_ERROR|SCAN_LOG|TIMEOUT_ERROR;]
ERROR=error-spec;
[ACTION=RESTART|STOP|NONE;]
[CONNECTION_STATE=co|di|on|*;]
[NEW_STATE=co|di|on|*;]
[MESSAGE="message-string"]
}
```

White space may be used between separated keyword-value pairs and between entries to format the file.

The meaning and permitted values of the keywords in a custom action file are as follows:

`ERROR_TYPE`
Indicates the type of the error that the server fault monitor has detected. The following values are permitted for this keyword:

`DBMS_ERROR`	Specifies that the error is a DBMS error.
`SCAN_LOG`	Specifies that the error is an alert that is logged in the alert log file.
`TIMEOUT_ERROR`	Specifies that the error is a timeout.

The `ERROR_TYPE` keyword is optional. If you omit this keyword, the error is assumed to be a DBMS error.

`ERROR`
Identifies the error. The data type and the meaning of *error-spec* are determined by the value of the `ERROR_TYPE` keyword as shown in the following table.

ERROR_TYPE	Data Type	Meaning
`DBMS_ERROR`	Integer	The error number of a DBMS error that is generated by Oracle
`SCAN_LOG`	Quoted regular expression	A string in an error message that Oracle has logged to the Oracle alert log file

ERROR_TYPE	Data Type	Meaning
TIMEOUT_ERROR	Integer	The number of consecutive timed-out probes since the server fault monitor was last started or restarted

You must specify the ERROR keyword. If you omit this keyword, the entry in the custom action file is ignored.

ACTION
Specifies the action that the server fault monitor is to perform in response to the error. The following values are permitted for this keyword:

NONE Specifies that the server fault monitor ignores the error.

STOP Specifies that the server fault monitor is stopped.

RESTART Specifies that the server fault monitor stops and restarts the Oracle 9i RAC server resource.

The ACTION keyword is optional. If you omit this keyword, the server fault monitor ignores the error.

CONNECTION_STATE
Specifies the required state of the connection between the database and the server fault monitor when the error is detected. The entry applies only if the connection is in the required state when the error is detected. The following values are permitted for this keyword:

* Specifies that the entry always applies, regardless of the state of the connection.

co Specifies that the entry applies only if the server fault monitor is attempting to connect to the database.

on Specifies that the entry applies only if the server fault monitor is online. The server fault monitor is online if it is connected to the database.

di Specifies that the entry applies only if the server fault monitor is disconnecting from the database.

The CONNECTION_STATE keyword is optional. If you omit this keyword, the entry always applies, regardless of the state of the connection.

NEW_STATE
Specifies the state of the connection between the database and the server fault monitor that the server fault monitor must attain after the error is detected. The following values are permitted for this keyword:

* Specifies that the state of the connection must remain unchanged.

`co` Specifies that the server fault monitor must disconnect from the database and reconnect immediately to the database.

`di` Specifies that the server fault monitor must disconnect from the database. The server fault monitor reconnects when it next probes the database.

The `NEW_STATE` keyword is optional. If you omit this keyword, the state of the database connection remains unchanged after the error is detected.

`MESSAGE`
Specifies an additional message that is printed to the resource's log file when this error is detected. The message must be enclosed in double quotes. This message is additional to the standard message that is defined for the error.

The `MESSAGE` keyword is optional. If you omit this keyword, no additional message is printed to the resource's log file when this error is detected.

Changing the Response to a DBMS Error

The action that the server fault monitor performs in response to each DBMS error is preset as listed in Table B–1. To determine whether you need to change the response to a DBMS error, consider the effect of DBMS errors on your database to determine if the preset actions are appropriate. For examples, see the subsections that follow:

- "Responding to an Error Whose Effects Are Major" on page 148
- "Ignoring an Error Whose Effects Are Minor" on page 149

To change the response to a DBMS error, create an entry in a custom action file in which the keywords are set as follows:

- `ERROR_TYPE` is set to `DBMS_ERROR`.
- `ERROR` is set to the error number of the DBMS error.
- `ACTION` is set to the action that you require.

Responding to an Error Whose Effects Are Major

If an error that the server fault monitor ignores affects more than one session, action by the server fault monitor might be required to prevent a loss of service.

For example, no action is preset for Oracle error 4031: `unable to allocate` *num-bytes* `bytes of shared memory`. However, this Oracle error indicates that the shared global area (SGA) has insufficient memory, is badly fragmented, or both states apply. If this error affects only a single session, ignoring the error might be appropriate. However, if this error affects more than one session, consider specifying that the server fault monitor restart the database.

The following example shows an entry in a custom action file for changing the response to a DBMS error to restart.

EXAMPLE 5–4 Changing the Response to a DBMS Error to Restart

```
{
ERROR_TYPE=DBMS_ERROR;
ERROR=4031;
ACTION=restart;
CONNECTION_STATE=*;
NEW_STATE=*;
MESSAGE="Insufficient memory in shared pool.";
}
```

This example shows an entry in a custom action file that overrides the preset action for DBMS error 4031. This entry specifies the following behavior:

- In response to DBMS error 4031, the action that the server fault monitor performs is restart.
- This entry applies regardless of the state of the connection between the database and the server fault monitor when the error is detected.
- The state of the connection between the database and the server fault monitor must remain unchanged after the error is detected.
- The following message is printed to the resource's log file when this error is detected:

  ```
  Insufficient memory in shared pool.
  ```

Ignoring an Error Whose Effects Are Minor

If the effects of an error to which the server fault monitor responds are minor, ignoring the error might be less disruptive than responding to the error.

For example, the preset action for Oracle error 4030: `out of process memory when trying to allocate` *num-bytes* `bytes` is restart. This Oracle error indicates that the server fault monitor could not allocate private heap memory. One possible cause of this error is that insufficient memory is available to the operating system. If this error affects more than one session, restarting the database might be appropriate. However, this error might not affect other sessions because these sessions do not require further private memory. In this situation, consider specifying that the server fault monitor ignore the error.

The following example shows an entry in a custom action file for ignoring a DBMS error.

EXAMPLE 5–5 Ignoring a DBMS Error

```
{
ERROR_TYPE=DBMS_ERROR;
ERROR=4030;
ACTION=none;
CONNECTION_STATE=*;
NEW_STATE=*;
MESSAGE="";
}
```

This example shows an entry in a custom action file that overrides the preset action for DBMS error 4030. This entry specifies the following behavior:

EXAMPLE 5–5 Ignoring a DBMS Error *(Continued)*

- The server fault monitor ignores DBMS error 4030.
- This entry applies regardless of the state of the connection between the database and the server fault monitor when the error is detected.
- The state of the connection between the database and the server fault monitor must remain unchanged after the error is detected.
- No additional message is printed to the resource's log file when this error is detected.

Changing the Response to Logged Alerts

The Oracle software logs alerts in a file that is identified by the `alert_log_file` extension property. The server fault monitor scans this file and performs actions in response to alerts for which an action is defined.

Logged alerts for which an action is preset are listed in Table B–2. Change the response to logged alerts to change the preset action, or to define new alerts to which the server fault monitor responds.

To change the response to logged alerts, create an entry in a custom action file in which the keywords are set as follows:

- `ERROR_TYPE` is set to `SCAN_LOG`.
- `ERROR` is set to a quoted regular expression that identifies a string in an error message that Oracle has logged to the Oracle alert log file.
- `ACTION` is set to the action that you require.

The server fault monitor processes the entries in a custom action file in the order in which the entries occur. Only the first entry that matches a logged alert is processed. Later entries that match are ignored. If you are using regular expressions to specify actions for several logged alerts, ensure that more specific entries occur before more general entries. Specific entries that occur after general entries might be ignored.

For example, a custom action file might define different actions for errors that are identified by the regular expressions `ORA-65` and `ORA-6`. To ensure that the entry that contains the regular expression `ORA-65` is not ignored, ensure that this entry occurs before the entry that contains the regular expression `ORA-6`.

The following example shows an entry in a custom action file for changing the response to a logged alert.

EXAMPLE 5–6 Changing the Response to a Logged Alert

```
{
ERROR_TYPE=SCAN_LOG;
ERROR="ORA-00600: internal error";
```

EXAMPLE 5–6 Changing the Response to a Logged Alert *(Continued)*

```
ACTION=RESTART;
}
```

This example shows an entry in a custom action file that overrides the preset action for logged alerts about internal errors. This entry specifies the following behavior:

- In response to logged alerts that contain the text `ORA-00600: internal error`, the action that the server fault monitor performs is restart.
- This entry applies regardless of the state of the connection between the database and the server fault monitor when the error is detected.
- The state of the connection between the database and the server fault monitor must remain unchanged after the error is detected.
- No additional message is printed to the resource's log file when this error is detected.

Changing the Maximum Number of Consecutive Timed-Out Probes

By default, the server fault monitor restarts the database after the second consecutive timed-out probe. If the database is lightly loaded, two consecutive timed-out probes should be sufficient to indicate that the database is hanging. However, during periods of heavy load, a server fault monitor probe might time out even if the database is functioning correctly. To prevent the server fault monitor from restarting the database unnecessarily, increase the maximum number of consecutive timed-out probes.

Caution – Increasing the maximum number of consecutive timed-out probes increases the time that is required to detect that the database is hanging.

To change the maximum number of consecutive timed-out probes allowed, create one entry in a custom action file for each consecutive timed-out probe that is allowed *except* the first timed-out probe.

Note – You are not required to create an entry for the first timed-out probe. The action that the server fault monitor performs in response to the first timed-out probe is preset.

For the last allowed timed-out probe, create an entry in which the keywords are set as follows:

- `ERROR_TYPE` is set to `TIMEOUT_ERROR`.
- `ERROR` is set to the maximum number of consecutive timed-out probes that are allowed.
- `ACTION` is set to `RESTART`.

For each remaining consecutive timed-out probe except the first timed-out probe, create an entry in which the keywords are set as follows:

- `ERROR_TYPE` is set to `TIMEOUT_ERROR`.
- `ERROR` is set to the sequence number of the timed-out probe. For example, for the second consecutive timed-out probe, set this keyword to 2. For the third consecutive timed-out probe, set this keyword to 3.
- `ACTION` is set to `NONE`.

Tip – To facilitate debugging, specify a message that indicates the sequence number of the timed-out probe.

The following example shows the entries in a custom action file for increasing the maximum number of consecutive timed-out probes to five.

EXAMPLE 5–7 Changing the Maximum Number of Consecutive Timed-Out Probes

```
{
ERROR_TYPE=TIMEOUT;
ERROR=2;
ACTION=NONE;
CONNECTION_STATE=*;
NEW_STATE=*;
MESSAGE="Timeout #2 has occurred.";
}

{
ERROR_TYPE=TIMEOUT;
ERROR=3;
ACTION=NONE;
CONNECTION_STATE=*;
NEW_STATE=*;
MESSAGE="Timeout #3 has occurred.";
}

{
ERROR_TYPE=TIMEOUT;
ERROR=4;
ACTION=NONE;
CONNECTION_STATE=*;
NEW_STATE=*;
MESSAGE="Timeout #4 has occurred.";
}

{
ERROR_TYPE=TIMEOUT;
ERROR=5;
ACTION=RESTART;
CONNECTION_STATE=*;
NEW_STATE=*;
MESSAGE="Timeout #5 has occurred. Restarting.";
}
```

This example shows the entries in a custom action file for increasing the maximum number of consecutive timed-out probes to five. These entries specify the following behavior:

EXAMPLE 5–7 Changing the Maximum Number of Consecutive Timed-Out Probes *(Continued)*

- The server fault monitor ignores the second consecutive timed-out probe through the fourth consecutive timed-out probe.
- In response to the fifth consecutive timed-out probe, the action that the server fault monitor performs is restart.
- The entries apply regardless of the state of the connection between the database and the server fault monitor when the timeout occurs.
- The state of the connection between the database and the server fault monitor must remain unchanged after the timeout occurs.
- When the second consecutive timed-out probe through the fourth consecutive timed-out probe occurs, a message of the following form is printed to the resource's log file:

  `Timeout #`*`number`* `has occurred.`
- When the fifth consecutive timed-out probe occurs, the following message is printed to the resource's log file:

  ```
  Timeout #5 has occurred. Restarting.
  ```

Propagating a Custom Action File to All Nodes in a Cluster

A server fault monitor must behave consistently on all cluster nodes. Therefore, the custom action file that the server fault monitor uses must be identical on all cluster nodes. After creating or modifying a custom action file, ensure that this file is identical on all cluster nodes by propagating the file to all cluster nodes. To propagate the file to all cluster nodes, use the method that is most appropriate for your cluster configuration:

- Locating the file on a file system that all nodes share
- Locating the file on a highly available local file system
- Copying the file to the local file system of each cluster node by using operating system commands such as the `rcp(1)` command or the `rdist(1)` command

Specifying the Custom Action File That a Server Fault Monitor Should Use

To apply customized actions to a server fault monitor, you must specify the custom action file that the fault monitor should use. Customized actions are applied to a server fault monitor when the server fault monitor reads a custom action file. A server fault monitor reads a custom action file when the you specify the file.

Specifying a custom action file also validates the file. If the file contains syntax errors, an error message is displayed. Therefore, after modifying a custom action file, specify the file again to validate the file.

Caution – If syntax errors in a modified custom action file are detected, correct the errors before the fault monitor is restarted. If the syntax errors remain uncorrected when the fault monitor is restarted, the fault monitor reads the erroneous file, ignoring entries that occur after the first syntax error.

How to Specify the Custom Action File That a Server Fault Monitor Should Use

1. **On a cluster node, become superuser or assume a role that provides `solaris.cluster.modify` RBAC authorization.**

2. **Set the `Custom_action_file` extension property of the `SUNW.scalable_rac_server` resource.**

   Set this property to the absolute path of the custom action file.

   ```
   # clresource set -p custom_action_file=filepath  server-resource
   ```

   `-p custom_action_file=`*filepath*
   Specifies the absolute path of the custom action file.

   server-resource
   Specifies the `SUNW.scalable_rac_server` resource.

◆ ◆ ◆ CHAPTER 6

Troubleshooting Support for Oracle RAC

If you encounter a problem with Support for Oracle RAC, troubleshoot the problem by using the techniques that are described in the following sections.

Verifying the Status of Support for Oracle RAC

The status of resource groups and resources for Support for Oracle RAC indicates the status of Oracle RAC in your cluster. Use Oracle Solaris Cluster maintenance commands to obtain this status information.

- To obtain status information for resource groups, use the `clresourcegroup`(1CL) command.
- To obtain status information for resources, use the `clresource`(1CL) command.

▼ How to Verify the Status of Support for Oracle RAC

1. **Become superuser or assume a role that provides `solaris.cluster.read` RBAC authorization.**
2. **Display status information for the Oracle Solaris Cluster objects in which you are interested.**

   For example:

   - **To display status information for all resource groups in your cluster, type the following command:**

     ```
     # clresourcegroup status +
     ```

- **To display status information for all resources in a resource group, type the following command:**

  ```
  # clresource status -g resource-group +
  ```

  resource-group Specifies the resource group that contains the resources whose status information you are displaying.

See Also For information about options that you can specify to filter the status information that is displayed, see the following man pages:

- `clresource`(1CL)
- `clresourcegroup`(1CL)

Examples of the Status of Support for Oracle RAC

The following examples show the status of resource groups and resources for a configuration of Support for Oracle RAC on a four-node cluster. Each node is a machine that uses the SPARC processor.

The cluster in this example is running version 10*g* Release 2 or 11*g* of Oracle RAC. The configuration in this example uses a Sun QFS shared file system on Solaris Volume Manager for Sun Cluster to store Oracle files. The configuration includes a multiple-owner volume-manager framework resource group to contain the volume manager resource.

The resource groups and resources for this configuration are shown in the following table.

		Resource Group Contents	
Resource Group	Purpose	Resource Type	Resource Instance Name
`rac-framework-rg`	RAC framework resource group	`SUNW.rac_framework`	`rac-framework-rs`
		`SUNW.rac_udlm`	`rac-udlm-rs`
		`SUNW.crs_framework`	`crs_framework-rs`
`vucmm-framework-rg`	multiple-owner volume-manager framework resource group	`SUNW.vucmm_framework`	`vucmm-framework-rs`
		`SUNW.vucmm_svm`	`vucmm-svm-rs`
`scaldg-rg`	Resource group for scalable device-group resources	`SUNW.ScalDeviceGroup`	`scaloradg-rs`

Resource Group	Purpose	Resource Group Contents	
		Resource Type	Resource Instance Name
qfsmds-rg	Resource group for Sun QFS metadata server resources	SUNW.qfs	qfs-db_qfs-OraHome-rs qfs-db_qfs-OraData-rs
scalmnt-rg	Resource group for scalable file-system mount-point resources	SUNW.ScalMountPoint	scal-db_qfs-OraHome-rs scal-db_qfs-OraData-rs
rac_server_proxy-rg	RAC database resource group	SUNW.scalable_rac_server_proxy	rac_server_proxy-rs

EXAMPLE 6–1 Status of a Faulty RAC Framework Resource Group

This example provides the following status information for a RAC framework resource group that is faulty.

- A reconfiguration error has prevented the `rac_framework` resource from starting on cluster node `pclus1`.
- The effects of this reconfiguration error on resource groups and other resources are as follows:
  - The `rac-framework-rg` resource group is offline and in the start failed state on cluster node `pclus1`.
  - The `rac_udlm` resource, the `rac_svm` resource, and the `crs_framework` resource are offline on cluster node `pclus1`.
  - All other multi-master resource groups and the resources that the groups contain are offline on cluster node `pclus1`.
  - All failover resource groups have failed over from cluster node `pclus1` to a secondary node.
- All multi-master resource groups and the resources that the groups contain are online on the remaining nodes.

```
# clresourcegroup status +

=== Cluster Resource Groups ===

Group Name             Node Name    Suspended   Status
----------             ---------    ---------   ------
rac-framework-rg       pclus1       No          Online faulted
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

vucmm-framework-rg     pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
```

EXAMPLE 6–1 Status of a Faulty RAC Framework Resource Group *(Continued)*

```
                      pclus4       No          Online

scaldg-rg             pclus1       No          Online
                      pclus2       No          Online
                      pclus3       No          Online
                      pclus4       No          Online

qfsmds-rg             pclus1       No          Offline
                      pclus2       No          Online
                      pclus3       No          Offline
                      pclus4       No          Offline

scalmnt-rg            pclus1       No          Online
                      pclus2       No          Online
                      pclus3       No          Online
                      pclus4       No          Online

rac_server_proxy-rg   pclus1       No          Pending online blocked
                      pclus2       No          Online
                      pclus3       No          Online
                      pclus4       No          Online

# clresource status -g rac-framework-rg +

=== Cluster Resources ===

Resource Name       Node Name    State          Status Message
-------------       ---------    -----          --------------
rac-framework-rs    pclus1       Start failed   Faulted - Error in previous reconfiguration.
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

rac-udlm-rs         pclus1       Offline        Offline
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

crs_framework-rs    pclus1       Offline        Offline
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

# clresource status -g vucmm-framework-rg +

=== Cluster Resources ===

Resource Name       Node Name    State          Status Message
-------------       ---------    -----          --------------
vucmm-framework-rs  pclus1       Online         Online
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

vucmm-svm-rs        pclus1       Offline        Offline
                    pclus2       Online         Online
```

EXAMPLE 6–1 Status of a Faulty RAC Framework Resource Group *(Continued)*

```
                    pclus3       Online         Online
                    pclus4       Online         Online

# clresource status -g scaldg-rg +

=== Cluster Resources ===

Resource Name       Node Name      State        Status Message
-------------       ---------      -----        --------------
scaloradg-rs        pclus1         Online       Online - Diskgroup online
                    pclus2         Online       Online - Diskgroup online
                    pclus3         Online       Online - Diskgroup online
                    pclus4         Online       Online - Diskgroup online

# clresource status -g qfsmds-rg +

=== Cluster Resources ===

Resource Name            Node Name    State     Status Message
-------------            ---------    -----     --------------
qfs-db_qfs-OraHome-rs    pclus1       Offline   Offline
                         pclus2       Online    Online - Service is online.
                         pclus3       Offline   Offline
                         pclus4       Offline   Offline

qfs-db_qfs-OraData-rs    pclus1       Offline   Offline
                         pclus2       Online    Online - Service is online.
                         pclus3       Offline   Offline
                         pclus4       Offline   Offline

# clresource status -g scalmnt-rg +

=== Cluster Resources ===

Resource Name             Node Name   State     Status Message
-------------             ---------   -----     --------------
scal-db_qfs-OraHome-rs    pclus1      Online    Online
                          pclus2      Online    Online
                          pclus3      Online    Online
                          pclus4      Online    Online

scal-db_qfs-OraData-rs    pclus1      Online    Online
                          pclus2      Online    Online
                          pclus3      Online    Online
                          pclus4      Online    Online

# clresource status -g rac_server_proxy-rg +

=== Cluster Resources ===

Resource Name           Node Name    State      Status Message
-------------           ---------    -----      --------------
rac_server_proxy-rs     pclus1       Offline    Offline
                        pclus2       Online     Online - Oracle instance UP
                        pclus3       Online     Online - Oracle instance UP
                        pclus4       Online     Online - Oracle instance UP
```

EXAMPLE 6–2 Status of a Faulty RAC Database Resource Group

This example provides the following status information for a RAC database resource group that is faulty:

- The Oracle RAC database on `pclus1` has failed to start. The effects of this failure are as follows:
  - The `rac_server_proxy-rg` resource group is online, but faulted on node `pclus1`.
  - The `rac_server_proxy-rs` resource is offline on node `pclus1`.
- All other multi-master resource groups and the resources that the groups contain are online on all nodes.
- All failover resource groups and the resources that the groups contain are online on their primary nodes and offline on the remaining nodes.

```
# clresourcegroup status +

=== Cluster Resource Groups ===

Group Name             Node Name    Suspended   Status
----------             ---------    ---------   ------
rac-framework-rg       pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

vucmm-framework-rg     pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

scaldg-rg              pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

qfsmds-rg              pclus1       No          Online
                       pclus2       No          Offline
                       pclus3       No          Offline
                       pclus4       No          Offline

scalmnt-rg             pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

rac_server_proxy-rg    pclus1       No          Online faulted
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

# clresource status -g rac_server_proxy-rg +

=== Cluster Resources ===
```

EXAMPLE 6–2 Status of a Faulty RAC Database Resource Group *(Continued)*

```
Resource Name           Node Name    State      Status Message
-------------           ---------    -----      --------------
rac_server_proxy-rs     pclus1       Offline    Offline - Oracle instance DOWN
                        pclus2       Online     Online - Oracle instance UP
                        pclus3       Online     Online - Oracle instance UP
                        pclus4       Online     Online - Oracle instance UP

# clresource status -g rac-framework-rg +

=== Cluster Resources ===

Resource Name         Node Name      State      Status Message
-------------         ---------      -----      --------------
rac-framework-rs      pclus1         Online     Online
                      pclus2         Online     Online
                      pclus3         Online     Online
                      pclus4         Online     Online

rac-udlm-rs           pclus1         Online     Online
                      pclus2         Online     Online
                      pclus3         Online     Online
                      pclus4         Online     Online


crs_framework-rs      pclus1         Online     Online
                      pclus2         Online     Online
                      pclus3         Online     Online
                      pclus4         Online     Online

# clresource status -g vucmm-framework-rg +

=== Cluster Resources ===

Resource Name       Node Name    State          Status Message
-------------       ---------    -----          --------------
vucmm-framework-rs  pclus1       Online         Online
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

vucmm-svm-rs        pclus1       Online         Online
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

# clresource status -g scaldg-rg +

=== Cluster Resources ===

Resource Name       Node Name       State       Status Message
-------------       ---------       -----       --------------
scaloradg-rs        pclus1          Online      Online - Diskgroup online
                    pclus2          Online      Online - Diskgroup online
                    pclus3          Online      Online - Diskgroup online
                    pclus4          Online      Online - Diskgroup online
```

EXAMPLE 6–2 Status of a Faulty RAC Database Resource Group *(Continued)*

```
# clresource status -g qfsmds-rg +

=== Cluster Resources ===

Resource Name            Node Name    State     Status Message
-------------            ---------    -----     --------------
qfs-db_qfs-OraHome-rs    pclus1       Online    Online - Service is online.
                         pclus2       Offline   Offline
                         pclus3       Offline   Offline
                         pclus4       Offline   Offline

qfs-db_qfs-OraData-rs    pclus1       Online    Online - Service is online.
                         pclus2       Offline   Offline
                         pclus3       Offline   Offline
                         pclus4       Offline   Offline

# clresource status -g scalmnt-rg +

=== Cluster Resources ===

Resource Name             Node Name    State    Status Message
-------------             ---------    -----    --------------
scal-db_qfs-OraHome-rs    pclus1       Online   Online
                          pclus2       Online   Online
                          pclus3       Online   Online
                          pclus4       Online   Online

scal-db_qfs-OraData-rs    pclus1       Online   Online
                          pclus2       Online   Online
                          pclus3       Online   Online
                          pclus4       Online   Online
```

EXAMPLE 6–3 Status of an Operational Oracle RAC Configuration

This example shows the status of an Oracle RAC configuration that is operating correctly. The example indicates that the status of resource groups and resources in this configuration is as follows:

- All multi-master resource groups and the resources that the groups contain are online on all nodes.
- All failover resource groups and the resources that the groups contain are online on their primary nodes and offline on the remaining nodes.

```
# clresourcegroup status +

=== Cluster Resource Groups ===

Group Name             Node Name    Suspended   Status
----------             ---------    ---------   ------
rac-framework-rg       pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online
```

EXAMPLE 6–3 Status of an Operational Oracle RAC Configuration *(Continued)*

```
vucmm-framework-rg     pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

scaldg-rg              pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

qfsmds-rg              pclus1       No          Online
                       pclus2       No          Offline
                       pclus3       No          Offline
                       pclus4       No          Offline

scalmnt-rg             pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online

rac_server_proxy-rg    pclus1       No          Online
                       pclus2       No          Online
                       pclus3       No          Online
                       pclus4       No          Online
```

```
# clresource status -g rac-framework-rg +

=== Cluster Resources ===

Resource Name         Node Name      State      Status Message
-------------         ---------      -----      --------------
rac-framework-rs      pclus1         Online     Online
                      pclus2         Online     Online
                      pclus3         Online     Online
                      pclus4         Online     Online

rac-udlm-rs           pclus1         Online     Online
                      pclus2         Online     Online
                      pclus3         Online     Online
                      pclus4         Online     Online

crs_framework-rs      pclus1         Online     Online
                      pclus2         Online     Online
                      pclus3         Online     Online
                      pclus4         Online     Online
```

```
# clresource status -g vucmm-framework-rg +

=== Cluster Resources ===

Resource Name       Node Name    State          Status Message
-------------       ---------    -----          --------------
vucmm-framework-rs  pclus1       Online         Online
                    pclus2       Online         Online
                    pclus3       Online         Online
```

EXAMPLE 6–3 Status of an Operational Oracle RAC Configuration *(Continued)*

```
                    pclus4       Online         Online

vucmm-svm-rs        pclus1       Online         Online
                    pclus2       Online         Online
                    pclus3       Online         Online
                    pclus4       Online         Online

# clresource status -g scaldg-rg +

=== Cluster Resources ===

Resource Name       Node Name       State       Status Message
-------------       ---------       -----       --------------
scaloradg-rs        pclus1          Online      Online - Diskgroup online
                    pclus2          Online      Online - Diskgroup online
                    pclus3          Online      Online - Diskgroup online
                    pclus4          Online      Online - Diskgroup online

# clresource status -g qfsmds-rg +

=== Cluster Resources ===

Resource Name            Node Name    State     Status Message
-------------            ---------    -----     --------------
qfs-db_qfs-OraHome-rs    pclus1       Online    Online - Service is online.
                         pclus2       Offline   Offline
                         pclus3       Offline   Offline
                         pclus4       Offline   Offline

qfs-db_qfs-OraData-rs    pclus1       Online    Online - Service is online.
                         pclus2       Offline   Offline
                         pclus3       Offline   Offline
                         pclus4       Offline   Offline

# clresource status -g scalmnt-rg +

=== Cluster Resources ===

Resource Name             Node Name    State    Status Message
-------------             ---------    -----    --------------
scal-db_qfs-OraHome-rs    pclus1       Online   Online
                          pclus2       Online   Online
                          pclus3       Online   Online
                          pclus4       Online   Online

scal-db_qfs-OraData-rs    pclus1       Online   Online
                          pclus2       Online   Online
                          pclus3       Online   Online
                          pclus4       Online   Online

# clresource status -g rac_server_proxy-rg +

=== Cluster Resources ===

Resource Name           Node Name     State     Status Message
-------------           ---------     -----     --------------
```

EXAMPLE 6–3 Status of an Operational Oracle RAC Configuration *(Continued)*

```
rac_server_proxy-rs     pclus1        Online    Online - Oracle instance UP
                        pclus2        Online    Online - Oracle instance UP
                        pclus3        Online    Online - Oracle instance UP
                        pclus4        Online    Online - Oracle instance UP
```

Sources of Diagnostic Information

If the state of a scalable device group resource or a file-system mount-point resource changes, the new state is logged through the `syslog(3C)` function.

The directories `/var/cluster/ucmm` and `/var/cluster/vucmm` contain the sources of diagnostic information that are shown in the following table.

Source	Location
Log files for previous multiple-owner volume-manager reconfigurations	`/var/cluster/vucmm/vucmm_reconf.log.0` (`0`,`1`,...)
Log file for the current userland cluster membership monitor (UCMM) reconfiguration	`/var/cluster/ucmm/ucmm_reconf.log`
Log files for previous UCMM reconfigurations	`/var/cluster/ucmm/ucmm_reconf.log.0` (`0`,`1`,...)
SPARC: Oracle UNIX Distributed Lock Manager (Oracle UDLM) (Oracle UDLM) core files	`/var/cluster/ucmm/dlm_`*nodename*`/cores` This location is dependent on the Oracle UDLM package. If you cannot find the Oracle log files at this location, contact Oracle support.
SPARC: Log files for Oracle UDLMevents	`/var/cluster/ucmm/dlm_`*nodename*`/logs` This location is dependent on the Oracle UDLM package. If you cannot find the Oracle log files at this location, contact Oracle support.

The directory `/var/opt/SUNWscor/oracle_server/proxy`*resource* contains log files for the resource that represents the Oracle 10*g* Release 2 or 11*g* RAC proxy server. Messages for server-side components and client-side components of the proxy server resource are written to separate files:

- Messages for server-side components are written to the file `message_log.`*resource*.
- Messages for client-side components are written to the file `message_log.client.`*resource*.

In these file names and directory names, *resource* is the name of the resource that represents the Oracle RAC server component.

The directory `/var/opt/SUNWscor/oracle_server` contains log files for the Oracle 9*i* RAC server resource. Each file is named `/var/opt/SUNWscor/oracle_server/message_log.`*resource*.

The system messages file also contains diagnostic information.

If a problem occurs with Support for Oracle RAC, consult these files to obtain information about the cause of the problem.

Common Problems and Their Solutions

The subsections that follow describe problems that can affect Support for Oracle RAC. Each subsection provides information about the cause of the problem and a solution to the problem.

Failure of a RAC Framework Resource Group

This section describes problems that can affect the RAC framework resource group.

Node Panic During Initialization of Support for Oracle RAC

If a fatal problem occurs during the initialization of Support for Oracle RAC, the node panics with an error messages similar to the following error message:

```
panic[cpu0]/thread=40037e60: Failfast: Aborting because "ucmmd" died 30 seconds
ago
```

Description: A component that the UCMM controls returned an error to the UCMM during a reconfiguration.

Cause: The most common causes of this problem are as follows:

- SPARC: The `ORCLudlm` package that contains the Oracle UDLM is not installed.

- SPARC: The version of the Oracle UDLM is incompatible with the version of Support for Oracle RAC.
- SPARC: The amount of shared memory is insufficient to enable the Oracle UDLM to start.

A node might also panic during the initialization of Support for Oracle RAC because a reconfiguration step has timed out. For more information, see "Node Panic Caused by a Timeout" on page 172.

Solution: For instructions to correct the problem, see "How to Recover From a Failure of the `ucmmd` Daemon or a Related Component" on page 167.

Note – When the node is a global-cluster voting node of the global cluster, the node panic brings down the entire machine. When the node is a zone-cluster node, the node panic brings down only that specific zone and other zones remain unaffected.

Failure of the `ucmmd` Daemon to Start

The UCMM daemon, `ucmmd`, manages the reconfiguration of Support for Oracle RAC. When a cluster is booted or rebooted, this daemon is started only after all components of Support for Oracle RAC are validated. If the validation of a component on a node fails, the `ucmmd` daemon fails to start on the node.

The most common causes of this problem are as follows:

- SPARC: The `ORCLudlm` package that contains the Oracle UDLM is not installed.
- An error occurred during a previous reconfiguration of a component Support for Oracle RAC.
- A step in a previous reconfiguration of Support for Oracle RAC timed out, causing the node on which the timeout occurred to panic.

For instructions to correct the problem, see "How to Recover From a Failure of the `ucmmd` Daemon or a Related Component" on page 167.

▼ How to Recover From a Failure of the `ucmmd` Daemon or a Related Component

Perform this task to correct the problems that are described in the following sections:

- "Node Panic During Initialization of Support for Oracle RAC" on page 166
- "Failure of the `ucmmd` Daemon to Start" on page 167

1 **To determine the cause of the problem, examine the log files for UCMM reconfigurations and the system messages file.**

For the location of the log files for UCMM reconfigurations, see "Sources of Diagnostic Information" on page 165.

When you examine these files, start at the most recent message and work backward until you identify the cause of the problem.

For more information about error messages that might indicate the cause of reconfiguration errors, see *Oracle Solaris Cluster Error Messages Guide*.

2 **Correct the problem that caused the component to return an error to the UCMM.**

For example:

- **SPARC: If your Oracle release requires Oracle UDLM and the `ORCLudlm` package that contains the Oracle UDLM is not installed, ensure that the package is installed.**

  Note – Oracle UDLM is required only when it is actually used.

  a. **Ensure that you have completed all the procedures that precede installing and configuring the Oracle UDLM software.**

     The procedures that you must complete are listed in Table 1–1.

  b. **Ensure that the Oracle UDLM software is correctly installed and configured.**

     For more information, see "SPARC: Installing the Oracle UDLM" on page 42.

- **SPARC: If the version of the Oracle UDLM is incompatible with the version of Support for Oracle RAC, install a compatible version of the package.**

  For more information, see "SPARC: Installing the Oracle UDLM" on page 42.

- **SPARC: If the amount of shared memory is insufficient to enable the Oracle UDLM to start, increase the amount of shared memory.**

  For more information, see "How to Configure Shared Memory for the Oracle RAC Software in the Global Cluster" on page 36.

- **If a reconfiguration step has timed out, increase the value of the extension property that specifies the timeout for the step.**

  For more information, see "Node Panic Caused by a Timeout" on page 172.

3 **If the solution to the problem requires a reboot, reboot the node where the problem occurred.**

The solution to only certain problems requires a reboot. For example, increasing the amount of shared memory requires a reboot. However, increasing the value of a step timeout does *not* require a reboot.

For more information about how to reboot a node, see "Shutting Down and Booting a Single Node in a Cluster" in *Oracle Solaris Cluster System Administration Guide*.

4 **On the node where the problem occurred, take offline and bring online the RAC framework resource group.**

This step refreshes the resource group with the configuration changes you made.

a. **Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization.**

b. **Type the command to take offline the RAC framework resource group and its resources.**

```
# clresourcegroup offline -n node rac-fmwk-rg
```

-n *node*	Specifies the node name or node identifier (ID) of the node where the problem occurred.
rac-fmwk-rg	Specifies the name of the resource group that is to be taken offline.

c. **Type the command to bring online and in a managed state the RAC framework resource group and its resources.**

```
# clresourcegroup online -emM -n node rac-fmwk-rg
```

Failure of a Multiple-Owner Volume-Manager Framework Resource Group

This section describes problems that can affect the multiple-owner volume-manager framework resource group.

Node Panic During Initialization of the Multiple-Owner Volume-Manager Framework

If a fatal problem occurs during the initialization of the multiple-owner volume-manager framework, the node panics with an error messages similar to the following error message:

Note – When the node is a global-cluster voting node of the global cluster, the node panic brings down the entire machine.

```
panic[cpu0]/thread=40037e60: Failfast: Aborting because "vucmmd" died 30 seconds
ago
```

Description: A component that the multiple-owner volume-manager framework controls returned an error to the multiple-owner volume-manager framework during a reconfiguration.

Cause: The most common causes of this problem is that the license for Veritas Volume Manager (VxVM) is missing or has expired.

A node might also panic during the initialization of the multiple-owner volume-manager framework because a reconfiguration step has timed out. For more information, see "Node Panic Caused by a Timeout" on page 172.

Solution: For instructions to correct the problem, see "How to Recover From a Failure of the vucmmd Daemon or a Related Component" on page 170.

Failure of the `vucmmd` Daemon to Start

The multiple-owner volume-manager framework daemon, `vucmmd`, manages the reconfiguration of the multiple-owner volume-manager framework. When a cluster is booted or rebooted, this daemon is started only after all components of the multiple-owner volume-manager framework are validated. If the validation of a component on a node fails, the `vucmmd` daemon fails to start on the node.

The most common causes of this problem are as follows:

- An error occurred during a previous reconfiguration of a component of the multiple-owner volume-manager framework.
- A step in a previous reconfiguration of the multiple-owner volume-manager framework timed out, causing the node on which the timeout occurred to panic.

For instructions to correct the problem, see "How to Recover From a Failure of the vucmmd Daemon or a Related Component" on page 170.

▼ How to Recover From a Failure of the `vucmmd` Daemon or a Related Component

Perform this task to correct the problems that are described in the following sections:

- "Node Panic During Initialization of the Multiple-Owner Volume-Manager Framework" on page 169
- "Failure of the vucmmd Daemon to Start" on page 170

1 **To determine the cause of the problem, examine the log files for multiple-owner volume-manager framework reconfigurations and the system messages file.**

For the location of the log files for multiple-owner volume-manager framework reconfigurations, see "Sources of Diagnostic Information" on page 165.

When you examine these files, start at the most recent message and work backward until you identify the cause of the problem.

For more information about error messages that might indicate the cause of reconfiguration errors, see *Oracle Solaris Cluster Error Messages Guide*.

2 **Correct the problem that caused the component to return an error to the multiple-owner volume-manager framework .**

For example:

- **If the license for VxVM is missing or has expired, ensure that VxVM is correctly installed and licensed.**

  a. **Verify that you have correctly installed your volume manager packages.**

  b. **If you are using VxVM, check that you have installed the software and check that the license for the VxVM cluster feature is valid.**

  Note – A zone cluster does not support VxVM.

- **If a reconfiguration step has timed out, increase the value of the extension property that specifies the timeout for the step.**

  For more information, see "Node Panic Caused by a Timeout" on page 172.

3 **If the solution to the problem requires a reboot, reboot the node where the problem occurred.**

The solution to only certain problems requires a reboot. For example, increasing the amount of shared memory requires a reboot. However, increasing the value of a step timeout does *not* require a reboot.

For more information about how to reboot a node, see "Shutting Down and Booting a Single Node in a Cluster" in *Oracle Solaris Cluster System Administration Guide*.

4 **On the node where the problem occurred, take offline and bring online the multiple-owner volume-manager framework resource group.**

This step refreshes the resource group with the configuration changes you made.

a. **Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization.**

b. Type the command to take offline the multiple-owner volume-manager framework resource group and its resources.

```
# clresourcegroup offline -n node vucmm-fmwk-rg
```

`-n` *node*	Specifies the node name or node identifier (ID) of the node where the problem occurred.
vucmm-fmwk-rg	Specifies the name of the resource group that is to be taken offline.

c. Type the command to bring online and in a managed state the multiple-owner volume-manager framework resource group and its resources.

```
# clresourcegroup online -emM -n node vucmm-fmwk-rg
```

SUNW.qfs Registration Fails Because the Registration File Is Not Found

Oracle Solaris Cluster resource-type registration files are located in the `/opt/cluster/lib/rgm/rtreg/` or `/usr/cluster/lib/rgm/rtreg/` directory. The `SUNW.qfs` resource-type registration file is located in the `/opt/SUNWsamfs/sc/etc/` directory.

If Oracle Solaris Cluster software is already installed when you install Sun QFS software, the necessary mapping to the `SUNW.qfs` registration file is automatically created. But if Oracle Solaris Cluster software is not already installed when you install Sun QFS software, the necessary mapping to the `SUNW.qfs` registration file is not made, even when Sun Cluster software is later installed. Attempts to register the `SUNW.qfs` resource type therefore fail because the Oracle Solaris Cluster software is unaware of the location of its registration file.

To enable Oracle Solaris Cluster software to locate the `SUNW.qfs` resource type, create a symbolic link to the directory:

```
# cd /usr/cluster/lib/rgm/rtreg
# ln -s /opt/SUNWsamfs/sc/etc/SUNW.qfs SUNW.qfs
```

Node Panic Caused by a Timeout

The timing out of any step in the reconfiguration of Support for Oracle RAC causes the node on which the timeout occurred to panic.

To prevent reconfiguration steps from timing out, tune the timeouts that depend on your cluster configuration. For more information, see "Guidelines for Setting Timeouts" on page 136.

If a reconfiguration step times out, use the Oracle Solaris Cluster maintenance commands to increase the value of the extension property that specifies the timeout for the step. For more information, see Appendix C, "Support for Oracle RAC Extension Properties."

After you have increased the value of the extension property, bring online the RAC framework resource group on the node that panicked.

Failure of a `SUNW.rac_framework` or `SUNW.vucmm_framework` Resource to Start

If a `SUNW.rac_framework` or `SUNW.vucmm_framework`resource fails to start, verify the status of the resource to determine the cause of the failure. For more information, see "How to Verify the Status of Support for Oracle RAC" on page 155.

The state of a resource that failed to start is shown as `Start failed`. The associated status message indicates the cause of the failure to start.

This section contains the following information:

`SUNW.rac_framework` Failure-to-Start Status Messages

The following status messages are associated with the failure of a `SUNW.rac_framework` resource to start:

`Faulted - ucmmd is not running`
: **Description:** The `ucmmd` daemon is not running on the node where the resource resides.

  Solution: For information about how to correct this problem, see "Failure of the `ucmmd` Daemon to Start" on page 167.

`Degraded - reconfiguration in progress`
: **Description:** The UCMM is undergoing a reconfiguration. This message indicates a problem only if the reconfiguration of the UCMM is not completed and the status of this resource persistently remains degraded.

  Cause: If this message indicates a problem, the cause of the failure is a configuration error in one or more components of Support for Oracle RAC.

  Solution: The solution to this problem depends on whether the message indicates a problem:

  - If the message indicates a problem, correct the problem as explained in "How to Recover From a Failure of the `ucmmd` Daemon or a Related Component" on page 167.
  - If the message does not indicate a problem, no action is required.

`Online`

Description: Reconfiguration of Oracle RAC was not completed until after the `START` method of the `SUNW.rac_framework` resource timed out.

Solution: For instructions to correct the problem, see "How to Recover From the Timing Out of the START Method" on page 175.

`SUNW.vucmm_framework` Failure-to-Start Status Messages

The following status messages are associated with the failure of a `SUNW.vucmm_framework` resource to start:

`Faulted - vucmmd is not running`

Description: The `vucmmd` daemon is not running on the node where the resource resides.

Solution: For information about how to correct this problem, see "Failure of the vucmmd Daemon to Start" on page 170.

`Degraded - reconfiguration in progress`

Description: The multiple-owner volume-manager framework is undergoing a reconfiguration. This message indicates a problem only if the reconfiguration of the multiple-owner volume-manager framework is not completed and the status of this resource persistently remains degraded.

Cause: If this message indicates a problem, the cause of the failure is a configuration error in one or more components of the volume manager reconfiguration framework.

Solution: The solution to this problem depends on whether the message indicates a problem:

- If the message indicates a problem, correct the problem as explained in "How to Recover From a Failure of the vucmmd Daemon or a Related Component" on page 170.
- If the message does not indicate a problem, no action is required.

`Online`

Description: Reconfiguration of Oracle RAC was not completed until after the `START` method of the `SUNW.vucmm_framework` resource timed out.

Solution: For instructions to correct the problem, see "How to Recover From the Timing Out of the START Method" on page 175.

▼ How to Recover From the Timing Out of the START Method

1 **Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization.**

2 **On the node where the `START` method timed out, take offline the framework resource group that failed to start.**

To perform this operation, switch the primary nodes of the resource group to the other nodes where the group is online.

`# clresourcegroup offline -n` *nodelist resource-group*

`-n` *nodelist* — Specifies a comma-separated list of other cluster nodes on which *resource-group* is online. Omit from this list the node where the `START` method timed out.

resource-group — Specifies the name of the framework resource group.

If your configuration uses both a multiple-owner volume-manager framework resource group and a RAC framework resource group, first take offline the multiple-owner volume-manager framework resource group. When the multiple-owner volume-manager framework resource group is offline, then take offline the RAC framework resource group.

If the RAC resource group was created by using the `clsetup` utility, the name of the resource group is `rac-framework-rg`.

3 **On all cluster nodes that can run Support for Oracle RAC, bring online the framework resource group that failed to come online.**

`# clresourcegroup online` *resource-group*

resource-group — Specifies that the resource group that you brought offline in Step 2 is to be moved to the `MANAGED` state and brought online.

Failure of a Resource to Stop

If a resource fails to stop, correct this problem as explained in "Clearing the STOP_FAILED Error Flag on Resources" in *Oracle Solaris Cluster Data Services Planning and Administration Guide*.

◆ ◆ ◆ CHAPTER 7

Modifying an Existing Configuration of Support for Oracle RAC

This chapter explains how to modify an existing configuration of Support for Oracle RAC.

Overview of Tasks for Modifying an Existing Configuration of Support for Oracle RAC

Table 7–1 summarizes the administration tasks for Support for Oracle RAC.

Perform these tasks whenever they are required.

TABLE 7–1 Tasks for Modifying an Existing Configuration of Support for Oracle RAC

Task	Instructions
Modifying online the resource for a scalable device group	"Modifying Online the Resource for a Scalable Device Group" on page 178
Extend an existing configuration of Support for Oracle RAC	"Extending an Existing Configuration of Support for Oracle RAC" on page 179
Migrate a legacy RAC configuration to use a multiple-owner volume-manager framework resource group	"Migrating a Volume-Manager Resource From a RAC Framework Resource Group to a Multiple-Owner Volume Manager Framework Resource Group" on page 189

TABLE 7–1 Tasks for Modifying an Existing Configuration of Support for Oracle RAC *(Continued)*

Task	Instructions
SPARC: (Oracle 11*g* release 2 only) Migrate an existing configure from Oracle UDLM to native SKGXN or vice versa.	"SPARC: Deploying Oracle Solaris Cluster Native SKGXN for Oracle RAC 11*g* Release 2" on page 192
Remove Support for Oracle RAC	"Removing Support for Oracle RAC" on page 196

Modifying Online the Resource for a Scalable Device Group

Modifying online the resource for a scalable device group involves changing the list of logical volumes that are to be monitored. The `LogicalDeviceList` extension property of the `SUNW.ScalDeviceGroup` resource type specifies the list of logical volumes in a global device group that are to be monitored.

▼ How to Modify Online the Resource for a Scalable Device Group

1 Become superuser or assume a role that provides `solaris.cluster.modify` RBAC authorization.

2 Modify the `LogicalDeviceList` extension property of the `ScalDeviceGroup` resource.

- **To add device groups to a `ScalDeviceGroup` resource, type the following command:**

  `# clresource set -p LogicalDeviceList+=`*logical-device-listscal-mp-rs*

  The addition of the logical volume is effective immediately.

- **To remove device groups from a `ScalDeviceGroup` resource, type the following command:**

  `# clresource set -p LogicalDeviceList-=`*logical-device-listscal-mp-rs*

  The removal of the logical volume is effective immediately.

Extending an Existing Configuration of Support for Oracle RAC

Extend an existing configuration of Support for Oracle RAC in any of the following situations:

- You are adding nodes to a cluster and you require Support for Oracle RAC to run on the nodes. See "How to Add Support for Oracle RAC to Selected Nodes " on page 179.
- You are adding a volume manager. See "How to Add a Volume Manager Resource to the Framework Resource Group" on page 186.

▼ How to Add Support for Oracle RAC to Selected Nodes

Perform this procedure if you are adding nodes to a cluster and you require Support for Oracle RAC to run on the nodes. Perform this procedure from only one node.

This task involves adding the selected nodes from the following resource groups in the following order:

- Resource groups for scalable file-system mount-point resources
- The multiple-owner volume-manager framework resource group, if used.
- The RAC framework resource group
- Resource groups for scalable device group resources
- Resource groups that contain resources for the Sun QFS metadata server
- Resource groups for logical hostname resources
- The resource group for the Oracle RAC database

Before You Begin

- Ensure that the required Support for Oracle RAC software packages are installed on each node to which you are adding Support for Oracle RAC. For more information, see "Installing the Support for Oracle RAC Packages" on page 40.
- Ensure that the node that you add is connected to the shared storage that is used by the Oracle RAC configuration.

1 Become superuser on any cluster node.

2 Add the nodes to any resource groups that contain scalable file-system mount-point resources.

If no resource groups that contain scalable file-system mount-point resources are configured, omit this step.

For each resource group to which you are adding nodes, run the following command:

```
# clresourcegroup add-node -S -n nodelist scal-mp-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes to which you are adding Support for Oracle RAC.

scal-mp-rg
Specifies the name of the resource group to which you are adding nodes.

3 Add the nodes to the multiple-owner volume-manager framework resource group, if used.

```
# clresourcegroup add-node -S -n nodelist vucmm-fmwk-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes to which you are adding the resource group.

vucmm-fmwk-rg
Specifies the name of the resource group to which you are adding nodes.

4 Add the nodes to the RAC framework resource group.

```
# clresourcegroup add-node -S -n nodelist rac-fmwk-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes to which you are adding Support for Oracle RAC.

rac-fmwk-rg
Specifies the name of the resource group to which you are adding nodes.

5 Add the nodes to any scalable device groups that you are using for Oracle files.

If you are not using any scalable device groups for Oracle files, omit this step.

How to perform this step depends on the type of the scalable device group.

- **For each Solaris Volume Manager for Sun Cluster multi-owner disk set, type the following command:**

  ```
  # metaset -s set-name -M -a -h nodelist
  ```

  -s *set-name*
  Specifies the Solaris Volume Manager for Sun Cluster multi-owner disk set to which you are adding nodes.

  -h *nodelist*
  Specifies a *space-separated* list of cluster nodes that you are adding to the multi-owner disk set.

- **For each VxVM shared-disk group, use Veritas commands to add the nodes to the VxVM shared-disk group.**

  For more information, see your VxVM documentation.

6 Add the nodes to any resource groups that contain scalable device group resources.

If no resource groups that contain scalable device group resources are configured, omit this step.

For each resource group to which you are adding nodes, run the following command:

```
# clresourcegroup add-node -S -n nodelist scal-dg-rg
```

`-n` *nodelist*
Specifies a comma-separated list of cluster nodes to which you are adding Support for Oracle RAC.

scal-dg-rg
Specifies the name of the resource group to which you are adding nodes.

7 Mount each shared file system that is to be accessed from the nodes that you are adding.

If no shared file systems are to be accessed from the nodes that you are adding, omit this step.

For each file system that you are mounting, type the following command:

```
# mount mount-point
```

mount-point Specifies the mount point of the file system that you are mounting.

8 Add the nodes to any resource groups that contain resources for the Sun QFS metadata server.

If no resource groups that contain resources for the Sun QFS metadata server are configured, omit this step.

For each resource group to which you are adding nodes, run the following command:

```
# clresourcegroup add-node -n nodelist qfs-mds-rg
```

`-n` *nodelist*
Specifies a comma-separated list of cluster nodes to which you are adding Support for Oracle RAC.

qfs-mds-rg
Specifies the name of the resource group to which you are adding nodes.

9 Bring online all resource groups to which you added nodes in Step 6.

These resource groups contain scalable device group resources.

If no resource groups that contain scalable device group resources are configured, omit this step.

For each resource group that you are bringing online, type the following command:

```
# clresourcegroup online scal-dg-rg
```

scal-dg-rg Specifies the name of the resource group that you are bringing online.

10 (Oracle 10*g* release 2 or 11*g* only) Start the Oracle Clusterware.

If you are using Oracle 9*i* or Oracle 10*g* release 1, omit this step.

```
# /etc/init.d/init.crs start
Startup will be queued to init within 30 seconds.
```

11 (Oracle 9*i* only) Add the nodes to all resource groups that contain logical hostname resources for each Oracle RAC database that is to run on the nodes.

If you are using Oracle 10*g* release 1, 10*g* release 2, or 11*g*, omit this step. For Oracle 10*g* release 1, 10*g* release 2, or 11*g*, no resource groups for logical hostname resources are configured.

For each resource group to which you are adding nodes, run the following command:

```
# clresourcegroup add-node -n nodelist lh-rg
```

`-n` *nodelist* Specifies a comma-separated list of cluster nodes to which you are adding Support for Oracle RAC.

lh-rg Specifies the name of the resource group to which you are adding nodes.

12 (Oracle 9*i*, 10*g* release 2, or 11*g* only) Add the nodes to the resource group for each Oracle RAC database that is to run on the nodes.

If you are using Oracle 10*g* release 1, omit this step. For Oracle 10*g* release 1, no resource groups for Oracle RAC databases are configured.

For each resource group to which you are adding nodes, run the following command:

```
# clresourcegroup add-node -S -n nodelist rac-db-rg
```

`-n` *nodelist* Specifies a comma-separated list of cluster nodes to which you are adding Support for Oracle RAC.

rac-db-rg Specifies the name of the resource group to which you are adding nodes.

13 (Oracle 10*g* release 2 or 11*g* only) For each node that you are adding, create the Oracle Clusterware resources that are required to represent Oracle Solaris Cluster resources.

Create an Oracle Clusterware resource for each Oracle Solaris Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend. For more information, see "How to Create an Oracle Clusterware Resource for Interoperation With Sun Cluster" on page 335.

14 (Oracle 9*i*, 10*g* release 2, or 11*g* only) Modify each resource for Oracle RAC databases to set a value of each per-node property for each node that you are adding.

If you are using Oracle 10*g* release 1, omit this step. For Oracle 10*g* release 1, no resource groups for Oracle RAC databases are configured.

For each resource that you are modifying, perform these steps:

a. Disable the resource.

```
# clresource disable rac-db-rs
```

rac-db-rs Specifies the name of the RAC database resource that you are disabling.

b. Set a value of each per-node property for each node that you are adding.

The per-node properties of each resource type for Oracle RAC databases are shown in the following table.

Resource Type	Properties
SUNW.scalable_rac_server_proxy	oracle_sid
SUNW.scalable_rac_listener	listener_name
SUNW.scalable_rac_server	alert_log_file
	oracle_sid

For information about extension properties of resource types for Oracle RAC databases, see the following sections:

- "SUNW.scalable_rac_server_proxy Extension Properties" on page 287
- "SUNW.scalable_rac_listener Extension Properties" on page 281
- "SUNW.scalable_rac_server Extension Properties" on page 282

```
# clresource set \
-p property{node}=value[...] \
[-p property{node}=value[...]][...] \
rac-db-rs
```

property Specifies the name of a per-node property that you are setting.

node Specifies the node for which you are setting a value for *property*.

value Specifies the value to which you are setting *property* for *node*.

rac-db-rs Specifies the name of the RAC database resource whose per-node properties you are setting.

c. Enable the resource.

```
# clresource enable rac-db-rs
```

rac-db-rs Specifies the name of the RAC database resource that you are enabling.

15 (Oracle 9*i*, 10*g* release 2, or 11*g* only) Bring online each resource group for Oracle RAC databases.

If you are using Oracle 10*g* release 1, omit this step. For Oracle 10*g* release 1, no resource groups for Oracle RAC databases are configured.

For each resource group that you are bringing online, run the following command:

```
# clresourcegroup online rac-db-rg
```

rac-db-rg Specifies the name of the resource group that you are bringing online.

Example 7–1 Adding Support for Oracle RAC to Selected Nodes

This example shows the sequence of operations that is required to add Support for Oracle RAC to nodes `pclus3` and `pclus4` of a four-node cluster.

The configuration of Support for Oracle RAC in this example is as follows:

- The version of Oracle RAC is 10*g* Release 2.
- Sun QFS shared file system on Solaris Volume Manager for Sun Cluster is used to store Oracle files.
- The mount points of file systems that are used for Oracle files are as follows:
  - Oracle database files: `/db_qfs/OraData`
  - Oracle binary files and related files: `/db_qfs/OraHome`
- The `oradg` disk set is used only by the Oracle RAC database.
- The name of the Oracle RAC database is `swb`.
- The Sun QFS shared file systems use a Solaris Volume Manager for Sun Cluster multi-owner disk set that is named `oradg`. The creation of this disk set is shown in Example 3–1.
- The configuration uses a multiple-owner volume-manager framework resource group.

The configuration of resource groups in this example is shown in the following table.

Resource Group	Purpose
`vucmm-framework-rg`	Multiple-owner volume-manager resource group.
`rac-framework-rg`	RAC framework resource group.
`scaldg-rg`	Resource group for scalable device-group resources.
`qfsmds-rg`	Resource group for Sun QFS metadata server resources.
`scalmnt-rg`	Resource group for scalable file-system mount-point resources.
`rac_server_proxy-rg`	Oracle RAC database resource group.

The resource groups that are required for this configuration are shown in Figure A–2.

1. To add the nodes to the resource group that contains scalable file-system mount-point resources, the following command is run:

   ```
   # clresourcegroup add-node -S -n pclus3,pclus4 scalmnt-rg
   ```

2. To add the nodes to the multiple-owner volume-manager framework resource group, the following command is run:

   ```
   # clresourcegroup add-node -S -n pclus3,plcus4 vucmm-framework-rg
   ```

3. To add the nodes to the RAC framework resource group, the following command is run:

```
# clresourcegroup add-node -S -n pclus3,plcus4 rac-framework-rg
```

4. To add the nodes to the Solaris Volume Manager for Sun Cluster multi-owner disk set `oradg`, the following command is run:

```
# metaset -s oradg -M -a -h  pclus3 pclus4
```

5. To add the nodes to the resource group that contains scalable device group resources, the following command is run:

```
# clresourcegroup add-node -S -n pclus3,pclus4 scaldg-rg
```

6. To mount the shared file systems that are to be accessed from the nodes that are being added, the following commands are run:

```
# mount /db_qfs/OraData
# mount /db_qfs/OraHome
```

7. To add the nodes to the resource group that contains resources for the Sun QFS metadata server, the following command is run:

```
# clresourcegroup add-node -n pclus3,pclus4 qfsmds-rg
```

8. To bring online the resource group that contains scalable device group resources, the following command is run:

```
# clresourcegroup online scaldg-rg
```

9. To start the Oracle Clusterware and to verify the correct startup of the Oracle Clusterware, the following commands are run:

```
# /etc/init.d/init.crs start
Startup will be queued to init within 30 seconds.
# /db_qfs/OraHome/crs/bin/crsctl check crs
CSS appears healthy
CRS appears healthy
EVM appears healthy
```

10. To add the nodes to the resource group for the Oracle RAC database, the following command is run:

```
# clresourcegroup add-node -S -n pclus3,pclus4 rac_server_proxy-rg
```

After the addition of the nodes to the resource group for the Oracle RAC database, the required Oracle Clusterware resources are created. The creation of these Oracle Clusterware resources is beyond the scope of this example.

11. To set required per-node properties for the RAC database resource, the following commands are run:

```
# clresource disable rac_server_proxy-rs
# clresource set -p oracle_sid\{3\}=swb3 -p \
oracle_sid\{4\}=swb4 rac_server_proxy-rs
# clresource enable rac_server_proxy-rs
```

The per-node property `oracle_sid` is set to `swb3` on node `pclus3` and to `swb4` on node `pclus4`.

12. To bring online the resource group for the Oracle RAC database, the following command is run:

```
# clresourcegroup online rac_server_proxy-rg
```

How to Add a Volume Manager Resource to the Framework Resource Group

Perform this task if you are adding a volume manager to an existing configuration of Support for Oracle RAC. The framework resource group must contain a resource that represents the volume manager that you are adding. You can add a volume manager resource only if the framework resource is disabled and if the framework daemon is stopped on all cluster nodes.

- If the cluster contains a `SUNW.vucmm_framework` based resource group, you add an instance of the `SUNW.vucmm_svm` or `SUNW.vucmm_cvm` resource type to that resource group.

  Do not add an instance of the `SUNW.rac_svm` or `SUNW.rac_cvm` resource type to the `SUNW.rac_framework` based resource group when a `SUNW.vucmm_framework` based resource group exists in the cluster.
- If the cluster does not contain a `SUNW.vucmm_framework` based resource group, you add an instance of the `SUNW.rac_svm` or `SUNW.rac_cvm` resource type to the `SUNW.rac_framework` based resource group.

Caution – This task requires downtime because you must disable the framework resource and reboot the nodes where Oracle RAC is running.

Before You Begin Ensure that the volume manager for which you are adding a resource is installed and configured on all nodes where Oracle RAC is to run.

1. **Become superuser on any cluster node.**

2. **Disable the framework resource in the framework resource group and any other resources that depend on this resource.**

   ```
   # clresource disable -r fmwk-rs
   ```

   fmwk-rs Specifies the name of the resource of type `SUNW.vucmm_framework` or `SUNW.rac_framework` that you are disabling.

3. **Reboot all the nodes that are in the node list of the framework resource group.**

4 **Register and add an instance of the resource type that represents the volume manager that you are adding.**

- **If you are adding Solaris Volume Manager for Sun Cluster, register and add the instance as follows:**

  a. **Register the Solaris Volume Manager for Sun Cluster resource type.**

  - **For a `SUNW.vucmm_framework` based resource group, register the `SUNW.vucmm_svm` resource type.**

    ```
    # clresourcetype register SUNW.vucmm_svm
    ```

  - **For a `SUNW.rac_framework` based resource group, register the `SUNW.rac_svm` resource type.**

    ```
    # clresourcetype register SUNW.rac_svm
    ```

  b. **Add an instance of the Solaris Volume Manager for Sun Cluster resource type to the framework resource group.**

  Ensure that this instance depends on the resource that you disabled in Step 2.

  ```
  # clresource create -g fmwk-rg \
    -t svm-rt \
    -p resource_dependencies=fmwk-rs  svm-rs
  ```

  `-g` *fmwk-rg*
  Specifies the name of the framework resource group. This resource group contains the resource of type `SUNW.vucmm_framework` or `SUNW.rac_framework` that you disabled in Step 2.

  svm-rt
  Specifies the name of the Solaris Volume Manager for Sun Cluster resource type.

  `-p resource_dependencies=`*fmwk-rs*
  Specifies that this instance depends on the resource that you disabled in Step 2.

  svm-rs
  Specifies the name that you are assigning to the resource of type `SUNW.vucmm_svm` or `SUNW.rac_svm`.

- **SPARC: If you are adding VxVM with the cluster feature, register and add the instance as follows.**

  a. **Register the VxVM volume manager resource type.**

  - **For a `SUNW.vucmm_framework` based resource group, register the `SUNW.vucmm_cvm` resource type.**

    ```
    # clresourcetype register SUNW.vucmm_cvm
    ```

- **For a `SUNW.rac_framework` based resource group, register the `SUNW.rac_cvm` resource type.**

  ```
  # clresourcetype register SUNW.rac_cvm
  ```

b. Add an instance of the VxVM volume manager resource type to the resource group that you disabled in Step 2.

Ensure that this instance depends on the resource that you disabled in Step 2.

```
# clresource create -g fmwk-rg \
  -t cvm-rt \
  -p resource_dependencies=fmwk-rs cvm-rs
```

`-g` *fmwk-rg*
: Specifies the name of the framework resource group. This resource group contains the resource that you disabled in Step 2.

cvm-rt
: Specifies the name of the Solaris Volume Manager for Sun Cluster resource type.

`-p resource_dependencies=`*fmwk-rs*
: Specifies that this instance depends on the resource that you disabled in Step 2.

cvm-rs
: Specifies the name that you are assigning to the resource of type `SUNW.vucmm_cvm` or `SUNW.rac_cvm`.

5 Bring online and in a managed state the framework resource group and its resources.

```
# clresourcegroup online -emM fmwk-rg
```

fmwk-rg Specifies that the framework resource group is to be moved to the `MANAGED` state and brought online. This resource group contains the resource that you disabled in Step 2.

Next Steps The next step depends on the volume manager that you are adding, as shown in the following table.

Volume Manager	Next Step
Solaris Volume Manager for Sun Cluster	"How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database" on page 75
SPARC: VxVM with the cluster feature	"How to Create a VxVM Shared-Disk Group for the Oracle RAC Database" on page 81

Migrating a Volume-Manager Resource From a RAC Framework Resource Group to a Multiple-Owner Volume Manager Framework Resource Group

Starting in the Sun Cluster 3.2 11/09 release, a new set of resource types manage multiple-owner volume manager resources in an Oracle RAC configuration. A resource that is based on the `SUNW.vucmm_svm` or `SUNW.vucmm_cvm` resource type is configured in a resource group that is based on the multiple-owner volume-manager framework resource type, `SUNW.vucmm_framework`. The `SUNW.rac_framework` resource group continues to contain other RAC resources, such as Oracle Clusterware and UDLM.

The `SUNW.vucmm_framework` resource type is a single-instance resource type. You can create only one resource of this type in the cluster.

▼ How to Migrate a Volume Manager Resource From a RAC Framework Resource Group to a Multiple-Owner Volume-Manager Framework Resource Group

Perform this procedure to use the multiple-owner volume manager framework to manage volume-manager resources in the Oracle RAC configuration.

1 Become superuser on any cluster node.

2 Create a scalable multiple-owner volume-manager framework resource group.

```
# clresourcegroup create -n nodelist -S vucmm-fmwk-rg
```

`-n nodelist=`*nodelist*
Specifies a comma-separated list of cluster nodes on which Support for Oracle RAC is to be enabled. The Support for Oracle RAC software packages must be installed on each node in this list.

Note – This node list must contain all nodes that are configured in the node list of the RAC framework resource group.

---

vucmm-fmwk-rg
Specifies the name that you are assigning to the resource group.

3 Register the `SUNW.vucmm_framework` resource type.

```
# clresourcetype register SUNW.vucmm_framework
```

4 Add an instance of the `SUNW.vucmm_framework` resource type to the resource group that you created in Step 2.

```
# clresource create -g vucmm-fmwk-rg -t SUNW.vucmm_framework vucmm-fmwk-rs
```

vucmm-fmwk-rs Specifies the name that you are assigning to the `SUNW.vucmm_framework` resource.

5 Set the `reservation_timeout` property of the volume-manager resource type.

Set the property to the same value as in the `SUNW.rac_framework` resource group.

a. Display the value of the `reservation_timeout` extension property for the `SUNW.rac_framework` resource type.

```
# clresource show -p reservation_timeout -t resource-type
```

resource-type
Specifies the resource type of the resource in the RAC resource group for which the `reservation_timeout` extension property is set. This resource type is `SUNW.rac_svm` or `SUNW.rac_cvm`.

b. Set the `reservation_timeout` extension property of the `SUNW.vucmm_framework` resource type.

```
# clresource set -p type_version=version \
-p reservation_timeout=timeout vucmm-framework-rs
```

version
Specifies the value of the `type_version` property for the version of `SUNW.rac_framework` to which you are migrating the instance.

timeout
Specifies the value to which you are setting the `reservation_timeout` extension property.

vucmm-framework-rs
Specifies the name of the resource of type `SUNW.vucmm_framework` on your cluster.

6 Register and add an instance of the resource type that represents the volume manager that you are using for Oracle files, if any.

- **If you are using Solaris Volume Manager for Sun Cluster, register and add the instance as follows:**

  a. Register the `SUNW.vucmm_svm` resource type.

  ```
  # clresourcetype register SUNW.vucmm_svm
  ```

b. Add an instance of the `SUNW.vucmm_svm` resource type to the resource group that you created in Step 2.

Ensure that this instance depends on the `vucmm_framework` resource that you created in Step 4.

```
# clresource create -g vucmm-fmwk-rg \
  -t SUNW.vucmm_svm \
  -p resource_dependencies=vucmm-fmwk-rs  vucmm-svm-rs
```

`-p resource_dependencies=`*vucmm-fmwk-rs*
Specifies that this instance depends on the `SUNW.vucmm_framework` resource that you created in Step 4.

vucmm-svm-rs
Specifies the name that you are assigning to the `SUNW.vucmm_svm` resource.

- **SPARC: If you are using VxVM with the cluster feature, register and add the instance as follows.**

a. Register the `SUNW.vucmm_cvm` resource type.

```
# clresourcetype register SUNW.vucmm_cvm
```

b. Add an instance of the `SUNW.vucmm_cvm` resource type to the resource group that you created in Step 2.

Ensure that this instance depends on the `vucmm_framework` resource that you created in Step 4.

```
# clresource create -g vucmm-fmwk-rg \
  -t SUNW.vucmm_cvm \
  -p resource_dependencies=vucmm-fmwk-rs  vucmm-cvm-rs
```

`-p resource_dependencies=`*vucmm-fmwk-rs*
Specifies that this instance depends on the `SUNW.vucmm_framework` resource that you created in Step 4.

vucmm-cvm-rs
Specifies the name that you are assigning to the `SUNW.vucmm_cvm` resource.

7 Verify the configuration of the multiple-owner volume-manager framework resource group.

```
# clresourcegroup show vucmm-fmwk-rg
```

8 Verify that the multiple-owner volume-manager framework resource group and its resources are online.

```
# clresourcegroup status
```

9 Bring online and in a managed state the multiple-owner volume-manager framework resource group and its resources.

```
# clresourcegroup online -emM vucmm-fmwk-rg
```

vucmm-fmwk-rg Specifies the name of the `SUNW.vucmm_framework` based resource group.

10 If your Oracle RAC configuration includes a `ScalDeviceGroup` resource that depends on a RAC volume manager resource, change the dependency to the equivalent multiple-owner volume-manager resource.

```
# clresource set -p resource_dependencies=vucmm-vol-mgr-rs{local_node} scal-dg-rs
```

vucmm-vol-mgr-rs
Specifies the name of the multiple-owner volume-manager resource.

- For Solaris Volume Manager for Sun Cluster, use the `SUNW.vucmm_svm` resource type.
- For VxVM with the cluster feature, use the `SUNW.vucmm_cvm` resource type.

scal-dg-rs
Specifies the `SUNW.ScalDeviceGroup` resource whose dependency you move to the *vucmm-vol-mgr-rs* resource.

11 Disable the RAC volume manager resource.

```
# clresource disable rac-vol-mgr-rs
```

rac-vol-mgr-rs
Specifies the `SUNW.rac_svm` or `SUNW.rac_cvm` resource that is used by the `SUNW.rac_framework` based resource group.

12 Delete the RAC volume manager resource from the RAC framework resource group.

```
# clresource delete -t rac-vol-mgr-rs rac-fmwk-rg
```

rac-fmwk-rg Specifies the name of the `SUNW.rac_framework` based resource group.

13 Verify that all resource groups for RAC are online.

```
# clresourcegroup status
```

14 Reboot each node, one node at a time.

The reboot refreshes the RAC framework with the configuration changes.

SPARC: Deploying Oracle Solaris Cluster Native SKGXN for Oracle RAC 11*g* Release 2

Oracle RAC specifies a set of APIs for distributed process monitoring and cluster configuration services. This set of APIs is known as System Kernel Generic Interface Node membership (SKGXN). Oracle Solaris Cluster and other cluster software use this API to communicate with Oracle RAC.

In previous releases of Sun Cluster software, configurations with Oracle RAC on a SPARC machine implement SKGXN through the Oracle UDLM package that is provided by Oracle. In this release of Oracle Solaris Cluster software for SPARC, with Oracle RAC 11*g* release 2 you can deploy the Oracle Solaris Cluster native SKGXN instead of Oracle UDLM. This section provides information about how to deploy Oracle Solaris Cluster native SKGXN on an existing Oracle RAC 11*g* release 2 configuration.

To configure a new Oracle RAC 11*g* release 2 configuration to use native SKGXN, follow the procedures in this manual without adding an Oracle UDLM resource to the RAC framework resource group. Native SKGXN is automatically used when the RAC framework resource group does not contain an Oracle UDLM resource.

- "SPARC: How to Prepare the Cluster to Change the SKGXN Interfaces" on page 193
- "SPARC: How to Convert From Oracle UDLM to Oracle Solaris Cluster Native SKGXN" on page 194
- "SPARC: How to Convert SKGXN from Oracle Solaris Cluster Native SKGXN to Oracle UDLM" on page 195
- "SPARC: How to Bring Oracle RAC Online After Switching the SKGXN" on page 196

SPARC: How to Prepare the Cluster to Change the SKGXN Interfaces

(Oracle 11*g* release 2 only) Perform this procedure to prepare a cluster to change the SKGXN interface.

Note – Migrating from one implementation of SKGXN to the other requires downtime for the Oracle RAC data service. You should plan downtime to perform this migration.

---

1 Become superuser.

2 Upgrade the RAC framework resource type to the latest installed version.

```
# grep -i RT_VERSION /usr/cluster/lib/rgm/SUNW.rac_framework
RT_VERSION = "N";
# clresourcetype register SUNW.rac_framework:N
```

Alternatively, to upgrade the resource type in the global zone, use Oracle Solaris Cluster Manager or the Resource Group option of the `clsetup` utility.

3 **On the existing resource of type `SUNW.rac_framework`, set the value of the `Type_version` property to the version you upgraded to in Step 2.**

In the following command example, replace *rac-fmwk-rs* with the actual name of your resource of type `SUNW.rac_framework` and replace *N* with the resource-type version you upgraded to.

```
# clresource set -p Type_version=N rac-fmwk-rs
```

4 **Disable Oracle Clusterware to prevent it from automatically starting on all nodes.**

```
# ${CRS_HOME}/bin/crsctl disable crs
```

5 **Stop Oracle Clusterware and all DBMS processes on all nodes.**

```
# ${CRS_HOME}/bin/crsctl stop crs
```

6 **Unmanage the RAC framework resource group.**

```
# clresource disable -g rac-fmwk-rg
# clresourcegroup offline rac-fmwk-rg
# clresourcegroup unmanage rac-fmwk-rg
```

7 **Reboot the cluster to ensure that the RAC framework processes have exited.**

Alternatively, reboot one node at a time, to allow clustered data services other than Oracle RAC to continue to provide service.

```
# scshutdown -g0 -y
```

▼ SPARC: How to Convert From Oracle UDLM to Oracle Solaris Cluster Native SKGXN

(Oracle 11*g* release 2 only) Perform this procedure to convert the SKGXN interface from Oracle UDLM to Oracle Solaris Cluster native SKGXN.

Before You Begin Ensure that you prepare the cluster to change the SKGXN interface. See "SPARC: How to Prepare the Cluster to Change the SKGXN Interfaces" on page 193.

1 **Become superuser.**

2 **Remove the Oracle UDLM resource from the RAC framework resource group.**

```
# clresource delete rac-udlm-rs
```

`rac-udlm-rs` The name of the `SUNW.rac_udlm` resource

3 **Remove the Oracle UDLM package from all nodes.**

```
# pkgrm ORCLudlm
```

4 **Enable the resources in the RAC framework resource group.**

```
# clresource enable -g rac-fmwk-rg
```

`-g` *rac-fmwk-rg* Specifies the name of the RAC framework resource group

5 **Manage the RAC framework resource group.**

```
# clresourcegroup manage -g rac-fmwk-rg
```

6 **Bring the RAC framework resource group online.**

```
# clresourcegroup online -g rac-fmwk-rg
```

Next Steps Bring Oracle RAC online. Go to "SPARC: How to Bring Oracle RAC Online After Switching the SKGXN" on page 196.

▼ SPARC: How to Convert SKGXN from Oracle Solaris Cluster Native SKGXN to Oracle UDLM

(Oracle 11*g* release 2 only) Perform this procedure to convert the SKGXN interface from Oracle Solaris Cluster native SKGXN to Oracle UDLM.

Before You Begin Prepare the cluster to change the SKGXN interface. See "SPARC: How to Prepare the Cluster to Change the SKGXN Interfaces" on page 193.

1 **Become superuser.**

2 **Install the `Oracle UDLM` package on all nodes.**

```
# pkgadd -d pkgdir ORCLudlm
```

3 **Create the Oracle UDLM resource in the RAC framework resource group.**

```
# clresource create -g rac-fmwk-rg -t SUNW.rac_udlm \
-y resource_dependencies=rac-fmwk-rg rac-udlm-rs
```

`-g` *rac-fmwk-rg* Specifies the name of the RAC framework resource group

`rac-udlm-rs` The name of the `SUNW.rac_udlm` resource

4 **Enable the resources.**

```
# clresource enable -g rac-fmwk-rg
```

5 **Manage the resource groups.**

```
# clresourcegroup manage -g rac-fmwk-rg
```

6 **Bring the resource groups online.**

```
# clresourcegroup online -g rac-fmwk-rg
```

Next Steps Bring Oracle RAC online. Go to "SPARC: How to Bring Oracle RAC Online After Switching the SKGXN" on page 196.

▼ SPARC: How to Bring Oracle RAC Online After Switching the SKGXN

(Oracle 11*g* release 2 only) Perform this procedure to bring Oracle RAC online after switching the SKGXN interface. Also refer to the Oracle documentation that corresponds to your version of Oracle RAC for information about bringing Oracle RAC online.

1 **Become superuser.**

2 **Enable Oracle Clusterware to start automatically in the future.**

```
# ${CRS_HOME}/bin/crsctl enable crs
```

3 **Start Oracle Clusterware and all DBMS processes.**

```
# ${CRS_HOME}/bin/crsctl stop crs
```

Removing Support for Oracle RAC

You can remove Support for Oracle RAC from the following entities:

- **A cluster.** See "How to Remove Support for Oracle RAC From a Cluster" on page 196.
- **Selected nodes in a cluster.** See "How to Remove Support for Oracle RAC From Selected Nodes" on page 202.

▼ How to Remove Support for Oracle RAC From a Cluster

Perform this task to remove Support for Oracle RAC from all nodes in a cluster.

On a cluster where multiple Oracle RAC databases are running, perform this task to remove an Oracle RAC database from the cluster. The remaining Oracle RAC databases continue to run in the cluster.

This task involves removing the following resource groups from the cluster in the following order:

- The resource group for the Oracle RAC database
- Resource groups for logical hostname resources
- Resource groups for scalable file-system mount-point resources
- Resource groups that contain resources for the Sun QFS metadata server
- Resource groups for scalable device group resources
- The RAC framework resource group
- The multiple-owner volume-manager framework resource group, if used.

Caution – You might perform this task to remove an Oracle RAC database from a cluster where multiple Oracle RAC databases are running. In this situation, do *not* remove any resource group on whose resources the remaining Oracle RAC databases depend.

For example, you might have configured multiple database file systems to depend on a single device group. In this situation, do *not* remove the resource group that contains the resource for the scalable device group.

Similarly, if multiple databases depend on the RAC framework resource group, do *not* remove this resource group.

Before You Begin Ensure that the cluster node from which you perform this task is booted in cluster mode.

1 **On one node of the cluster, become superuser.**

2 **(Oracle 9*i*, 10*g* release 2, or 11*g* only) Remove the resource group for each Oracle RAC database that you are removing.**

If you are using Oracle 10*g* release 1, omit this step. For Oracle 10*g* release 1, no resource groups for Oracle RAC databases are configured.

For each Oracle RAC database that you are removing, type the following command:

```
# clresourcegroup delete -F rac-db-rg
```

rac-db-rg Specifies the resource group that you are removing.

3 **(Oracle 9*i* only) Remove all resource groups for logical hostname resources that are used by each Oracle RAC database that you are removing.**

If you are using Oracle 10*g* release 1, 10*g* release 2, or 11*g*, omit this step. For Oracle 10*g* release 1, 10*g* release 2, and 11*g*, no resource groups for logical hostname resources are configured.

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F lh-rg
```

lh-rg Specifies the resource group that you are removing.

4 **Use Oracle utilities to remove from the cluster each Oracle RAC database that you no longer require.**

5 **If you are removing Support for Oracle RAC entirely, use Oracle utilities to remove the following items from all nodes in a cluster:**

- The Oracle RAC software
- The Oracle Clusterware software

6 **(Oracle 10*g* release 2 or 11*g* only) Disable the Oracle Clusterware framework resource.**

If you are using Oracle 9*i* or Oracle 10*g* release 1, omit this step. For Oracle 9*i* and Oracle 10*g* release 1, no Oracle Clusterware framework resource is configured.

```
# clresource disable crs-framework-rs
```

crs-framework-rs Specifies the name of the resource that you are disabling. This resource is the instance of the `SUNW.crs_framework` resource type that is configured in the cluster.

7 **Remove any resource groups that contain scalable file-system mount-point resources.**

If no resource groups that contain scalable file-system mount-point resources are configured, omit this step.

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F scal-mp-rg
```

scal-mp-rg Specifies the resource group that you are removing.

8 **Remove any resource groups that contain resources for the Sun QFS metadata server.**

If no resource groups that contain resources for the Sun QFS metadata server are configured, omit this step.

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F qfs-mds-rg
```

qfs-mds-rg Specifies the resource group that you are removing.

9 **Remove the Sun QFS shared file systems that were represented by resources in the resource group that you deleted in Step 8.**

For instructions for performing this task, see Using SAM-QFS With Sun Cluster.

10 **Remove any resource groups that contain scalable device group resources.**

If no resource groups that contain scalable device group resources are configured, omit this step.

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F scal-dg-rg
```

scal-dg-rg Specifies the resource group that you are removing.

11 Destroy any scalable device groups that were affected by the removal of resource groups in Step 10.

How to perform this step depends on the type of the scalable device group.

- **For each Solaris Volume Manager for Sun Cluster multi-owner disk set, destroy the disk set as follows:**

  a. Remove all metadevices such as volumes, soft partitions, and mirrors from the disk set.

  Use the `metaclear(1M)` command for this purpose.

  ```
  # metaclear -s scal-dg-ms -a
  ```

  `-s` *scal-dg-ms* Specifies the name of the disk set from which you are removing metadevices.

  b. Remove all global devices from the disk set.

  ```
  # metaset -s scal-dg-ms -d -f alldevices
  ```

  `-s` *scal-dg-ms* Specifies the name of the disk set from which you are removing global devices.

  alldevices Specifies a space-separated list that contains *all* global devices that were added to the disk set when the disk set was created. The format of each device ID path name is `/dev/did/dsk/d`*N*, where *N* is the device number.

  c. Remove all nodes from the disk set that you are destroying.

  The removal of all nodes from a disk set destroys the disk set.

  ```
  # metaset -s scal-dg-ms -d -h allnodes
  ```

  `-s` *scal-dg-ms* Specifies the name of the disk set that you are destroying.

  `-h` *allnodes* Specifies a space-separated list that contains *all* nodes that were added to the disk set when the disk set was created.

- **For each VxVM shared-disk group, use Veritas commands to destroy the VxVM shared-disk group.**

  For more information, see your VxVM documentation.

Note – If you are removing an Oracle RAC database from a cluster where multiple Oracle RAC databases are running, omit the remaining steps in this procedure.

---

12 Remove the RAC framework resource group.

```
# clresourcegroup delete -F rac-fmwk-rg
```

rac-fmwk-rg Specifies the resource group that you are removing.

13 Remove the multiple-owner volume-manager framework resource group, if used.

```
# clresourcegroup delete -F vucmm-fmwk-rg
```

vucmm-fmwk-rg Specifies the resource group that you are removing.

14 Unregister the resource type of each resource that you removed in this procedure.

```
# clresourcetype unregister resource-type-list
```

resource-type-list Specifies a comma-separated list of the names of the resource types that you are unregistering. For a list of the resource types that are associated with Support for Oracle RAC, see "Automatically Generated Names for Oracle Solaris Cluster Objects" on page 132.

15 (Optional) From each node in the cluster, uninstall the Support for Oracle RAC software packages.

Use the `uninstaller` program for this purpose. For more information, see Chapter 8, "Uninstalling," in *Sun Java Enterprise System 5 Update 1 Installation Guide for UNIX*.

16 Reboot each node in the cluster.

Example 7–2 Removing Support for Oracle RAC From a Cluster

This example shows the sequence of operations that is required to remove Support for Oracle RAC from all nodes of a four-node cluster. The nodes in this cluster are named `pclus1`, `pclus2`, `pclus3`, and `pclus4`. Only one Oracle RAC database is configured on the cluster.

The configuration of Support for Oracle RAC in this example is as follows:

- The version of Oracle RAC is 10*g* Release 2.
- Sun QFS shared file system on Solaris Volume Manager for Sun Cluster is used to store Oracle files.
- The mount points of file systems that are used for Oracle files are as follows:
  - Oracle database files: `/db_qfs/OraData`
  - Oracle binary files and related files: `/db_qfs/OraHome`
- The `oradg` disk set is used only by the Oracle RAC database.
- The name of the Oracle RAC database is `swb`.
- The Sun QFS shared file systems use a Solaris Volume Manager for Sun Cluster multi-owner disk set that is named `oradg`. The creation of this disk set is shown in Example 3–1.

- The configuration uses a multiple-owner volume-manager framework resource group.

The configuration of resource groups in this example is shown in the following table.

Resource Group	Purpose
`vucmm-framework-rg`	Multiple-owner volume-manager resource group.
`rac-framework-rg`	RAC framework resource group.
`scaldg-rg`	Resource group for scalable device-group resources.
`qfsmds-rg`	Resource group for Sun QFS metadata server resources.
`scalmnt-rg`	Resource group for scalable file-system mount-point resources.
`rac_server_proxy-rg`	Oracle RAC database resource group.

The resource groups that are required for this configuration are shown in Figure A–2.

1. To remove the resource group for the Oracle RAC database, the following command is run:

   ```
   # clresourcegroup delete -F rac_server_proxy-rg
   ```

   After the removal of this resource group, Oracle utilities are used to remove the following items:

   - The Oracle RAC database
   - The Oracle RAC software
   - The Oracle Clusterware software

   The removal of these items is beyond the scope of this example.
2. To disable the Oracle Clusterware framework resource, the following command is run:

   ```
   # clresource disable crs_framework-rs
   ```
3. To remove the resource group that contains scalable file-system mount-point resources, the following command is run:

   ```
   # clresourcegroup delete -F scalmnt-rg
   ```
4. To remove the resource group that contains resources for the Sun QFS metadata server, the following command is run:

   ```
   # clresourcegroup delete -F qfsmds-rg
   ```

   After the removal of this resource group, Sun QFS utilities are used to remove the Sun QFS shared file systems that are used for Oracle files. The removal of these file systems is beyond the scope of this example.
5. To remove the resource group that contains scalable device group resources, the following command is run:

   ```
   # clresourcegroup delete -F scaldg-rg
   ```

6. To destroy the Solaris Volume Manager for Sun Cluster multi-owner disk set `oradg`, the following commands are run:

```
# metaclear -s oradg -a
# metaset -s oradg -d \
-f /dev/did/dsk/d8 /dev/did/dsk/d9 /dev/did/dsk/d15 /dev/did/dsk/d16
# metaset -s oradg -d -h pclus1 pclus2 pclus3 pclus4
```

   The following global devices are removed from the disk set:

   - `/dev/did/dsk/d8`
   - `/dev/did/dsk/d9`
   - `/dev/did/dsk/d15`
   - `/dev/did/dsk/d16`

7. To remove the RAC framework resource group, the following command is run:

```
# clresourcegroup delete -F rac-framework-rg
```

8. To remove the multiple-owner volume-manager framework resource group, the following command is run:

```
# clresourcegroup delete -F vucmm-framework-rg
```

9. To unregister the resource type of each resource that was removed, the following command is run:

```
# clresourcetype unregister \
SUNW.scalable_rac_server_proxy,\
SUNW.ScalMountPoint,\
SUNW.qfs,\
SUNW.ScalDeviceGroup,\
SUNW.rac_svm,\
SUNW.crs_framework,\
SUNW.rac_udlm,\
SUNW.rac_framework
```

   This configuration is running on the SPARC platform. Therefore, `SUNW.rac_udlm` is in the list of resource types that are unregistered.

   After the unregistration of these resource types, the following operations are performed:

   - The removal of the Support for Oracle RAC software packages
   - The reboot of each node in the cluster

   These operations are beyond the scope of this example.

▼ How to Remove Support for Oracle RAC From Selected Nodes

Perform this task to remove Support for Oracle RAC from selected nodes.

On a cluster where multiple Oracle RAC databases are running, perform this task to remove an Oracle RAC database from selected nodes. The Oracle RAC database that you remove continues to run on the other cluster nodes. The remaining Oracle RAC databases continue to run on the selected nodes.

This task involves removing the selected nodes from the following resource groups in the following order:

- The resource group for the Oracle RAC database
- Resource groups for logical hostname resources
- Resource groups for scalable file-system mount-point resources
- Resource groups that contain resources for the Sun QFS metadata server
- Resource groups for scalable device group resources
- The RAC framework resource group

Caution – You might perform this task to remove an Oracle RAC database from selected nodes of a cluster where multiple Oracle RAC databases are running. In this situation, do *not* remove the nodes from any resource group on whose resources the remaining Oracle RAC databases depend. For example, you might have configured multiple database file systems to depend on a single device group. In this situation, do *not* remove the nodes from the resource group that contains the resource for the scalable device group. Similarly, if multiple databases depend on the RAC framework resource group, do *not* remove the nodes from this resource group.

1 Become superuser.

2 (Oracle 9*i*, 10*g* release 2, or 11*g* only) Remove the nodes from the resource group for each Oracle RAC database that you are removing.

If you are using Oracle 10*g* release 1, omit this step. For Oracle 10*g* release 1, no resource groups for Oracle RAC databases are configured.

For each Oracle RAC database that you are removing, perform the following steps:

a. Take offline the resource group for the Oracle RAC database on the nodes from which you are removing Support for Oracle RAC.

```
# clresourcegroup offline -n nodelist rac-db-rg
```

`-n` *nodelist*
Specifies a comma-separated list of cluster nodes from which you are taking offline the resource group.

rac-db-rg
Specifies the name of the resource group that you are taking offline.

b. Remove the nodes from the node list of the resource group for the Oracle RAC database.

```
# clresourcegroup remove-node -n nodelist rac-db-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

rac-db-rg
Specifies the name of the resource group from which you are removing nodes.

3 (Oracle 9*i* only) Remove the nodes from all resource groups for logical hostname resources that are used by each Oracle RAC database that you are removing.

If you are using Oracle 10*g* release 1, Oracle10*g* release 2, or 11*g*, omit this step. For Oracle 10*g* release 1, 10*g* release 2, and 11*g*, no resource groups for logical hostname resources are configured.

For each resource group from which you are removing nodes, perform the following steps:

a. Switch the resource group to a node from which you are *not* removing Support for Oracle RAC.

```
# clresourcegroup switch -n node-to-stay lh-rg
```

node-to-stay Specifies the node to which you are switching the resource group. This node must be a node from which you are *not* removing Support for Oracle RAC.

lh-rg Specifies the name of the resource group that you are switching to another node.

b. Remove the nodes from the node list of the resource group.

```
# clresourcegroup remove-node -n nodelist lh-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

lh-rg
Specifies the name of the resource group from which you are removing nodes.

4 (Oracle 9*i* only) Remove each resource group for logical hostname resources whose primary node you removed from resource groups in Step 3.

These resource groups are no longer required because the Oracle RAC database instances that the groups serve are being removed.

Do *not* remove any resource groups from which you removed *only* secondary nodes in Step 3.

For each resource group that you are removing, type the following command:

```
# clresourcegroup remove -F lh-rg-rm-prim
```

lh-rg-rm-prim Specifies the name of the resource group that you are removing.

5 (Oracle 10*g* release 2 or 11*g* only) Remove each node that you are removing from the list of nodes where the Oracle Clusterware resource for the Oracle database runs.

If you are using Oracle 9*i* or Oracle 10*g* release 1 , omit this step. For Oracle 9*i* and Oracle 10*g* release 1, no Oracle Clusterware resources that represent Oracle Solaris Cluster resources are configured.

Note – In this step, the syntax of Oracle commands for Oracle release 10*g* release 2 or 11*g* is provided. If you are using an Oracle version other than 10*g* release 2 or 11*g*, see your Oracle documentation for the correct command syntax.

```
# crs-home/bin/crs_register ora.dbname.sid.inst \
-update -r "ora.node-name.vip"
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

dbname Specifies the database name of the database instance that the Oracle Clusterware resource represents.

sid Specifies the Oracle SID of the database instance that the Oracle Clusterware resource represents.

node-name Specifies the host name of the node where the Oracle Clusterware resource runs.

6 (Oracle 10*g* release 2 or 11*g* only) From each node that you are removing, remove each Oracle Clusterware resource that represents an Oracle Solaris Cluster resource from whose resource group you are removing nodes.

An Oracle Clusterware resource is configured for each Oracle Solaris Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend.

If you are using Oracle 9*i* or Oracle 10*g* release 1 , omit this step. For Oracle 9*i* and Oracle 10*g* release 1, no Oracle Clusterware resources that represent Oracle Solaris Cluster resources are configured.

Note – In this step, the syntax of Oracle commands for Oracle 10*g* release 2 or 11*g* is provided. If you are using an Oracle version other than 10*g* release 2 or 11*g*, see your Oracle documentation for the correct command syntax.

For each Oracle Clusterware resource that you are removing, perform the following steps on each node from which you are removing the resource:

a. Stop the Oracle Clusterware resource that you are removing.

```
# crs-home/bin/crs_stop sun.node-name.sc-rs
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

node-name Specifies the host name of the node where the Oracle Clusterware resource runs.

sc-rs Specifies the name of the Oracle Solaris Cluster resource that the Oracle Clusterware resource represents.

b. Unregister the Oracle Clusterware resource that you are removing.

```
# crs-home/bin/crs_unregister sun.node-name.sc-rs
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

node-name Specifies the host name of the node where the Oracle Clusterware resource runs.

sc-rs Specifies the name of the Oracle Solaris Cluster resource that the Oracle Clusterware resource represents.

c. Delete the profile for the Oracle Clusterware resource that you are removing.

```
# crs-home/bin/crs_profile -delete sun.node-name.sc-rs \
-dir /var/cluster/ucmm/profile
```

crs-home Specifies the Oracle Clusterware home directory. This directory contains the Oracle Clusterware binary files and Oracle Clusterware configuration files.

node-name Specifies the host name of the node where the Oracle Clusterware resource runs.

sc-rs Specifies the name of the Oracle Solaris Cluster resource that the Oracle Clusterware resource represents.

7 Use Oracle utilities to remove the following items from each node from which you are removing Support for Oracle RAC:

- The Oracle RAC database
- Oracle Clusterware

8 Switch any resource groups that contain resources for the Sun QFS metadata server to a node from which you are *not* removing Support for Oracle RAC.

If no configured resource groups contain resources for the Sun QFS metadata server, omit this step.

```
# clresourcegroup switch -n node-to-stay qfs-mds-rg
```

node-to-stay Specifies the node to which you are switching the resource group. This node must be a node from which you are *not* removing Support for Oracle RAC.

qfs-mds-rg Specifies the name of the resource group that you are switching to another node.

9 Remove the nodes from any resource groups that contain scalable file-system mount-point resources.

If no resource groups that contain scalable file-system mount-point resources are configured, omit this step.

For each resource group from which you are removing nodes, perform the following steps:

a. Take offline the resource group on the nodes from which you are removing Support for Oracle RAC.

```
# clresourcegroup offline -n nodelist scal-mp-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes from which you are taking offline the resource group.

scal-mp-rg
Specifies the name of the resource group that you are taking offline.

b. Remove the nodes from the node list of the resource group.

```
# clresourcegroup remove-node -n nodelist scal-mp-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

scal-mp-rg
Specifies the name of the resource group from which you are removing nodes.

10 Remove the nodes from the node list of any resource groups that contain resources for the Sun QFS metadata server.

If no resource groups that contain resources for the Sun QFS metadata server are configured, omit this step.

The resource groups to modify are the resource groups that you switched to another node in Step 8.

```
# clresourcegroup remove-node -n nodelist qfs-mds-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

qfs-mds-rg
Specifies the name of the resource group from which you are removing nodes.

11 Remove the configuration of the Sun QFS shared file systems from the nodes.

For instructions for performing this task, see Using SAM-QFS With Sun Cluster.

12 Remove the nodes from any resource groups that contain scalable device group resources.

If no resource groups that contain scalable device group resources are configured, omit this step.

For each resource group from which you are removing nodes, perform the following steps:

a. Take offline the resource group on the nodes from which you are removing Support for Oracle RAC.

```
# clresourcegroup offline -n nodelist scal-dg-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes from which you are taking offline the resource group.

scal-dg-rg
Specifies the name of the resource group that you are taking offline.

b. Remove the nodes from the node list of the resource group.

```
# clresourcegroup remove-node -n nodelist scal-dg-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

scal-dg-rg
Specifies the name of the resource group from which you are removing nodes.

13 Remove the nodes from any scalable device groups that were affected by the removal of nodes from resource groups in Step 12.

How to perform this step depends on the type of the scalable device group.

- **For each Solaris Volume Manager for Sun Cluster multi-owner disk set, type the following command:**

  ```
  # metaset -s scal-dg-ms -d -h nodelist
  ```

  -s *scal-dg-ms* Specifies the name of the disk set from which you are removing nodes.

  -h *nodelist* Specifies a space-separated list of the nodes that you are removing from the disk set.

- **For each VxVM shared-disk group, use Veritas commands to remove the nodes from the VxVM shared-disk group.**

  For more information, see your VxVM documentation.

Note – If you are removing an Oracle RAC database from selected nodes of a cluster where multiple Oracle RAC databases are running, omit the remaining steps in this procedure.

14 Remove the nodes from the RAC framework resource group.

a. Take offline the resource group on the nodes from which you are removing Support for Oracle RAC.

```
# clresourcegroup offline -n nodelist rac-fmwk-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes from which you are taking offline the resource group.

rac-fmwk-rg
Specifies the name of the resource group that you are taking offline.

b. Remove the nodes from the node list of the resource group.

```
# clresourcegroup remove-node -n nodelist rac-fmwk-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

rac-fmwk-rg
Specifies the name of the resource group from which you are removing nodes.

15 Remove the nodes from the multiple-owner volume-manager framework resource group, if used.

a. Take offline the resource group on the nodes from which you are removing Support for Oracle RAC.

```
# clresourcegroup offline -n nodelist vucmm-fmwk-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes from which you are taking offline the resource group.

vucmm-fmwk-rg
Specifies the name of the resource group that you are taking offline.

b. Remove the nodes from the node list of the resource group.

```
# clresourcegroup remove-node -n nodelist vucmm-fmwk-rg
```

-n *nodelist*
Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

vucmm-fmwk-rg
Specifies the name of the resource group from which you are removing nodes.

16 (Optional) From each node that you removed, uninstall the Support for Oracle RAC software packages.

Use the `uninstaller` program for this purpose. For more information, see Chapter 8, "Uninstalling," in *Sun Java Enterprise System 5 Update 1 Installation Guide for UNIX*.

17 Reboot each node from which you from which you removed Support for Oracle RAC.

Example 7–3 Removing Support for Oracle RAC From Selected Nodes

This example shows the sequence of operations that is required to remove Support for Oracle RAC from nodes `pclus3` and `pclus4` of a four-node cluster.

The configuration of Support for Oracle RAC in this example is as follows:

- The version of Oracle RAC is 10*g* Release 2.
- Sun QFS shared file system on Solaris Volume Manager for Sun Cluster is used to store Oracle files.
- The mount points of file systems that are used for Oracle files are as follows:
  - Oracle database files: `/db_qfs/OraData`
  - Oracle binary files and related files: `/db_qfs/OraHome`
- The `oradg` disk set is used only by the Oracle RAC database.
- The name of the Oracle RAC database is `swb`.
- The Sun QFS shared file systems use a Solaris Volume Manager for Sun Cluster multi-owner disk set that is named `oradg`. The creation of this disk set is shown in Example 3–1.
- The configuration uses a multiple-owner volume-manager framework resource group.

The configuration of resource groups in this example is shown in the following table.

Resource Group	Purpose
`vucmm-framework-rg`	Multiple-owner volume-manager resource group.
`rac-framework-rg`	RAC framework resource group.
`scaldg-rg`	Resource group for scalable device-group resources.
`qfsmds-rg`	Resource group for Sun QFS metadata server resources.
`scalmnt-rg`	Resource group for scalable file-system mount-point resources.
`rac_server_proxy-rg`	Oracle RAC database resource group.

The resource groups that are required for this configuration are shown in Figure A–2.

1. To remove nodes `pclus3` and `pclus4` from the resource group for the Oracle RAC database, the following commands are run:

```
# clresourcegroup offline -n pclus3,pclus4 rac_server_proxy-rg
# clresourcegroup remove-node -n pclus3,pclus4 rac_server_proxy-rg
```

2. To remove nodes `pclus3` and `pclus4` from the node list of the Oracle Clusterware resource for the Oracle RAC database, the following commands are run:

```
# /db_qfs/OraHome/crs/bin/crs_register ora.swb.swb3.inst \
-update -r "ora.pclus3.vip"
# /db_qfs/OraHome/crs/bin/crs_register ora.swb.swb4.inst \
-update -r "ora.pclus4.vip"
```

3. To remove from nodes `pclus3` and `pclus4` the Oracle Clusterware resources that represent Oracle Solaris Cluster resources, the following commands are run:

Removal of resource for Oracle database files from node plcus3

```
# /db_qfs/OraHome/crs/bin/crs_stop sun.pclus3.scaloramnt-OraData-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclus3.scaloramnt-OraData-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclus3.scaloramnt-OraData-rs \
-dir /var/cluster/ucmm/profile
```

Removal of resource for Oracle binary files from node plcus3

```
# /db_qfs/OraHome/crs/bin/crs_stop sun.pclus3.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclus3.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclus3.scaloramnt-OraHome-rs \
-dir /var/cluster/ucmm/profile
```

Removal of resource for Oracle database files from node plcus4

```
# /db_qfs/OraHome/crs/bin/crs_stop sun.pclus4.scaloramnt-OraData-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclus4.scaloramnt-OraData-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclus4.scaloramnt-OraData-rs \
-dir /var/cluster/ucmm/profile
```

Removal of resource for Oracle binary files from node plcus4

```
# /db_qfs/OraHome/crs/bin/crs_stop sun.pclus4.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclus4.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclus4.scaloramnt-OraHome-rs \
-dir /var/cluster/ucmm/profile
```

The commands remove Oracle Clusterware resources that represent the following Oracle Solaris Cluster resources:

- `scaloramnt-OraData-rs` – A resource of type `SUNW.ScalMountPoint` that represents the mount point of the file system for database files
- `scaloramnt-OraHome-rs` – A resource of type `SUNW.ScalMountPoint` that represents the mount point of the file system for binary files and associated files

After the resource is removed from nodes pclus3 and pclus4, Oracle utilities are used to remove the following items from these nodes:

- The Oracle RAC database
- The Oracle RAC software
- The Oracle Clusterware software

The removal of these items is beyond the scope of this example.

4. To switch the resource group that contains resources for the Sun QFS metadata server to node pclus1, the following command is run:

```
# clresourcegroup switch -n pclus1 qfsmds-rg
```

5. To remove nodes pclus3 and pclus4 from the resource group that contains scalable file-system mount-point resources, the following commands are run:

```
# clresourcegroup offline -n pclus3,pclus4 scalmnt-rg
# clresourcegroup remove-node -n pclus3,pclus4 scalmnt-rg
```

6. To remove nodes pclus3 and pclus4 from the node list of the resource group that contains resources for the Sun QFS metadata server, the following command is run:

```
# clresourcegroup remove-node -n pclus3,pclus4 qfsmds-rg
```

After nodes pclus3 and pclus4 are removed from the node list, the configuration of the Sun QFS shared file systems is removed from theses nodes. This operation is beyond the scope of this example.

7. To remove nodes pclus3 and pclus4 from the resource group that contains scalable device group resources, the following commands are run:

```
# clresourcegroup offline -n pclus3,pclus4 scaldg-rg
# clresourcegroup remove-node -n pclus3,pclus4 scaldg-rg
```

8. To remove nodes pclus3 and pclus4 from the Solaris Volume Manager for Sun Cluster multi-owner disk set oradg, the following command is run:

```
# metaset -s oradg -d -h pclus3 pclus4
```

9. To remove nodes pclus3 and pclus4 from the RAC framework resource group, the following commands are run:

```
# clresourcegroup offline -n pclus3,pclus4 rac-framework-rg
# clresourcegroup remove-node -n pclus3,pclus4 rac-framework-rg
```

10. To remove nodes pclus3 and pclus4 from the multiple-owner volume-manager framework resource group, the following commands are run:

```
# clresourcegroup offline -n pclus3,pclus4 vucmm-framework-rg
# clresourcegroup remove-node -n pclus3,pclus4 vucmm-framework-rg
```

After the removal of nodes pclus3 and pclus4 from the framework resource groups, the Support for Oracle RAC software packages can optionally be removed from the nodes.

After the removal is complete, the status of resource groups and resources is as follows:

```
# clresourcegroup status

=== Cluster Resource Groups ===

Group Name             Node Name    Suspended   Status
----------             ---------    ---------   ------
rac-framework-rg       pclus1       No          Online
                       pclus2       No          Online

vucmm-framework-rg     pclus1       No          Online
                       pclus2       No          Online

scaldg-rg              pclus1       No          Online
                       pclus2       No          Online

qfsmds-rg              pclus1       No          Online
                       pclus2       No          Offline

scalmnt-rg             pclus1       No          Online
                       pclus2       No          Online

rac_server_proxy-rg    pclus1       No          Online
                       pclus2       No          Online

# clresource status

=== Cluster Resources ===

Resource Name            Node Name    State     Status Message
-------------            ---------    -----     --------------
rac-framework-rs         pclus1       Online    Online
                         pclus2       Online    Online

rac-udlm-rs              pclus1       Online    Online
                         pclus2       Online    Online

crs_framework-rs         pclus1       Online    Online
                         pclus2       Online    Online

vucmm-svm-rs             pclus1       Online    Online
                         pclus2       Online    Online

scaloradg-rs             pclus1       Online    Online - Diskgroup online
                         pclus2       Online    Online - Diskgroup online

qfs-mds-rs               pclus1       Online    Online - Service is online.
                         pclus2       Offline   Offline

scaloramnt-OraData-rs    pclus1       Online    Online
                         pclus2       Online    Online

scaloramnt-OraHome-rs    pclus1       Online    Online
                         pclus2       Online    Online

rac_server_proxy-rs      pclus1       Online    Online - Oracle instance UP
                         pclus2       Online    Online - Oracle instance UP
```

◆ ◆ ◆ CHAPTER 8

Upgrading Support for Oracle RAC

This chapter explains how to upgrade a configuration of Oracle Solaris Cluster Support for Oracle Real Application Clusters (Support for Oracle RAC).

If you upgrade Oracle Solaris Cluster core software, you must also upgrade the Support for Oracle RAC software. For more information, see *Oracle Solaris Cluster Upgrade Guide*.

Your existing configuration of Support for Oracle RAC might not include the Oracle RAC framework resource group. In this situation, you *must* register and configure the Oracle RAC framework resource group after upgrading Oracle Solaris Cluster software. Otherwise, Oracle RAC cannot run with Oracle Solaris Cluster software. For more information, see "Registering and Configuring the RAC Framework Resource Group" on page 65.

Information about how to upgrade an existing configuration is provided in the subsections that follow:

Note – The `SUNW.oracle_rac_server` resource type and the `SUNW.oracle_listener` resource type are unchanged in Oracle Solaris Cluster 3.3 software. If you choose to continue to use the `SUNW.oracle_rac_server` resource type and the `SUNW.oracle_listener` resource type for Oracle 9*i* RAC database instances, no upgrade of these resource types is required.

Upgrading Resources in Support for Oracle RAC

Changes to resource types of resources in Oracle Solaris Cluster 3.3 Support for Oracle RAC are summarized in the following table.

TABLE 8–1 Changes to Support for Oracle RAC Resource Types

Resource Type	Changes
`SUNW.scalable_rac_server_proxy`	Range of the `client_retry_interval` extension property is increased to 1–3600. Range of the `monitor_probe_interval` extension property is increased to 1–3600, and the default is changed to 300. Range of the `proxy_probe_timeout` extension property is increased to 5–3600, and the default is changed to 120.
`SUNW.crs_framework`	New methods now enable Support for Oracle RAC to control Oracle Clusterware starts and stops and to disable Oracle Clusterware autostart.

Upgrade these resource types if you are upgrading from an earlier version of Support for Oracle RAC.

For general instructions that explain how to upgrade a resource type, see "Upgrading a Resource Type" in *Oracle Solaris Cluster Data Services Planning and Administration Guide*.

The information that you require to complete the upgrade of the RAC framework resource types is provided in the subsections that follow:

- "Information for Registering the New Versions of Support for Oracle RAC Resource Types" on page 216
- "How to Preserve a Nondefault Value of the `reservation_timeout` Extension Property" on page 218
- "How to Remove the Resource of Type `SUNW.rac_hwraid`" on page 220

Information for Registering the New Versions of Support for Oracle RAC Resource Types

The following table lists the name of each Support for Oracle RAC resource type and its Resource Type Registration (RTR) file name.

TABLE 8–2 Support for Oracle RAC Resource Types

Resource Type	RTR File
(Oracle 10*g* and 11*g* release 1 only) SUNW.asm_diskgroup	/opt/cluster/lib/rgm/rtreg/SUNW.asm_diskgroup
SUNW.crs_framework	/usr/cluster/lib/rgm/rtreg/SUNW.crs_framework
SUNW.qfs[1]	/usr/cluster/lib/rgm/rtreg/SUNW.qfs
SPARC: SUNW.rac_cvm[2]	/usr/cluster/lib/rgm/rtreg/SUNW.rac_cvm
SUNW.rac_framework	/usr/cluster/lib/rgm/rtreg/SUNW.rac_framework
SUNW.rac_svm[2]	/usr/cluster/lib/rgm/rtreg/SUNW.rac_svm
SPARC: SUNW.rac_udlm	/usr/cluster/lib/rgm/rtreg/SUNW.rac_udlm
(Oracle 11*g* release 2 only) SUNW.scalable_asm_diskgroup_proxy	/opt/cluster/lib/rgm/rtreg/SUNW.scalable_asm_diskgroup_proxy
SUNW.scalable_asm_instance	/opt/cluster/lib/rgm/rtreg/SUNW.scalable_asm_instance
SUNW.scalable_asm_instance_proxy	/opt/cluster/lib/rgm/rtreg/SUNW.scalable_asm_instance_proxy
SUNW.scalable_rac_listener	/opt/cluster/lib/rgm/rtreg/SUNW.scalable_rac_listener
SUNW.scalable_rac_server	/opt/cluster/lib/rgm/rtreg/SUNW.scalable_rac_server
SUNW.scalable_rac_server_proxy	/opt/cluster/lib/rgm/rtreg/SUNW.scalable_rac_server_proxy
SUNW.ScalDeviceGroup	/opt/cluster/lib/rgm/rtreg/SUNW.ScalDeviceGroup
SUNW.ScalMountPoint	/opt/cluster/lib/rgm/rtreg/SUNW.ScalMountPoint
SPARC: SUNW.vucmm_cvm	/usr/cluster/lib/rgm/rtreg/SUNW.vucmm_cvm
SUNW.vucmm_framework	/usr/cluster/lib/rgm/rtreg/SUNW.vucmm_framework
SUNW.vucmm_svm	/usr/cluster/lib/rgm/rtreg/SUNW.vucmm_svm
SUNW.wait_zc_boot	/usr/cluster/lib/rgm/rtreg/SUNW.wait_zc_boot

[1] Delivered in the Sun QFS product

[2] Deprecated. Instead use SUNW.vucmm_cvm in a SUNW.vucmm_framework resource group. See "Migrating a Volume-Manager Resource From a RAC Framework Resource Group to a Multiple-Owner Volume Manager Framework Resource Group" on page 189.

To determine the version of the resource type that is currently registered, use the following command:

```
# clresourcetype show resource-type
```

resource-type Specifies the resource type whose version you are determining.

To determine the version of the latest installed version of a resource type, whether it is registered or not, use the following command:

```
# grep -i RT_VERSION /path/RTRfilename
```

If the version of the latest installed resource type is later than the registered version, migrate to the newer version to ensure full functionality.

Note – In the global zone, Oracle Solaris Cluster Manager or the Resource Group option of `clsetup` locates for you the available resource type versions you can upgrade to.

▼ How to Preserve a Nondefault Value of the `reservation_timeout` Extension Property

The `reservation_timeout` extension property specifies the timeout value in seconds for the reservation step of a reconfiguration of Support for Oracle RAC.

In Sun Cluster release 3.1, `reservation_timeout` was an extension property of the following resource types:

- SPARC: `SUNW.rac_cvm`
- `SUNW.rac_hwraid`
- `SUNW.rac_svm`

As of Sun Cluster release 3.2, `reservation_timeout` is an extension property *only* of the `SUNW.rac_framework` and `SUNW.vucmm_framework` resource types. This property is removed from the `SUNW.rac_cvm` resource type and the `SUNW.rac_svm` resource type. The `SUNW.rac_hwraid` resource type is removed as of Sun Cluster release 3.2.

A nondefault value of the `reservation_timeout` extension property might be set for an existing resource of type `SUNW.rac_hwraid`, `SUNW.rac_cvm`, or `SUNW.rac_svm`. If you require this value after you upgrade to Oracle Solaris Cluster release 3.3, perform this task to preserve the value.

1 *Before* **you migrate or remove the resource for which the `reservation_timeout` extension property is set, obtain the value of this property.**

```
# clresource show -p reservation_timeout -t resource-type
```

resource-type Specifies the resource type of the resource for which the `reservation_timeout` extension property is set, namely, `SUNW.rac_cvm`, `SUNW.rac_hwraid`, or `SUNW.rac_svm`.

2 **When you migrate the existing instance of the `SUNW.rac_framework` resource type, set the `reservation_timeout` extension property to the value that you obtained in Step 1.**

```
# clresource set -p type_version=version \
-p reservation_timeout=timeout  rac-framework-rs
```

version
Specifies the value of the `type_version` property for the version of `SUNW.rac_framework` to which you are migrating the instance.

timeout
Specifies the value to which you are setting the `reservation_timeout` extension property. This value is the value that you obtained in Step 1.

rac-framework-rs
Specifies the name of the resource of type `SUNW.rac_framework` on your cluster. If the `scsetup` utility of Sun Cluster release 3.1 was used to create the RAC framework resource group, this resource is named `rac_framework`.

Example 8–1 Preserving a Nondefault Value of the `reservation_timeout` Extension Property

This example shows the commands for preserving a nondefault value of the `reservation_timeout` extension property that was set for a resource of type `SUNW.rac_svm`.

Before the migration of the resource of type `SUNW.rac_svm`, the following command is run to obtain the value of the `reservation_timeout` extension property:

```
# clresource show -p reservation_timeout -t SUNW.rac_svm

=== Resources ===

Resource:                                       rac-svm-rs

  --- Standard and extension properties ---

  Reservation_timeout:                          350
    Class:                                        extension
    Description:                                  Timeout (in seconds) for the
reservation step of Oracle Solaris Cluster framework for RAC
    Per-node:                                     False
    Type:                                         int
```

During the migration of the resource of type `SUNW.rac_framework`, the following command is run to set properties as follows:

- The `type_version` property is set to *N*.
- The `reservation_timeout` extension property is set to 350 seconds.

```
# clresource set -p type_version=N \
-p reservation_timeout=350 rac_framework
```

See Also The following documentation:

- "Information for Registering the New Versions of Support for Oracle RAC Resource Types" on page 216
- The `clresource`(1CL) man page

▼ How to Remove the Resource of Type `SUNW.rac_hwraid`

As of Sun Cluster release 3.2, the `SUNW.rac_hwraid` resource type is obsolete and is not supplied with this release. Any resource of type `SUNW.rac_hwraid` is not required after an upgrade to Oracle Solaris Cluster 3.3. If a resource of this obsolete type is configured on your cluster, remove the resource.

1. **Become superuser or assume a role that provides `solaris.cluster.modify` RBAC authorization.**

2. **Delete the instance of `SUNW.rac_hwraid`, if any.**

```
# clresource delete -F -t SUNW.rac_hwraid +
```

Note – If no resource of type `SUNW.rac_hwraid` is configured on your cluster, a message to this effect is displayed. Ignore this message.

Adding Storage Resources for Oracle Files

Oracle Solaris Cluster 3.3 software includes resource types that provide fault monitoring and automatic fault recovery for global device groups and file systems.

If you are using global device groups or shared file systems for Oracle files, add storage resources to manage the availability of the storage on which the Oracle software depends.

Before adding storage resources for Oracle files, ensure that you have upgraded resources in the RAC framework resource group. For more information, see "Upgrading Resources in Support for Oracle RAC" on page 216.

For detailed instructions for adding storage resources for Oracle files, see "Registering and Configuring Storage Resources for Oracle Files" on page 82.

Follow these instructions only if you are adding storage resources to an existing configuration of storage management schemes. If you are extending an existing configuration of Support for Oracle RAC by supporting additional storage management schemes, see the following sections:

- Chapter 2, "Configuring Storage for Oracle Files"

- "How to Add a Volume Manager Resource to the Framework Resource Group" on page 186

Adding Resources for Interoperation With Oracle 10*g* Release 2 or 11*g* Oracle Clusterware

Oracle Solaris Cluster 3.3 software includes resource types that enable Oracle Solaris Cluster software and Oracle 10*g* release 2 or 11*g* Oracle Clusterware to interoperate. These resource types also enable Oracle RAC database instances to be administered from Oracle Solaris Cluster.

Note – Oracle Solaris Cluster resource types cannot be used to interoperate with Oracle Clusterware version 10*g* release 1. Instead, use version 10*g* release 1 Oracle Clusterware commands to start and shut down Oracle RAC database instances.

Before adding resources for interoperation with Oracle Clusterware version 10*g* release 2 or 11*g*, ensure that you have performed the tasks in the following sections:

- "Upgrading Resources in Support for Oracle RAC" on page 216
- "Adding Storage Resources for Oracle Files" on page 220

For detailed instructions for adding resources for interoperation with Oracle Clusterware version 10*g* release 2 or 11*g* Oracle Clusterware, see "Configuring Resources for Oracle RAC Database Instances" on page 107.

Upgrading to Oracle Solaris Cluster 3.3 Resource Types for Oracle 9*i* RAC Database Instances

As of the Sun Cluster 3.2 release, the failover resource types that automate the startup and shutdown of Oracle 9*i* RAC database instances can be replaced with multi-master resource types. The multi-master resources simplify the administration and enhance the scalability of Support for Oracle RAC.

Upgrading from Sun Cluster 3.1 software to Oracle Solaris Cluster 3.3 resource types for Oracle 9*i* RAC database instances involves the tasks that are described in the following sections:

1. "How to Take Offline Resource Groups for Oracle RAC Database Components and Restart the Components" on page 222
2. "How to Modify the Configuration of Oracle Solaris Cluster Resources for Oracle 9*i* RAC Database Instances" on page 223

Perform these tasks in the order in which they are listed.

▼ How to Take Offline Resource Groups for Oracle RAC Database Components and Restart the Components

This task involves taking offline resource groups for Oracle RAC database components and restarting the components on each cluster node individually.

This sequence of operations avoids downtime by ensuring that the Oracle RAC database components remain online on other cluster nodes.

Perform this task on each cluster node individually.

Before You Begin Ensure that you have upgraded resources in the RAC framework resource group. For more information, see "Upgrading Resources in Support for Oracle RAC" on page 216.

1 Become superuser on the cluster node.

2 If you configured Oracle listener resources in separate resource groups, take offline the resource groups from the node.

If you configured Oracle listener resources in the same resource group as the Oracle RAC server resource, omit this step.

For each resource group that you are taking offline, type the following command:

```
# clresourcegroup offline -n node rac-listener-rg
```

-n *node*	Specifies the node from which you are taking the resource group offline. This node is the node where you are performing this task.
rac-listener-rg	Specifies the resource group for Oracle listener resources that you are taking offline.

When the resource group is taken offline, the Oracle listener is stopped on the node. The Oracle listeners on other nodes are unaffected.

3 Take offline the resource group for the Oracle RAC server from the node.

```
# clresourcegroup offline -n node rac-server-rg
```

-n *node*	Specifies the node from which you are taking the resource group offline. This node is the node where you are performing this task.
rac-server-rg	Specifies the resource group for the Oracle RAC server resource that you are taking offline.

When the resource group is taken offline, the Oracle RAC server is stopped on the node. The Oracle RAC servers on other nodes are unaffected.

If you configured Oracle listener resources in the same resource group as the Oracle RAC server resource, the listeners are also stopped on the node. The Oracle listeners on other nodes are unaffected.

4 **Use Oracle utilities to restart the Oracle RAC server and Oracle listener on the node.**

Next Steps Go to "How to Modify the Configuration of Oracle Solaris Cluster Resources for Oracle 9*i* RAC Database Instances" on page 223.

▼ How to Modify the Configuration of Oracle Solaris Cluster Resources for Oracle 9*i* RAC Database Instances

Perform this task from one cluster node.

Before You Begin Ensure that the task that is explained in "How to Take Offline Resource Groups for Oracle RAC Database Components and Restart the Components" on page 222 has been performed.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **If you configured Oracle listener resources in separate resource groups, remove the resource groups.**

If you configured Oracle listener resources in the same resource group as Oracle RAC server resources, omit this step.

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F rac-listener-rg
```

rac-listener-rg Specifies the resource group for Oracle listener resources that you are removing.

3 **Remove all resource groups that contain an Oracle RAC server resource.**

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F rac-server-rg
```

rac-server-rg Specifies the resource group that contains an Oracle RAC server resource that you are removing.

4 **If you are using a Sun QFS shared file system, remove all resource groups that contain resources for the Sun QFS metadata server.**

If you are not using a Sun QFS shared file system, omit this step.

For each resource group that you are removing, type the following command:

```
# clresourcegroup delete -F qfs-mds-rg
```

qfs-mds-rg Specifies the resource group that contains resources for the Sun QFS metadata server that you are removing.

5 If you are using global device groups or shared file systems for Oracle files, add storage resources to manage the availability of the storage on which the Oracle software depends.

If you are not using global device groups or shared file systems for Oracle files, omit this step.

For instructions for performing this task, see "Adding Storage Resources for Oracle Files" on page 220.

6 Configure new resources for Oracle 9*i* RAC database instances.

For instructions for performing this task, see "Configuring Resources for Oracle RAC Database Instances" on page 107.

◆ ◆ ◆ APPENDIX A

Sample Configurations of This Data Service

The software architectures of Oracle 9*i* and Oracle 10*g* or 11*g* are different. As a result of these differences, the configuration of resources and resource groups for the Support for Oracle RAC depends on the version of Oracle that you are using. For each version of Oracle that this data service supports, the configuration of resources and resource groups also depends on the combination of storage management schemes that you are using for Oracle files.

The sections that follow illustrate the configuration of resources and resource groups for typical combinations of storage management schemes for Oracle 9*i* and Oracle 10*g* or 11*g* on the SPARC platform for both the global cluster and a zone cluster. On the x86 platform, no Oracle UDLM resource is required.

Note – These diagrams reflect, where applicable, the use of the multiple-owner volume-manager framework, `SUNW.vucmm_framework`. This framework was introduced in the Sun Cluster 3.2 11/09 release to contain volume-manager resources, if used by the Oracle RAC configuration, instead of using the RAC framework resource group.

Oracle RAC configurations that configure volume-manager resources with the RAC framework, `SUNW.rac_framework`, are still supported in this release. The use of `SUNW.rac_framework` to contain volume-manager resources might become deprecated in a future Oracle Solaris Cluster release. See "Legacy Configurations" on page 243 for all diagrams that use `SUNW.rac_framework` instead of `SUNW.vucmm_framework` for volume-manager resources.

Sample Oracle 10*g* or 11*g* Configurations in the Global Cluster

FIGURE A–1 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster

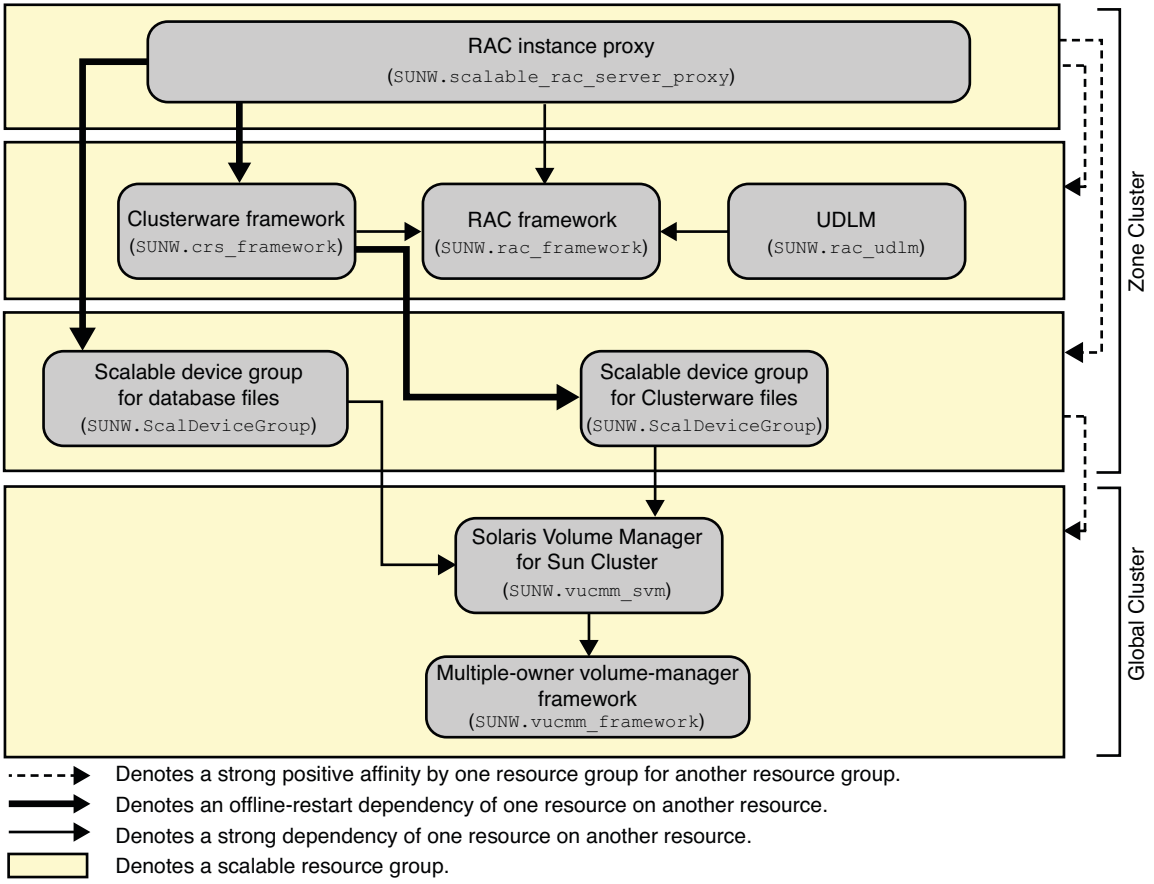


FIGURE A–2 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System

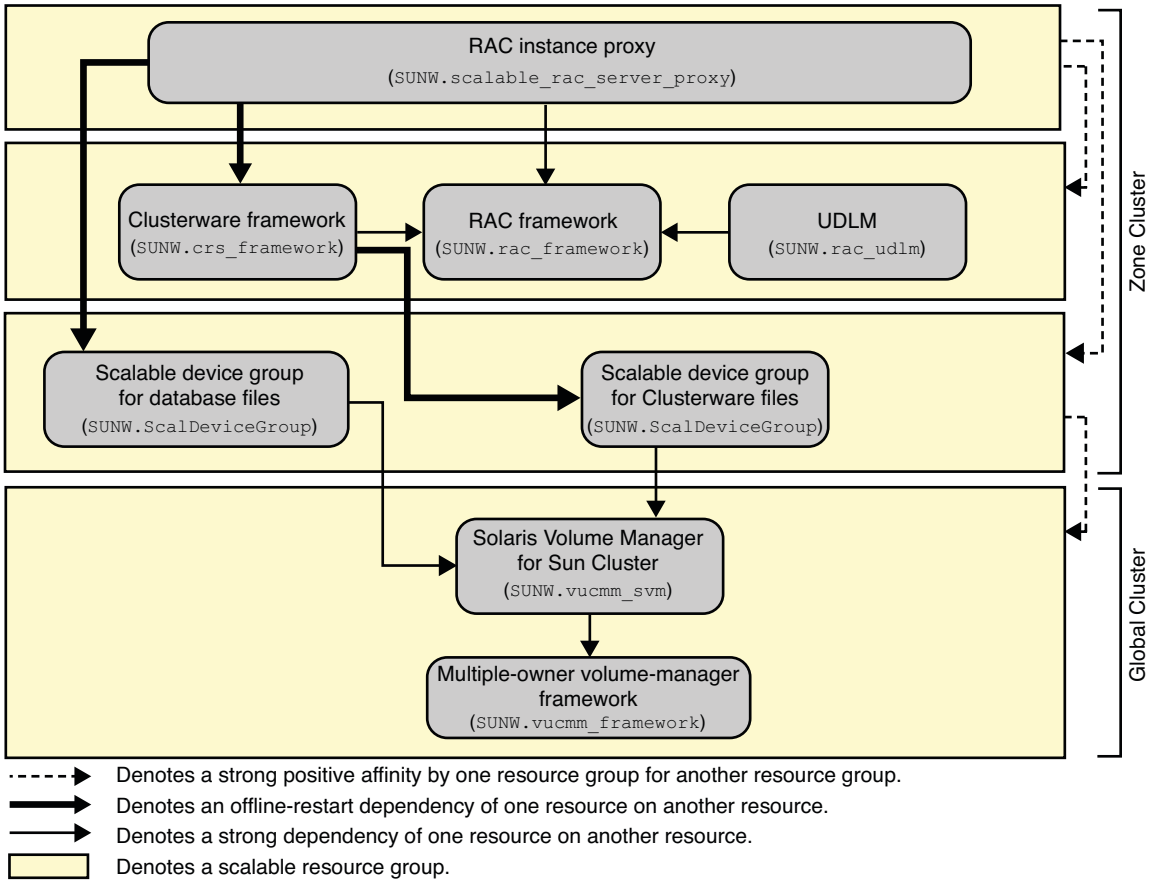


FIGURE A–3 Configuration of Oracle 10*g* or 11*g* With Sun QFS Shared File System and Hardware RAID

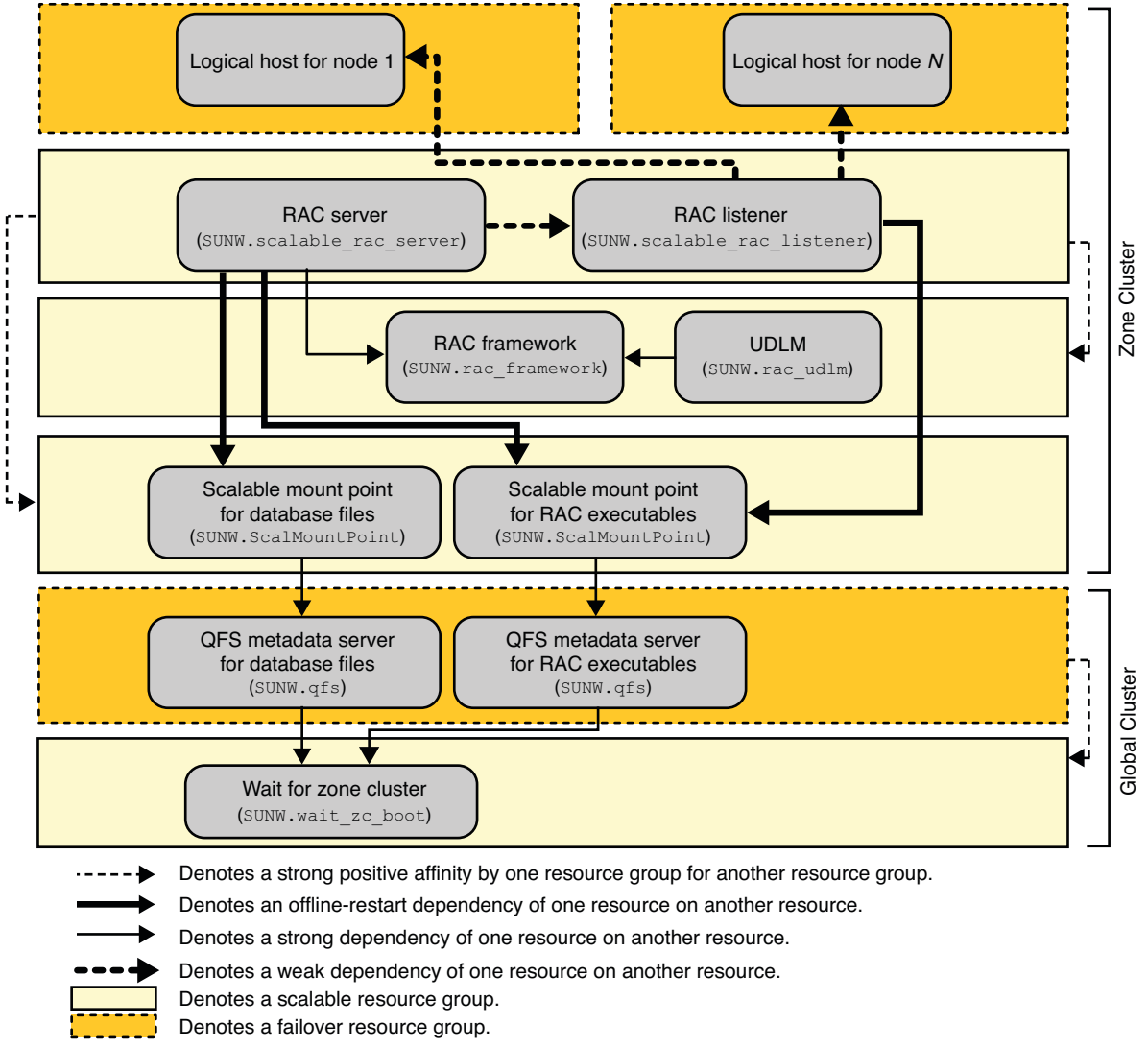


FIGURE A–4 Configuration of Oracle 10*g* or 11*g* With a NAS Device for Oracle RAC in a Global Cluster

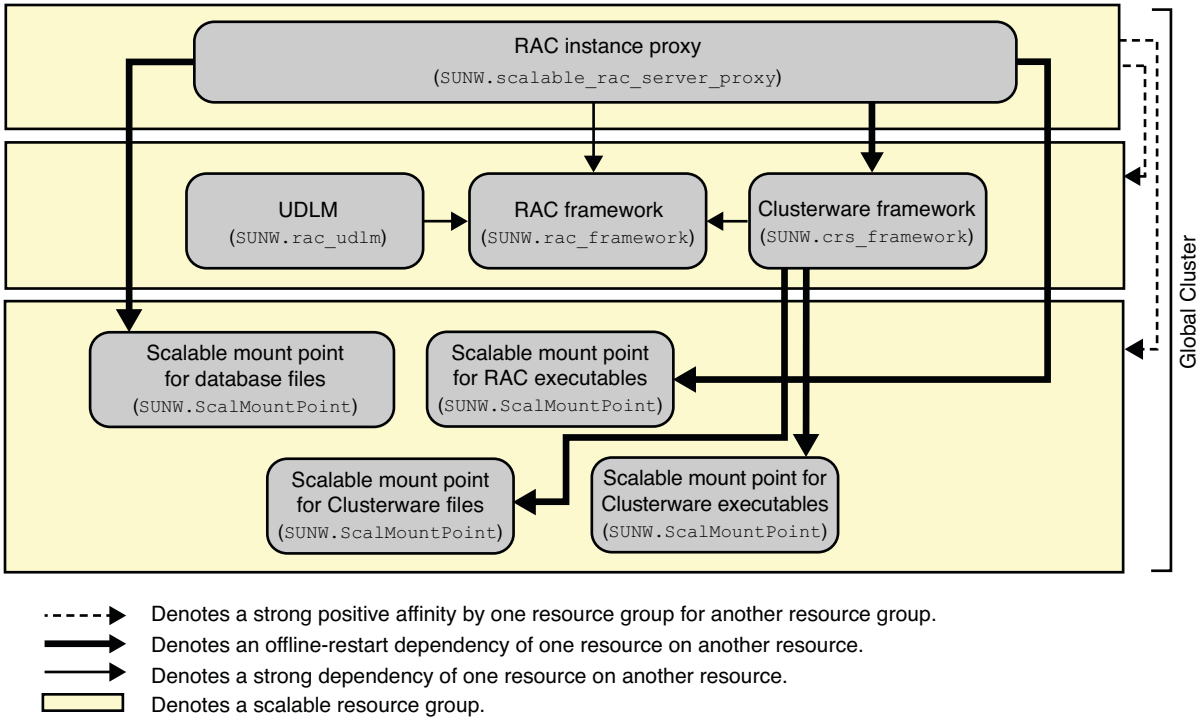
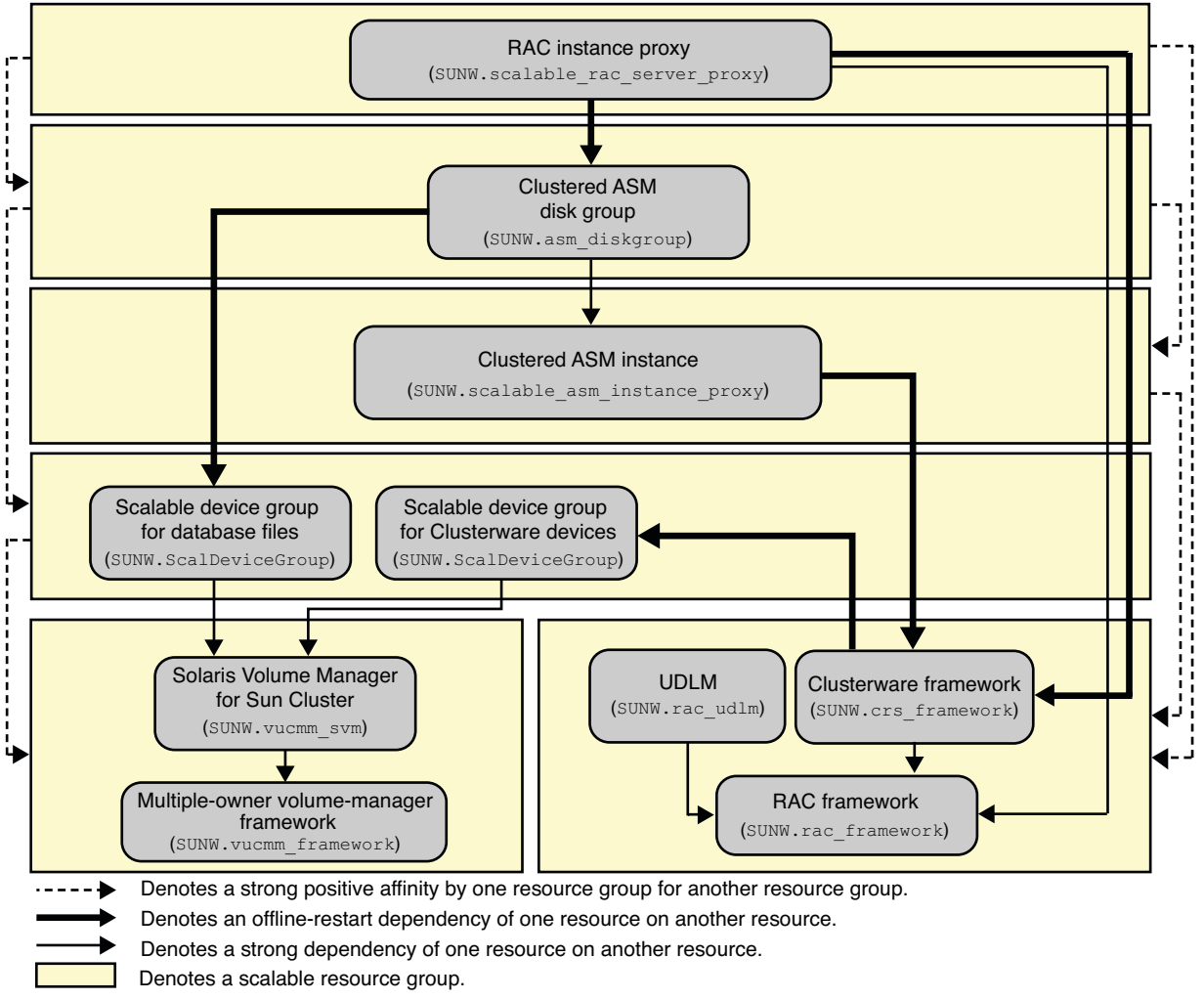
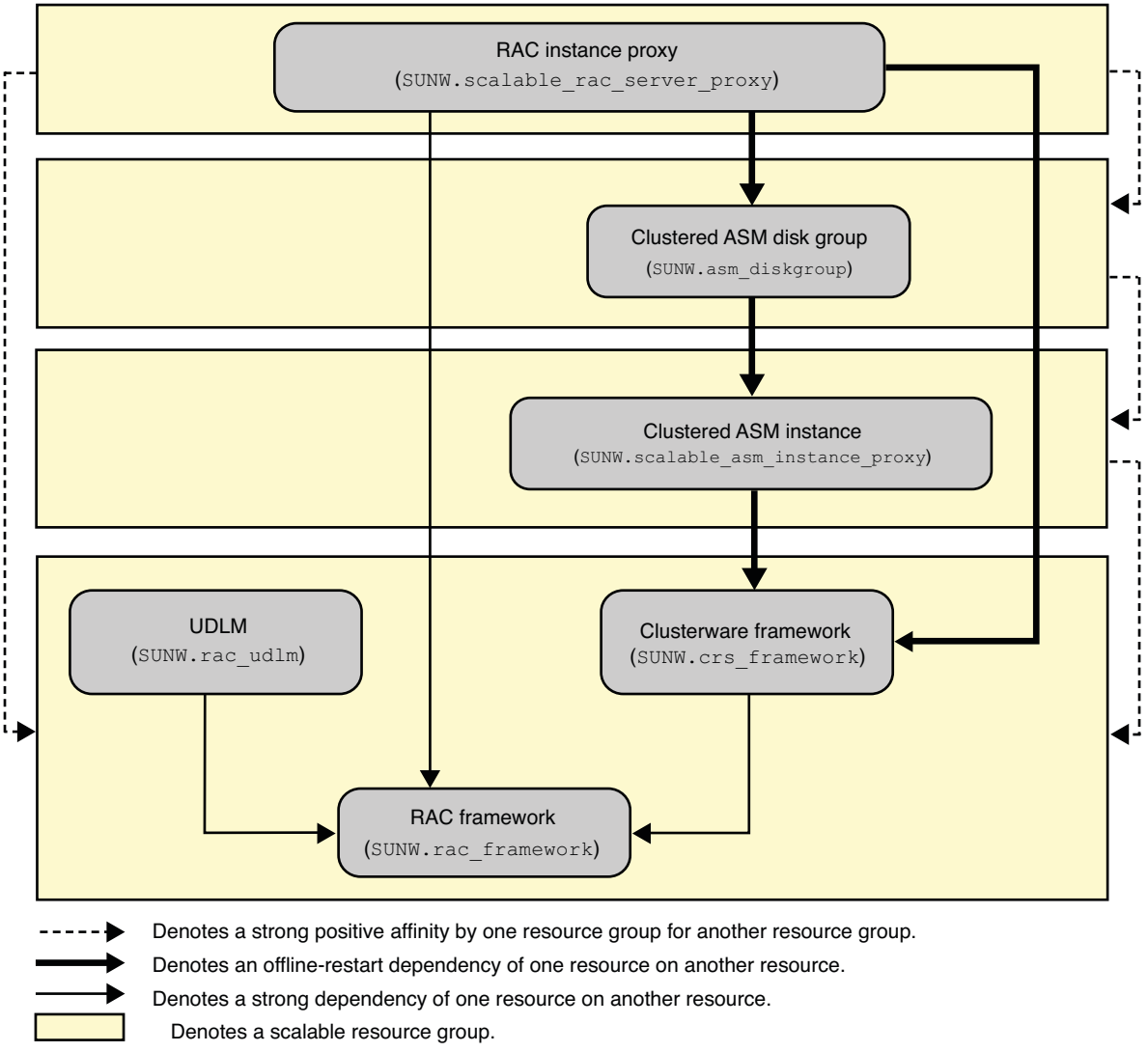


FIGURE A–5 Configuration of Oracle 10*g* or 11*g* With Oracle ASM and Solaris Volume Manager for Sun Cluster



Note – For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type instead of the `SUNW.asm_diskgroup` resource type in the clustered ASM disk group.

FIGURE A–6 Configuration of Oracle 10*g* or 11*g* With Oracle ASM and Hardware RAID



Note – For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type instead of the `SUNW.asm_diskgroup` resource type in the clustered ASM disk group.

Sample Oracle 9*i* Configurations in the Global Cluster

FIGURE A–7 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster

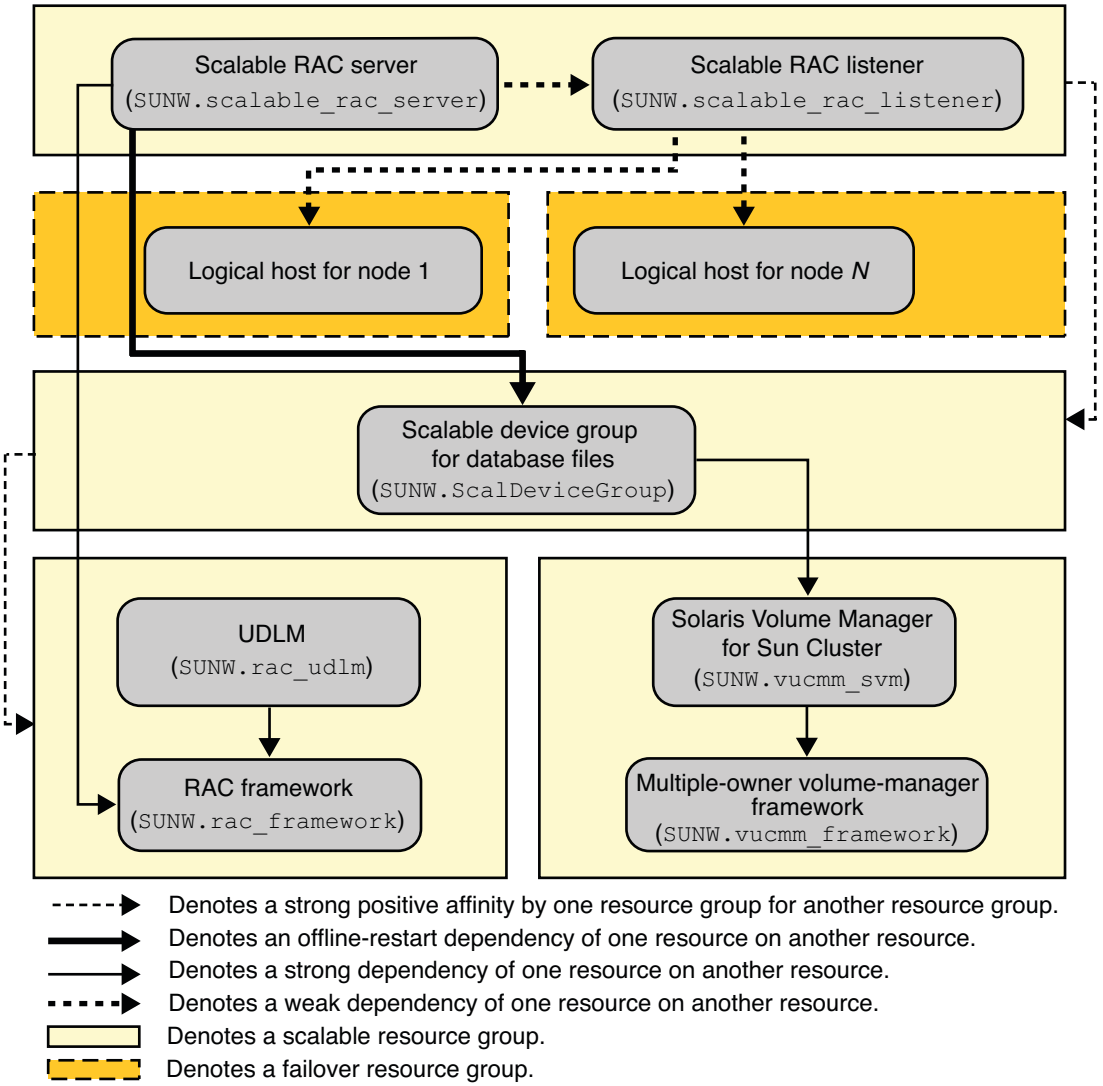


FIGURE A–8 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System

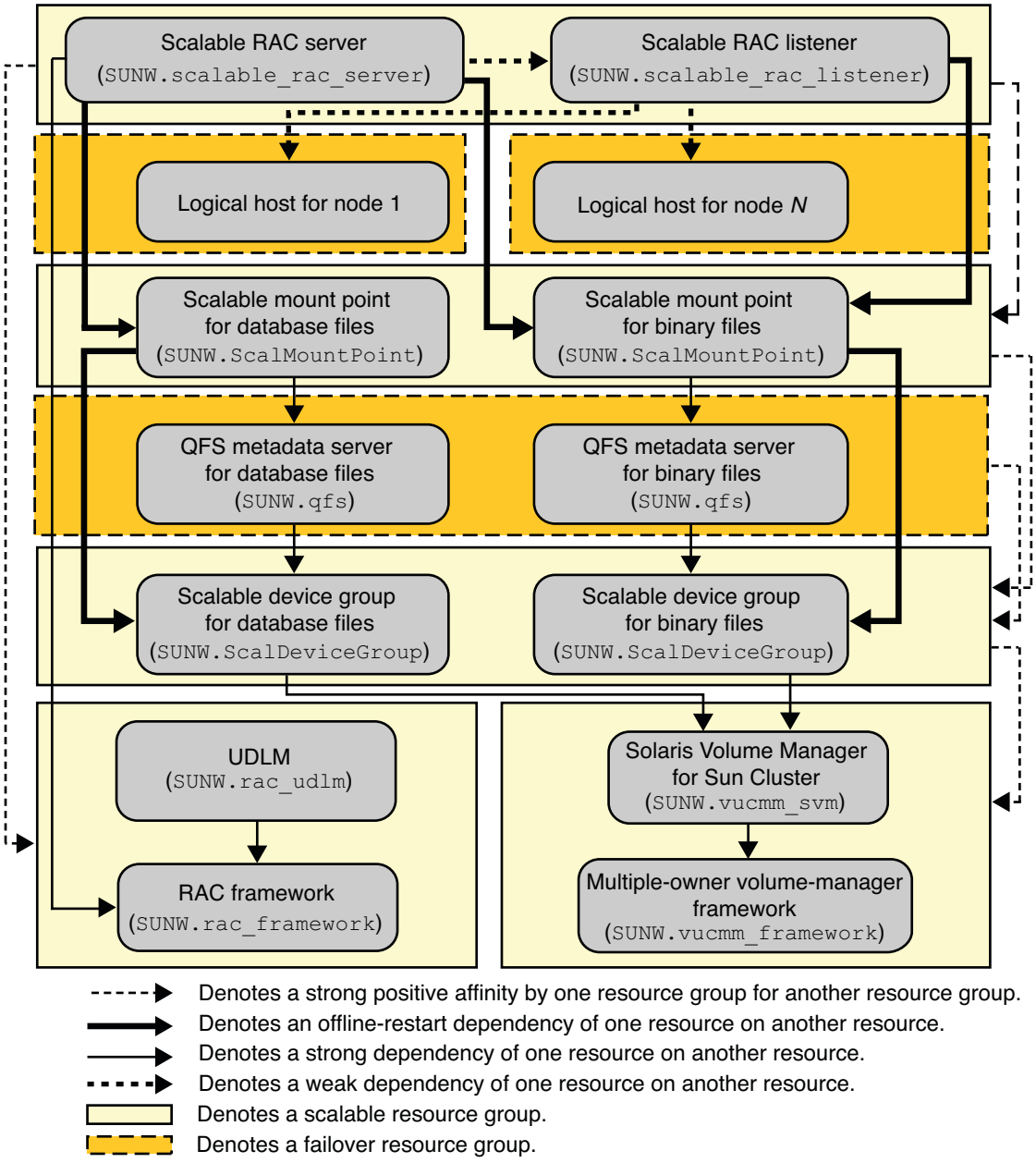
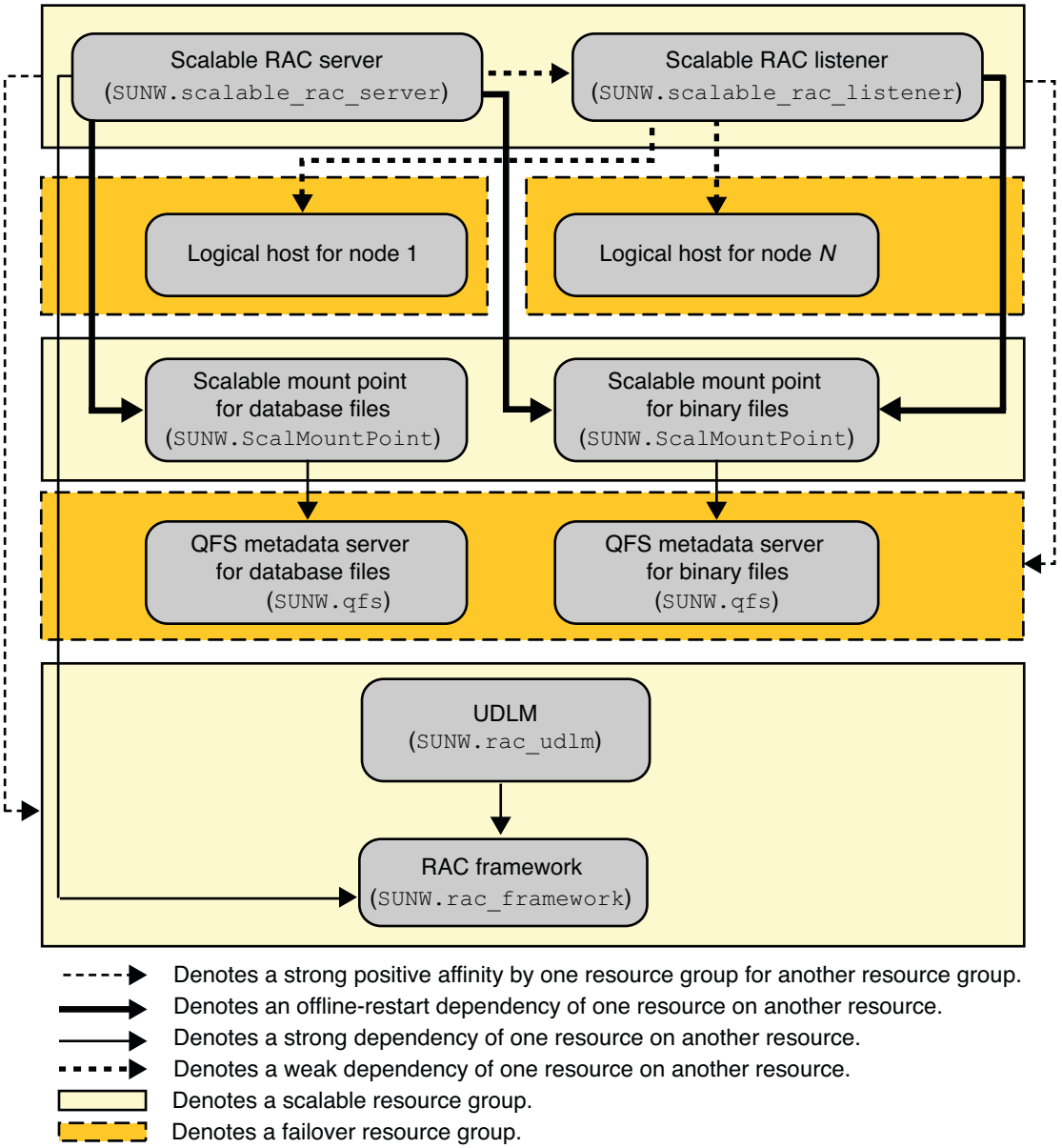


FIGURE A–9 Configuration of Oracle 9*i* With Sun QFS Shared File System and Hardware RAID



Sample Oracle 10*g* or 11*g* Configurations in a Zone Cluster

FIGURE A–10 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster in a Zone Cluster

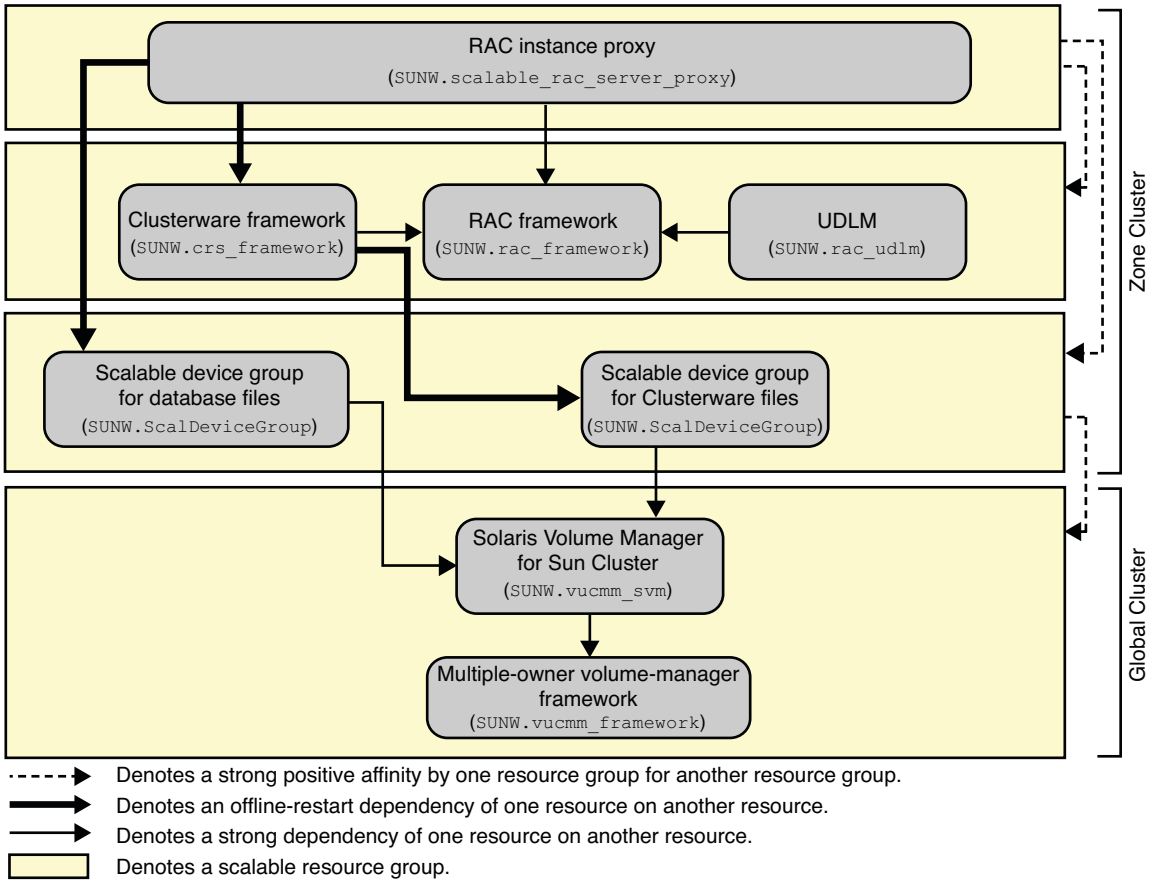


FIGURE A–11 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System in a Zone Cluster

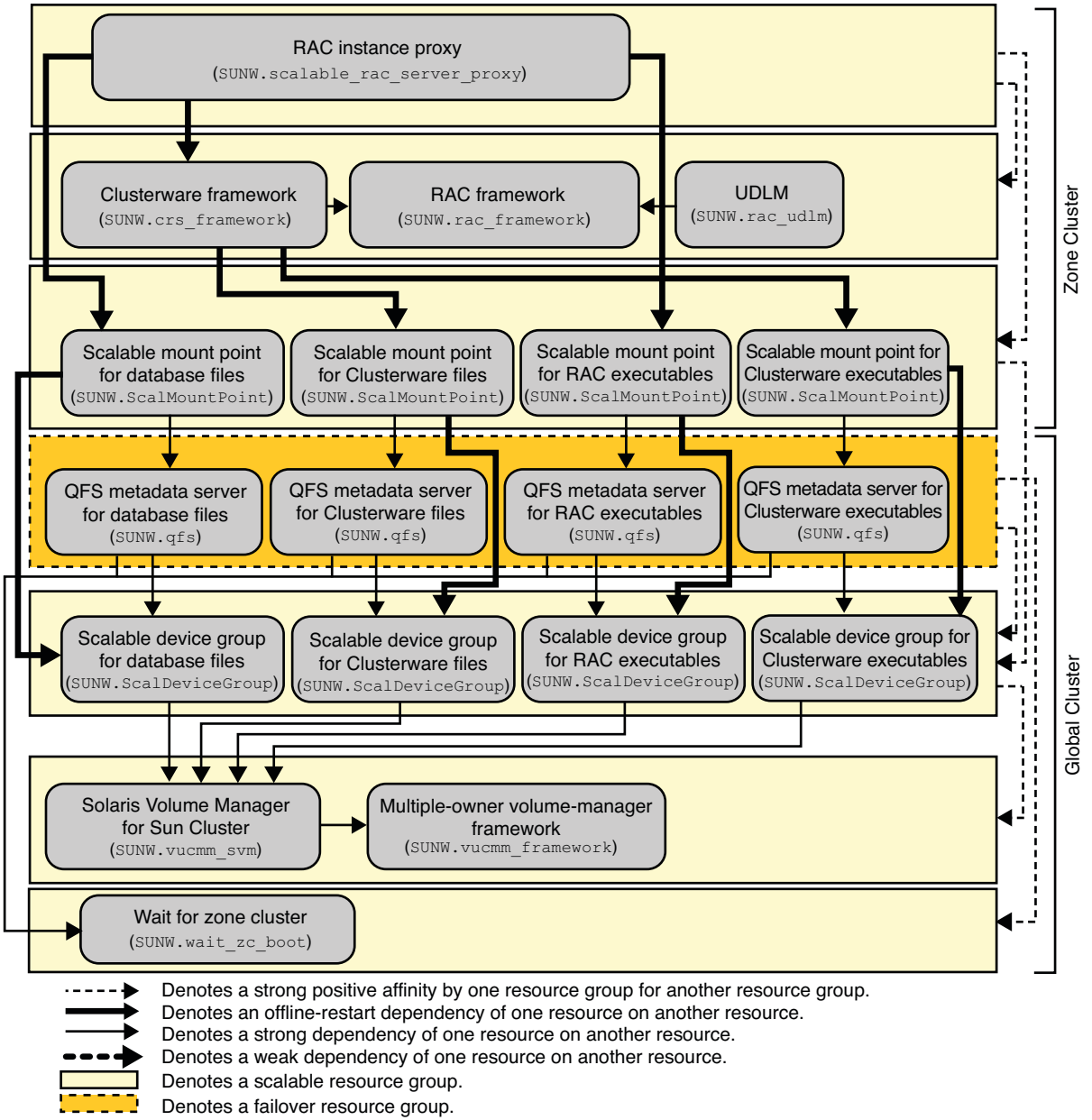


FIGURE A–12 Configuration of Oracle 10*g* or 11*g* With Sun QFS Shared File System and Hardware RAID in a Zone Cluster

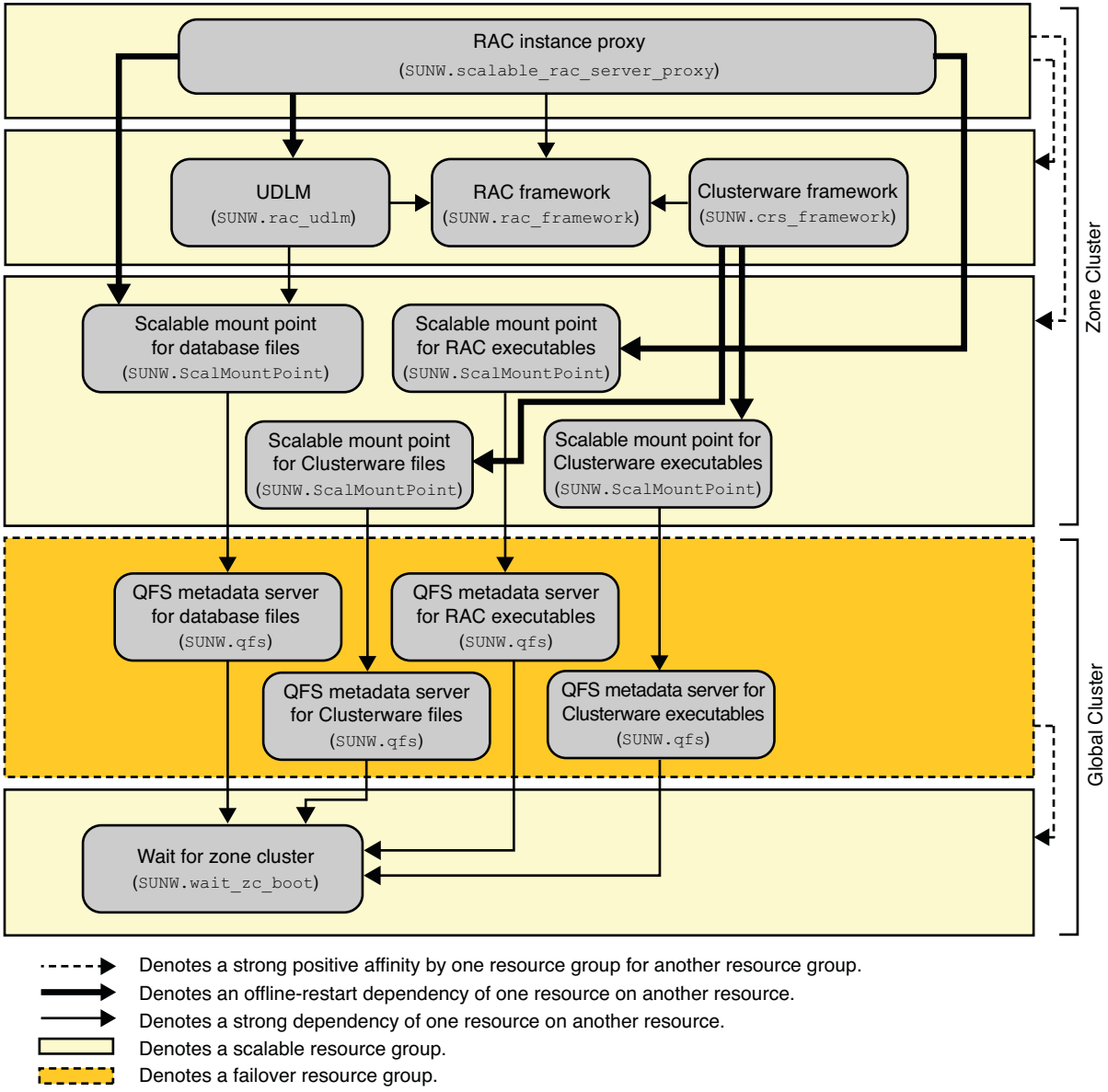


FIGURE A–13 Configuration of Oracle 10*g* or 11*g* With a NAS Device for Oracle RAC in a Zone Cluster

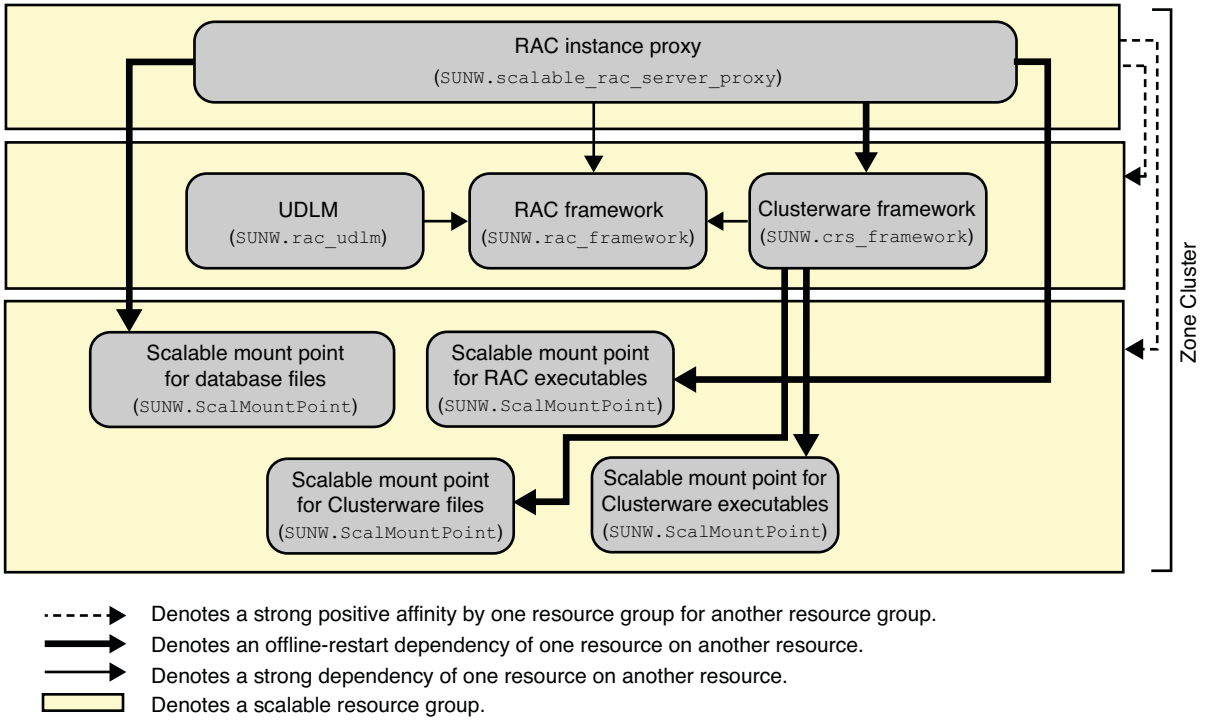
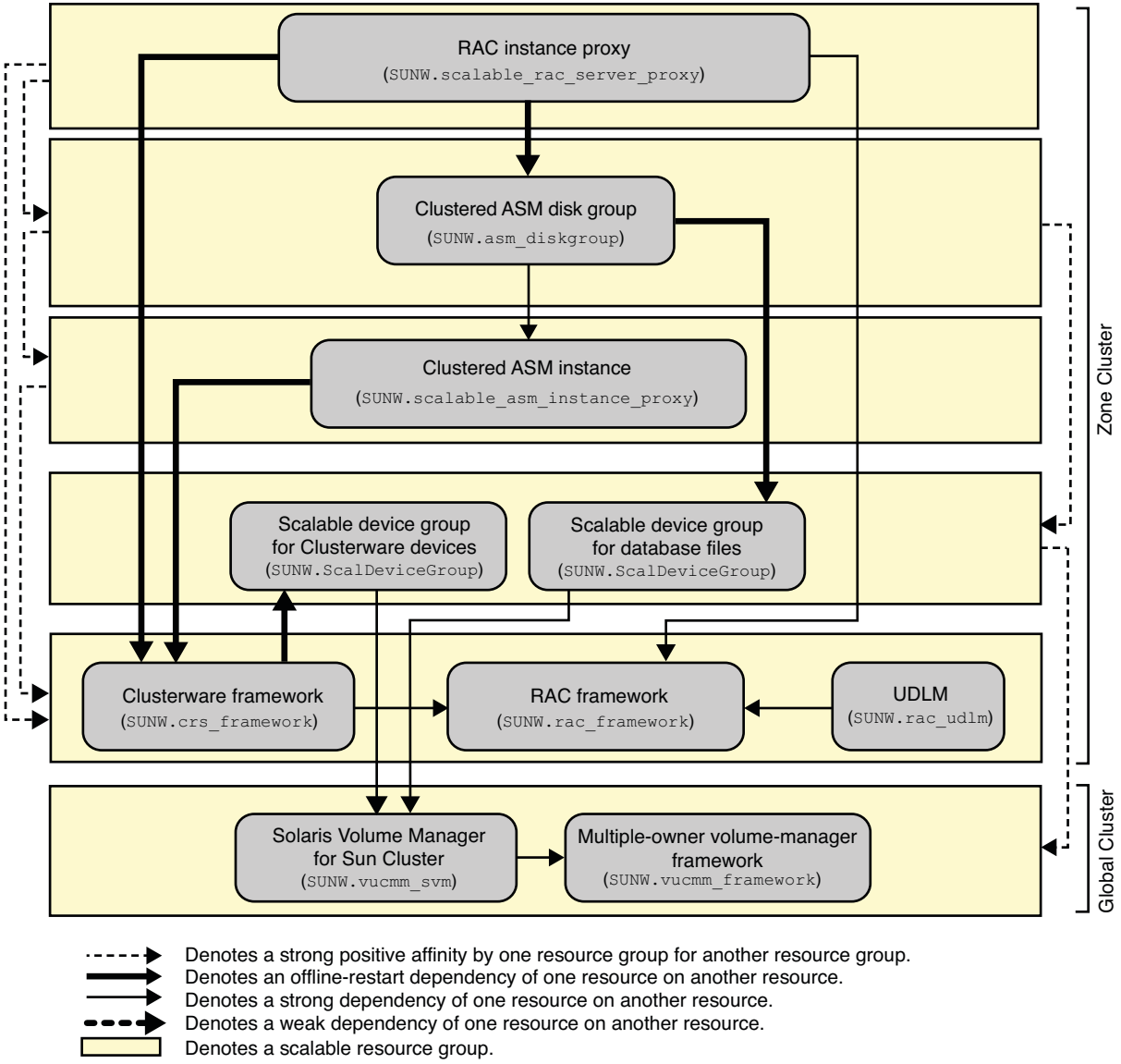
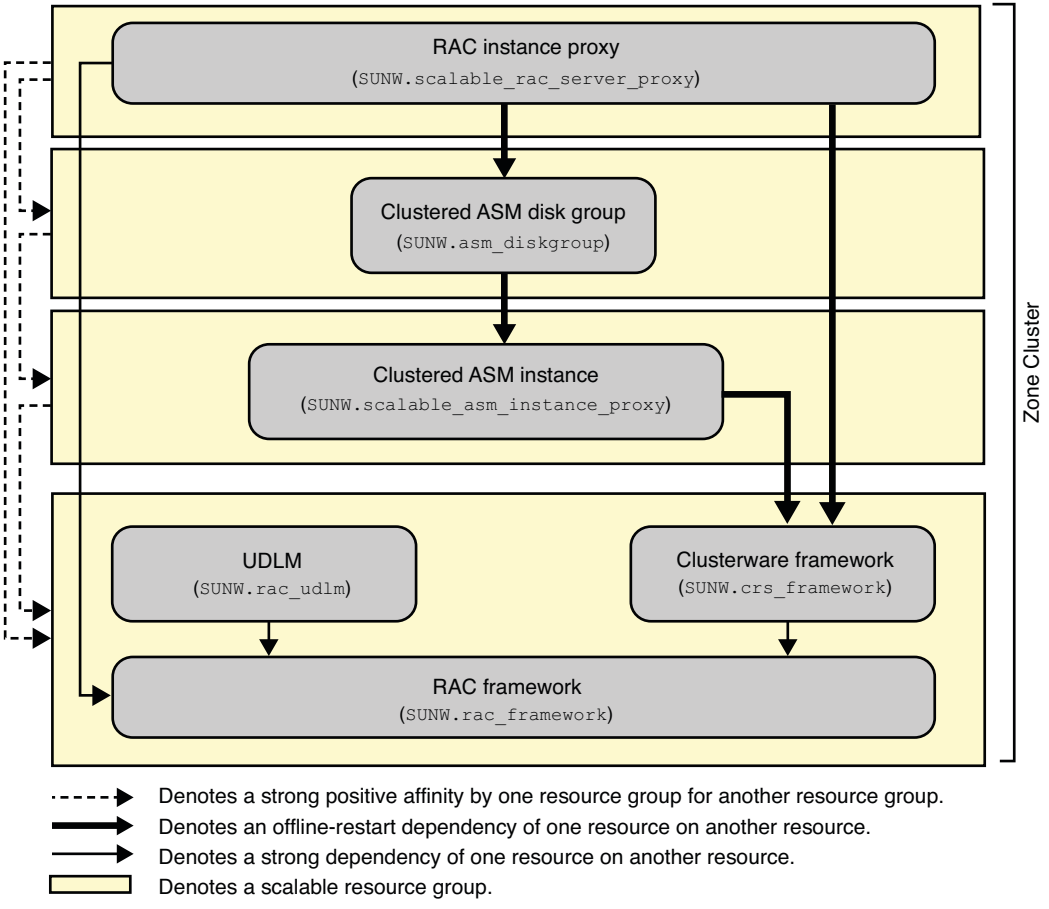


FIGURE A–14 Configuration of Oracle 10*g* or 11*g* With Oracle ASM and Solaris Volume Manager for Sun Cluster in a Zone Cluster



Note – For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type instead of the `SUNW.asm_diskgroup` resource type in the clustered ASM disk group.

FIGURE A–15 Configuration of Oracle 10*g* or 11*g* With Oracle ASM and Hardware RAID in a Zone Cluster



Note – For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type instead of the `SUNW.asm_diskgroup` resource type in the clustered ASM disk group.

Sample Oracle 9*i* Configurations in a Zone Cluster

FIGURE A–16 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster in a Zone Cluster

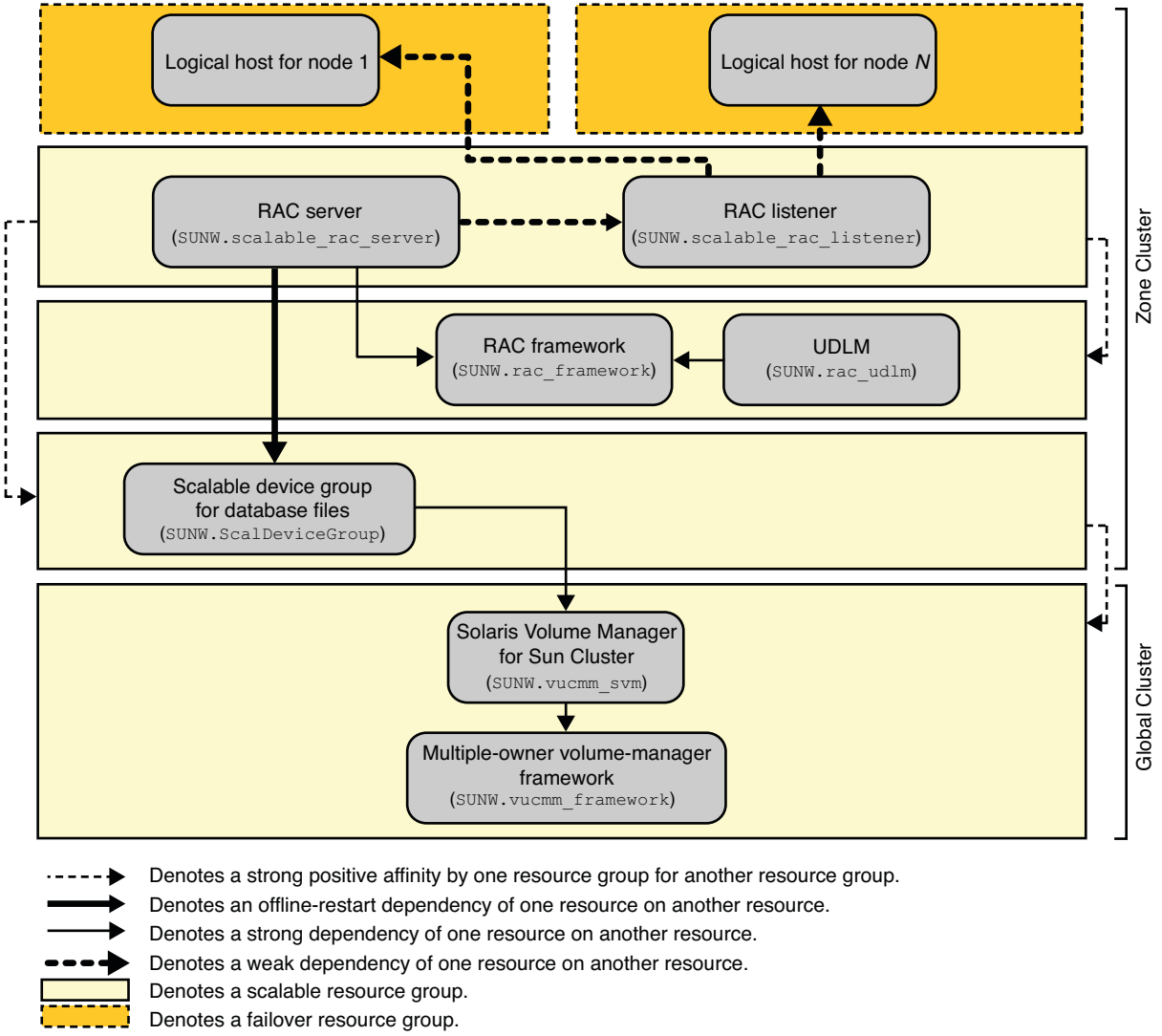


FIGURE A–17 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System in a Zone Cluster

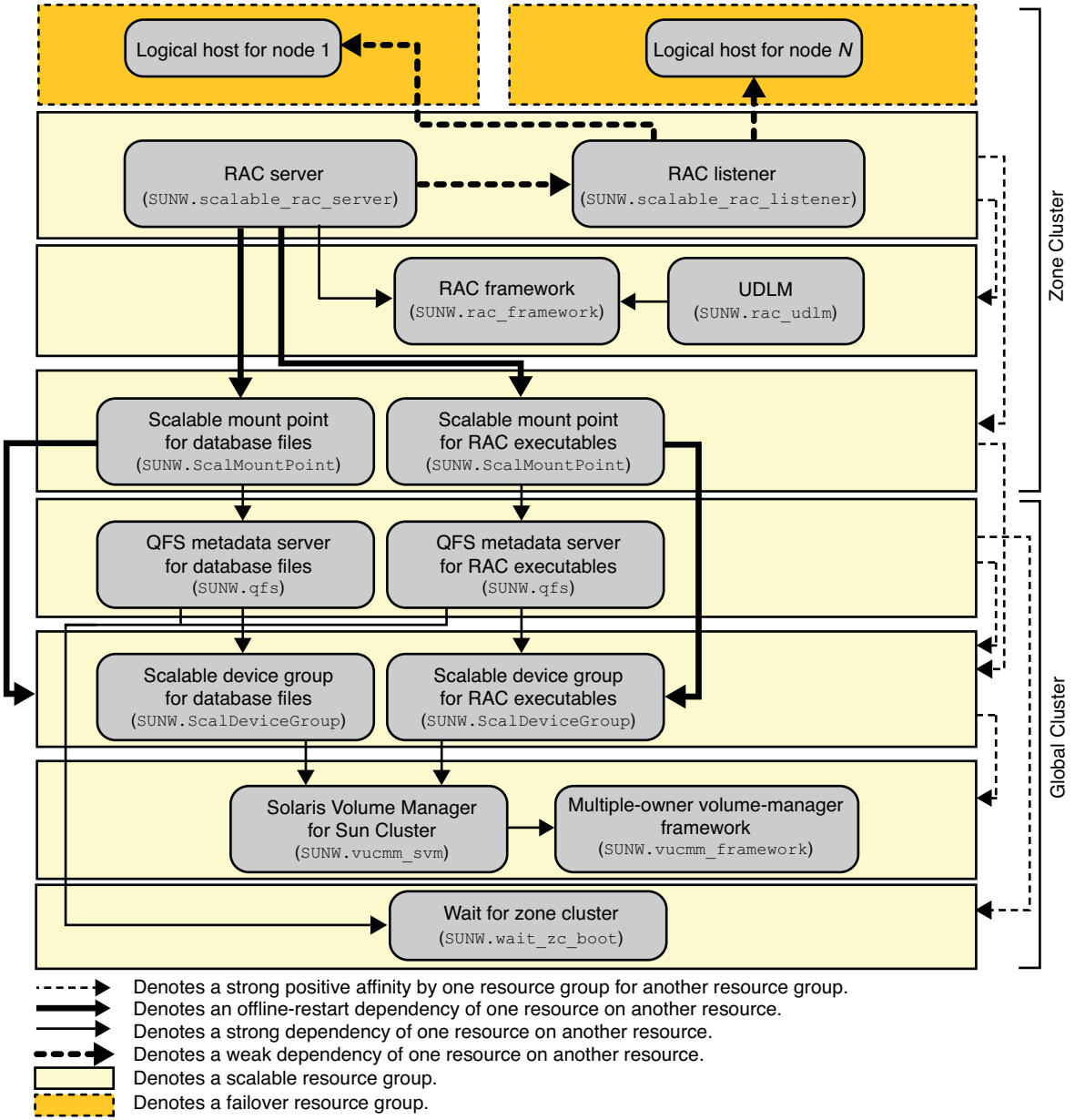
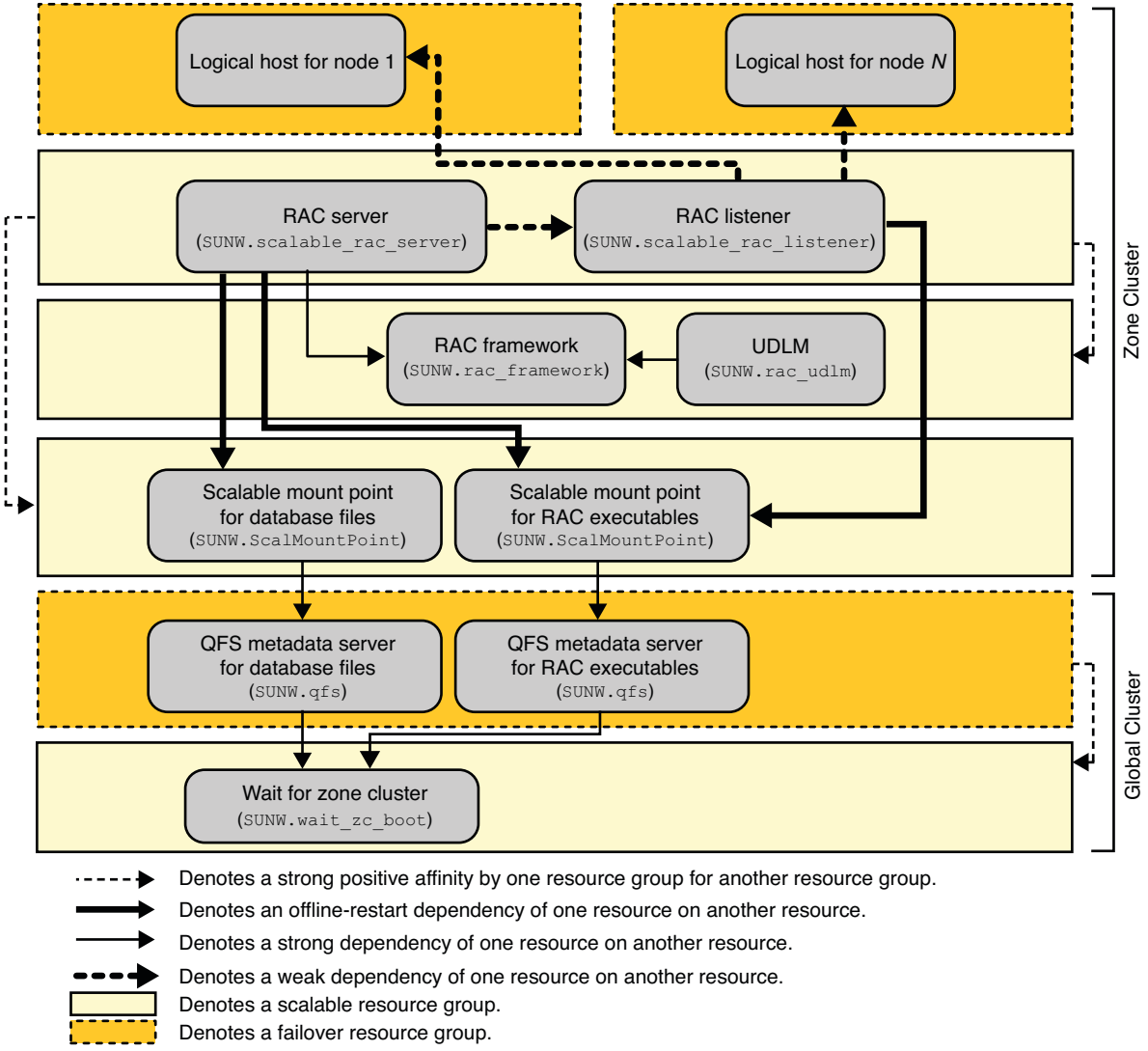


FIGURE A–18 Configuration of Oracle 9*i* With Sun QFS Shared File System and Hardware RAID in a Zone Cluster



Legacy Configurations

This section illustrates configurations that use the RAC framework resource group (`SUNW.rac_framework`) to contain the multiple-owner volume-manager resources, rather than the multiple-owner volume-manager framework resource group (`SUNW.vucmm_framework`). These configurations are still supported in this release, but might become deprecated in a future Oracle Solaris Cluster release.

FIGURE A–19 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster (Legacy)

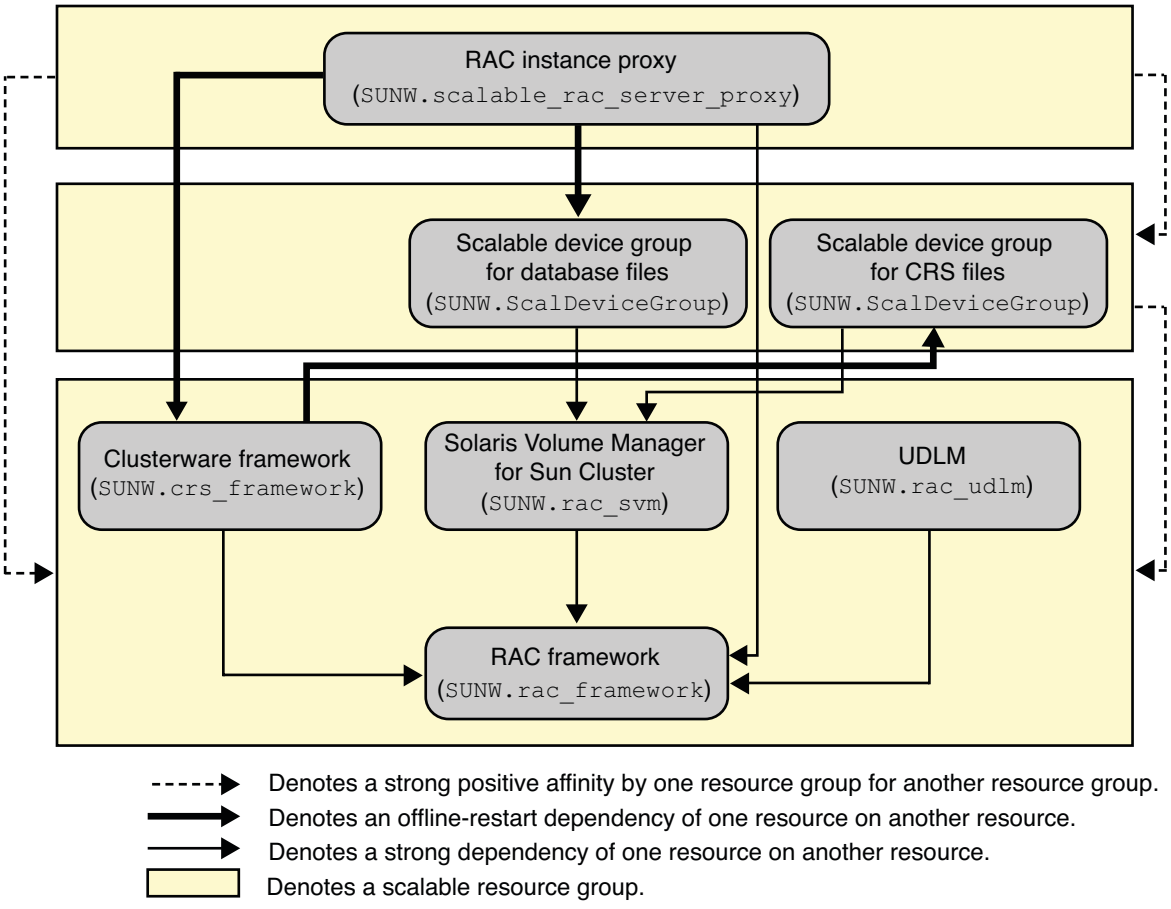


FIGURE A–20 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System (Legacy)

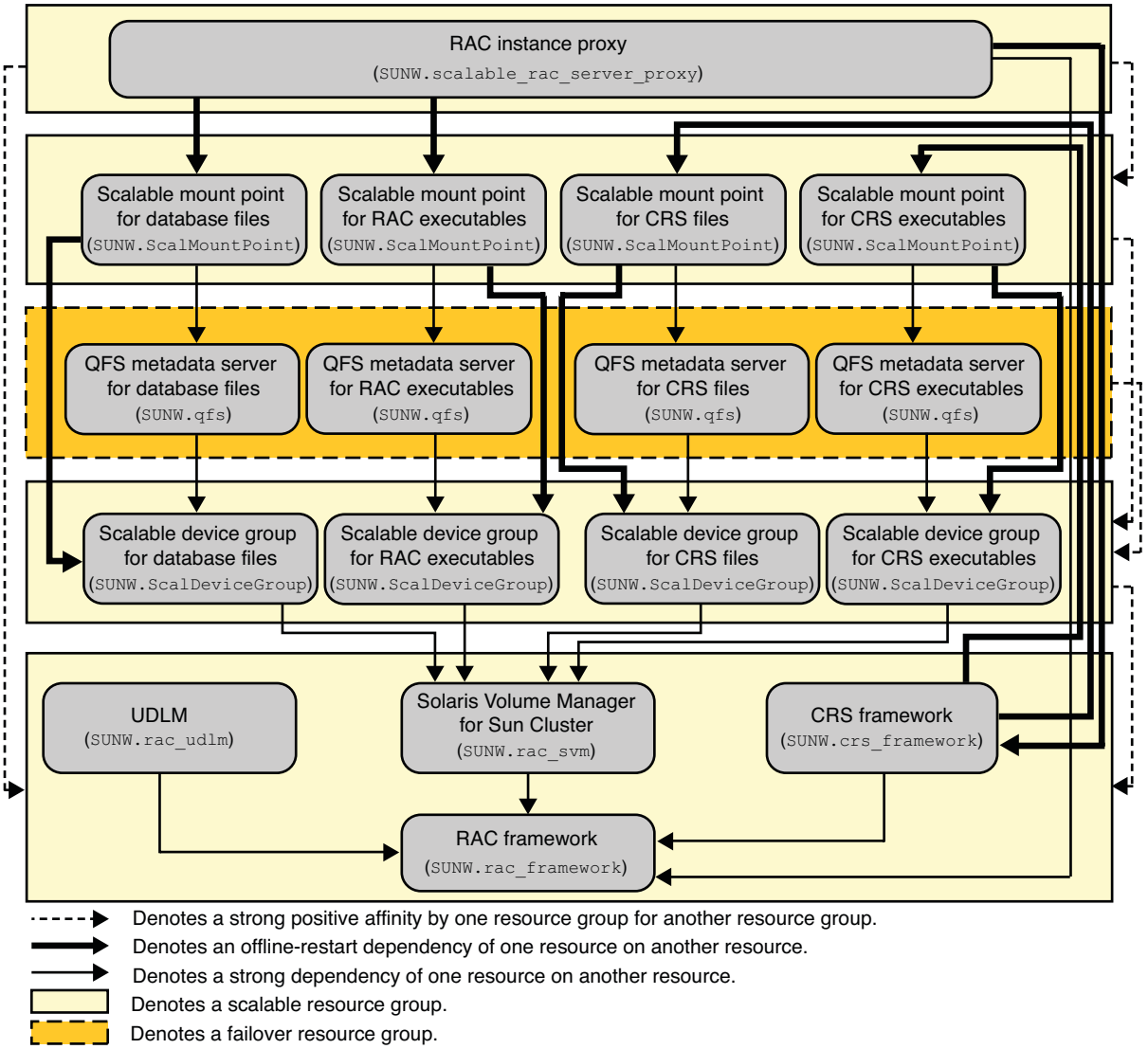


FIGURE A–21 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster (Legacy)

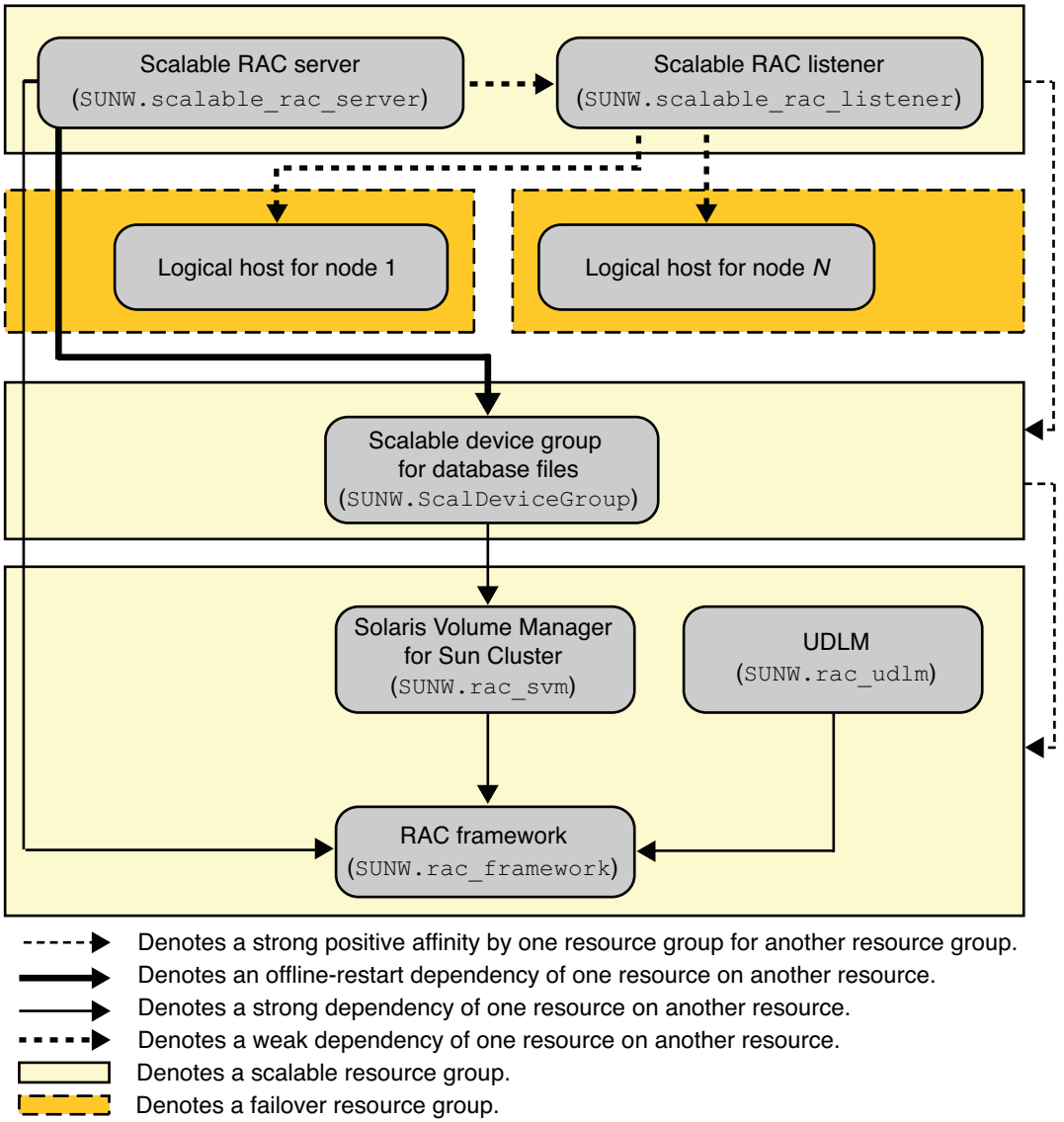


FIGURE A–22 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System (Legacy)

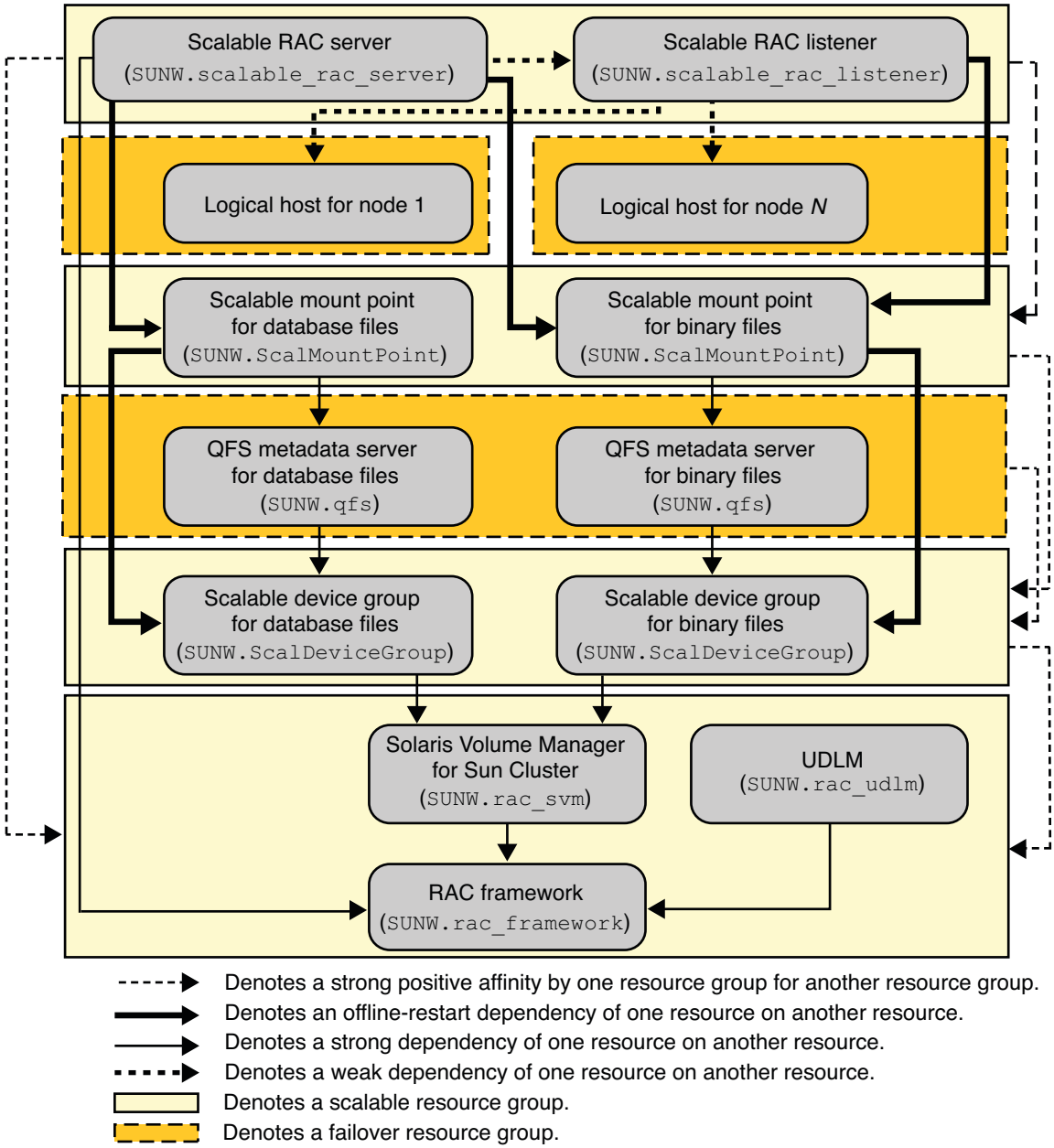


FIGURE A–23 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster in a Zone Cluster (Legacy)

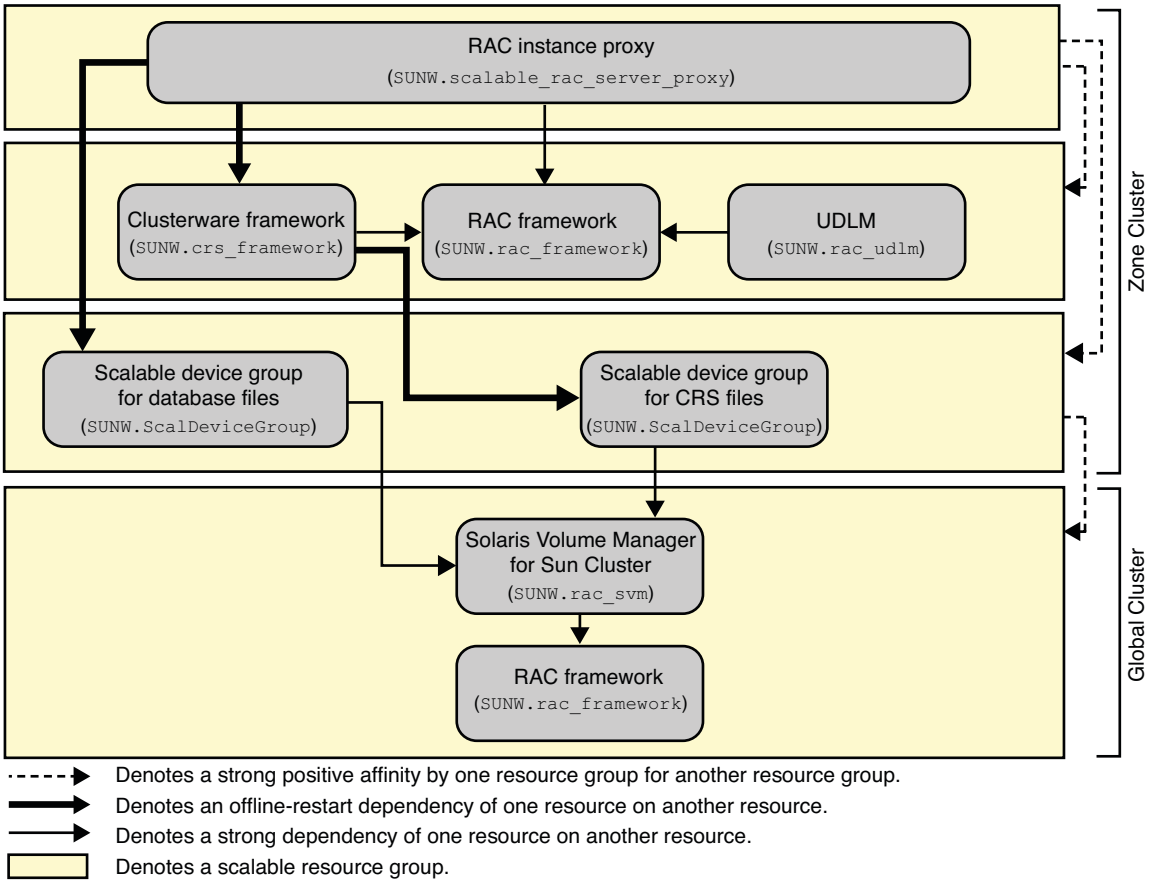


FIGURE A–24 Configuration of Oracle 10*g* or 11*g* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System in a Zone Cluster (Legacy)

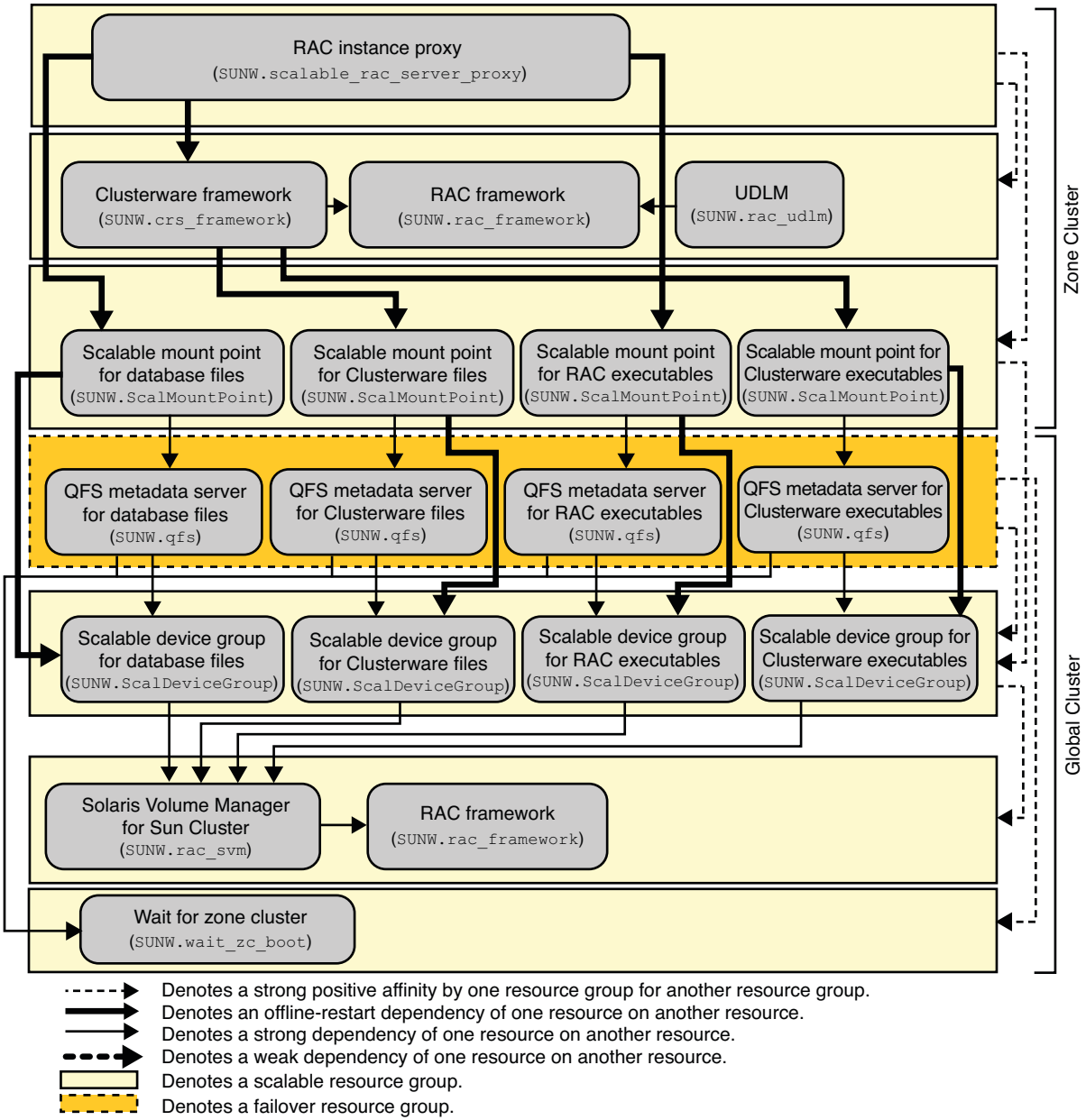


FIGURE A–25 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster in a Zone Cluster (Legacy)

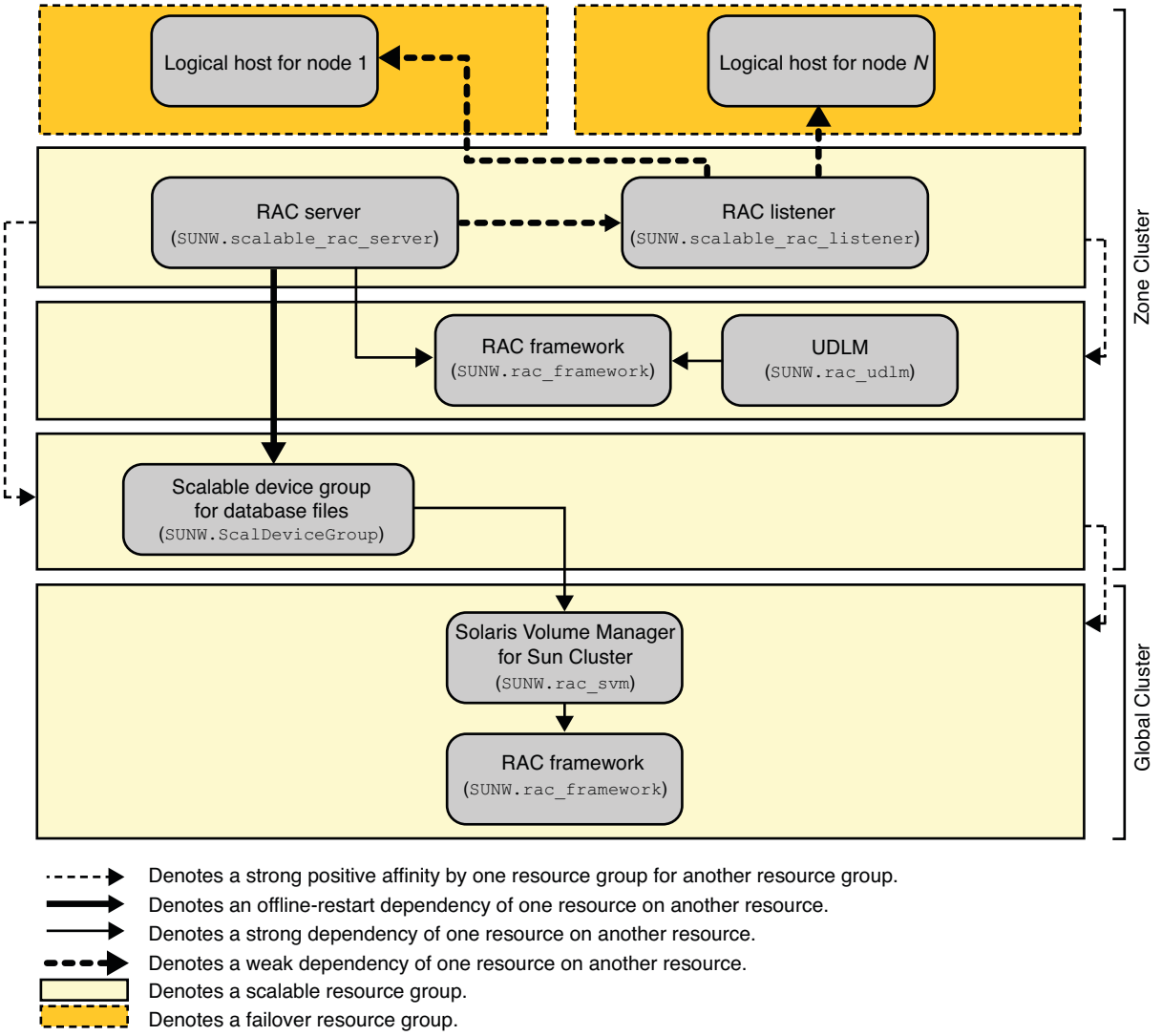
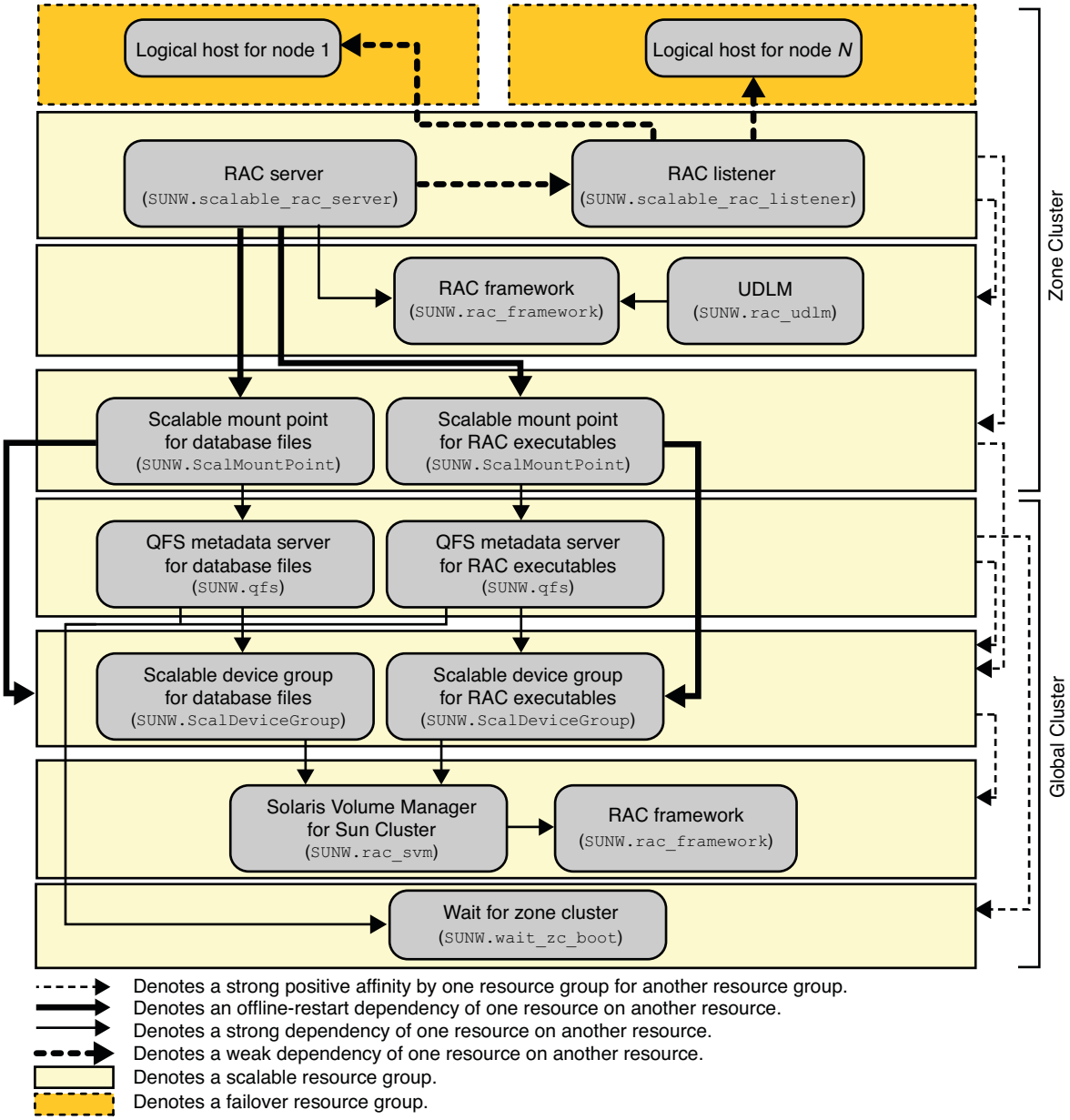


FIGURE A–26 Configuration of Oracle 9*i* With Solaris Volume Manager for Sun Cluster and Sun QFS Shared File System in a Zone Cluster (Legacy)



◆ ◆ ◆ APPENDIX B

Preset Actions for DBMS Errors and Logged Alerts

Preset actions for database management system (DBMS) errors and logged alerts are listed as follows:

- DBMS errors for which an action is preset are listed in Table B–1.
- Logged alerts for which an action is preset are listed in Table B–2.

TABLE B–1 Preset Actions for DBMS Errors

Error Number	Action	Connection State	New State	Message
18	NONE	co	di	Max. number of DBMS sessions exceeded
20	NONE	co	di	Max. number of DBMS processes exceeded
28	NONE	on	di	Session killed by DBA, will reconnect
50	RESTART	*	di	O/S error occurred while obtaining an enqueue. See o/s error.
51	NONE	*	di	timeout occurred while waiting for resource
55	NONE	*	*	maximum number of DML locks in DBMS exceeded
62	STOP	*	di	Need to set DML_LOCKS in init.ora file to value other than 0
107	RESTART	*	di	failed to connect to ORACLE listener process
257	NONE	*	di	archiver error. Connect internal only, until freed.
290	RESTART	*	di	Operating system archival error occurred. Check alert log.
447	RESTART	*	di	fatal error in background process
448	RESTART	*	di	normal completion of background process
449	RESTART	*	di	background process ‘%s’ unexpectedly terminated with error %s
470	RESTART	*	di	Oracle background process died

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
471	RESTART	*	di	Oracle background process died
472	RESTART	*	di	Oracle background process died
473	RESTART	*	di	Oracle background process died
474	RESTART	*	di	SMON died, warm start required
475	RESTART	*	di	Oracle background process died
476	RESTART	*	di	Oracle background process died
477	RESTART	*	di	Oracle background process died
480	RESTART	*	di	LCK* process terminated with error
481	RESTART	*	di	LMON process terminated with error
482	RESTART	*	di	LMD* process terminated with error
602	RESTART	*	di	internal programming exception
604	NONE	on	di	Recursive error
705	RESTART	*	di	inconsistent state during start up
942	NONE	on	*	Warning - V$SYSSTAT not accessible - check grant on V_$SYSSTAT
1001	NONE	on	di	Lost connection to database
1002	NONE	on	*	Internal error in HA-DBMS Oracle
1003	NONE	on	di	Resetting database connection
1012	NONE	on	di	Not logged on
1012	RESTART	di	co	Not logged on
1014	NONE	*	*	ORACLE shutdown in progress
1017	STOP	*	*	Please correct login information in HA-DBMS Oracle database configuration
1031	NONE	on	*	Insufficient privileges to perform DBMS operations - check Oracle user privileges
1033	NONE	co	co	Oracle is in the shutdown or initialization process
1033	NONE	*	di	Oracle is in the shutdown or initialization process
1034	RESTART	co	co	Oracle is not available
1034	RESTART	di	co	Oracle is not available

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
1034	NONE	on	di	Oracle is not available
1035	RESTART	co	co	Access restricted - restarting database to reset
1041	NONE	on	di	
1041	NONE	di	co	
1045	NONE	co	*	Fault monitor user lacks CREATE SESSION privilege logon denied.
1046	RESTART	*	di	cannot acquire space to extend context area
1050	RESTART	*	di	cannot acquire space to open context area
1053	RESTART	*	*	user storage address cannot be read or written
1054	RESTART	*	*	user storage address cannot be read or written
1075	NONE	co	on	Already logged on
1089	NONE	on	di	immediate shutdown in progresss
1089	NONE	*	*	Investigate! Could be hanging!
1090	NONE	*	di	shutdown in progress - connection is not permitted
1092	NONE	*	di	ORACLE instance terminated. Disconnection forced
1513	RESTART	*	*	invalid current time returned by operating system
1542	NONE	on	*	table space is off-line - please correct!
1552	NONE	on	*	rollback segment is off-line - please correct!
1950	NONE	on	*	Insufficient privileges to perform DBMS operations - check Oracle user privileges
2701	STOP	*	*	HA-DBMS Oracle error - ORACLE_HOME did not get set!
2703	RESTART	*	di	
2704	RESTART	*	di	
2709	RESTART	*	di	
2710	RESTART	*	di	
2719	RESTART	*	di	
2721	RESTART	*	*	
2726	STOP	*	*	Could not locate ORACLE executables - check ORACLE_HOME setting
2735	RESTART	*	*	osnfpm: cannot create shared memory segment

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
2811	RESTART	*	*	Unable to attach shared memory segment
2839	RESTART	*	*	Sync of blocks to disk failed.
2840	RESTART	*	*	
2846	RESTART	*	*	
2847	RESTART	*	*	
2849	RESTART	*	*	
2842	RESTART	*	*	Client unable to fork a server - Out of memory
3113	RESTART	co	di	lost connection
3113	NONE	on	di	lost connection
3113	NONE	di	di	lost connection
3114	NONE	*	co	Not connected?
4030	RESTART	*	*	
4032	RESTART	*	*	
4100	RESTART	*	*	communication area cannot be allocated insufficient memory
6108	STOP	co	*	Can’t connect to remote database - make sure SQL*Net server is up
6114	STOP	co	*	Can’t connect to remote database - check SQL*Net configuration
7205	RESTART	*	di	
7206	RESTART	*	di	
7208	RESTART	*	di	
7210	RESTART	*	di	
7211	RESTART	*	di	
7212	RESTART	*	di	
7213	RESTART	*	di	
7214	RESTART	*	di	
7215	RESTART	*	di	
7216	RESTART	*	di	
7218	RESTART	*	di	

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
7219	RESTART	*	*	`slspool: unable to allocate spooler argument buffer.`
7223	RESTART	*	*	`slspool: fork error, unable to spawn spool process. - Resource limit reached`
7224	RESTART	*	*	
7229	RESTART	*	*	
7232	RESTART	*	*	
7234	RESTART	*	*	
7238	RESTART	*	*	`slemcl: close error.`
7250	RESTART	*	*	
7251	RESTART	*	*	
7252	RESTART	*	*	
7253	RESTART	*	*	
7258	RESTART	*	*	
7259	RESTART	*	*	
7263	RESTART	*	*	
7269	RESTART	*	*	
7279	RESTART	*	*	
7280	RESTART	*	*	
7296	RESTART	*	*	
7297	RESTART	*	*	
7306	RESTART	*	*	
7310	RESTART	*	*	
7315	RESTART	*	*	
7321	RESTART	*	*	
7322	RESTART	*	*	
7324	RESTART	*	*	
7325	RESTART	*	*	
7351	RESTART	*	*	

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
7361	RESTART	*	*	
7404	RESTART	*	*	
7414	RESTART	*	*	
7415	RESTART	*	*	
7417	RESTART	*	*	
7418	RESTART	*	*	
7419	RESTART	*	*	
7430	RESTART	*	*	
7455	RESTART	*	*	
7456	RESTART	*	*	
7466	RESTART	*	*	
7470	RESTART	*	*	
7475	RESTART	*	*	
7476	RESTART	*	*	
7477	RESTART	*	*	
7478	RESTART	*	*	
7479	RESTART	*	*	
7481	RESTART	*	*	
9706	RESTART	*	*	
9716	RESTART	*	*	
9718	RESTART	*	*	
9740	RESTART	*	*	
9748	RESTART	*	*	
9747	RESTART	*	*	
9749	RESTART	*	*	
9751	RESTART	*	*	
9755	RESTART	*	*	

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
9757	RESTART	*	*	
9756	RESTART	*	*	
9758	RESTART	*	*	
9761	RESTART	*	*	
9765	RESTART	*	*	
9779	RESTART	*	*	
9829	RESTART	*	*	
9831	RESTART	*	*	
9834	RESTART	*	*	
9836	RESTART	*	*	
9838	RESTART	*	*	
9837	RESTART	*	*	
9844	RESTART	*	*	
9845	RESTART	*	*	
9846	RESTART	*	*	
9847	RESTART	*	*	
9853	RESTART	*	*	
9854	RESTART	*	*	
9856	RESTART	*	*	
9874	RESTART	*	*	
9876	RESTART	*	*	
9877	RESTART	*	*	
9878	RESTART	*	*	
9879	RESTART	*	*	
9885	RESTART	*	*	
9888	RESTART	*	*	
9894	RESTART	*	*	

TABLE B–1 Preset Actions for DBMS Errors *(Continued)*

Error Number	Action	Connection State	New State	Message
9909	RESTART	*	*	
9912	RESTART	*	*	
9913	RESTART	*	*	
9919	RESTART	*	*	
9943	RESTART	*	*	
9947	RESTART	*	*	
9948	RESTART	*	*	
9949	RESTART	*	*	
9950	RESTART	*	*	
12505	STOP	*	*	TNS:listener could not resolve SID given in connect descriptor.Check listener configuration file.
12541	STOP	*	*	TNS:no listener. Please verify connect_string property, listener and TNSconfiguration.
12545	SWITCH	*	*	Please check HA-Oracle parameters. Connect failed because target host or object does not exist
27100	STOP	*	*	Shared memory realm already exists
99999	RESTART	*	di	Monitor detected death of Oracle background processes.

TABLE B–2 Preset Actions for Logged Alerts

Alert String	Action	Connection State	New State	Message
ORA-07265	RESTART	*	di	Semaphore access problem
found dead multi-threaded server	NONE	*	*	Warning: Multi-threaded Oracle server process died (restarted automatically)
found dead dispatcher	NONE	*	*	Warning: Oracle dispatcher process died (restarted automatically)

◆ ◆ ◆ APPENDIX C

Support for Oracle RAC Extension Properties

The extension properties that you can set for each Support for Oracle RAC resource type are listed in the following sections:

You can update some extension properties dynamically. You can update others, however, only when you create or disable a resource. For more information, see "How to Modify an Extension Property That Is Tunable Only When a Resource Is Disabled" on page 138. The Tunable entries indicate when you can update each property.

For information about all system-defined properties, see the `r_properties(5)` man page and the `rg_properties(5)` man page.

For information about `SUNW.qfs` extension properties, see the SUNW.qfs(5) (`http://wikis.sun.com/`

download/attachments/192643233/SUNW.qfs.5.txt?version=2[amp
]modificationDate=1265132574000) man page.

SUNW.asm_diskgroup Extension Properties

asm_diskgroups
: This property specifies the Oracle ASM disk group. If required, more than one Oracle ASM disk group can be specified as a comma separated list.

  Data Type: String array

  Default: Not applicable

  Range: Not applicable

  Tunable: When disabled

Child_mon_level(integer)
: Provides control over the processes that are monitored through the Process Monitor Facility (PMF). This property denotes the level to which the forked child processes are monitored. Omitting this property or setting this property to the default value is the same as omitting the -Coption for pmfadm(1M). All child processes and their descendents are monitored.

  Category: Optional

  Default: -1

  Tunable When disabled

debug_level

Note – All SQL*Plus and srvmgr messages that the Oracle ASM disk group resource issues are written to the log file /var/opt/SUNWscor/oracle_asm/message_log.${RESOURCE}.

This property indicates the level to which debug messages for the Oracle ASM disk group resources are logged. When the debug level is increased, more debug messages are written to the system log /var/adm/messages as follows:

0	No debug messages
1	Function Begin and End messages
2	All debug messages and function Begin/End messages

You can specify a different value of the debug_level extension property for each node that can master the resource.

Data Type: Integer

Range: 0–2

Default: 0

Tunable: Any time

`Failover_Enabled`(boolean)
Allows the resources to fail over. If this property is set to `False`, failover of the resource is disabled. You can use this property to prevent the application resource from initiating a failover of the resource group.

Note – Use the `Failover_mode` property instead of the `Failover_enabled` extension property because `Failover_mode` better controls failover behavior. For more information, see the descriptions of the `LOG_ONLY` and `RESTART_ONLY` values for `Failover_mode` in `r_properties(5)`.

Category: Optional

Default: True

Tunable: When disabled

`Log_level`
Specifies the level, or type, of diagnostic messages that are logged by GDS. You can specify `None`, `Info`, or `Err` for this property. When you specify `None`, diagnostic messages are not logged by GDS. When you specify `Info`, both information and error messages are logged. When you specify `Err`, only error messages are logged.

Category: Optional

Default: Info

Tunable: Any time

`Network_aware`(boolean)
This property specifies whether an application uses the network.

Category: Optional

Default: `False`

Tunable: At creation

`Monitor_retry_count`
This property specifies the number of PMF restart allowed for the fault monitor.

Default: 4

Tunable: Any time

`Monitor_retry_interval`
This property specifies the number of PMF restart allowed for the fault monitor.

Default: 2

Tunable: Any time

`probe_command`(string)
Specifies the command that periodically checks the health of the single instance Oracle ASM.

Category: Required

Default: `/opt/SUNWscor/oracle_asm/bin/asm_control probe -R %RS_NAME -G %RG_NAME -T %RT_NAME`

Tunable: None

`Probe_timeout`(integer)
This property specifies the timeout value, in seconds, for the probe command.

Category: Optional

Default: 30 seconds

Tunable: Any time

`Start_command`(string)
Specifies the command that mounts the Oracle ASM disk group.

Category: Required

Default: `/opt/SUNWscor/oracle_asm/bin/asm_control start -R %RS_NAME -G %RG_NAME -T %RT_NAME`

Tunable: None

`Stop_command`(string)
Specifies the command that dismounts the Oracle ASM disk group.

Category: Required

Default: `/opt/SUNWscor/oracle_asm/bin/asm_control stop -R %RS_NAME -G %RG_NAME -T %RT_NAME`

Tunable: None

`Stop_signal`(integer)
Specifies the command that send stop signal to the Oracle ASM disk group.

Category: Optional

Default: 15

Tunable: When disabled

`Validate_command`(string)
Specifies the absolute path to the command that validates the application, although currently not used.

Category: Optional

Default: NULL

Tunable: When disabled

SUNW.crs_framework Extension Properties

The SUNW.crs_framework resource type has no extension properties.

SPARC: SUNW.rac_cvm Extension Properties

`cvm_abort_step_timeout`
This property specifies the timeout (in seconds) for the abort step of a reconfiguration of the Veritas Volume Manager (VxVM) component of Support for Oracle RAC.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_return_step_timeout`
This property specifies the timeout (in seconds) for the return step of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_start_step_timeout`
This property specifies the timeout (in seconds) for the start step of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step1_timeout`
This property specifies the timeout (in seconds) for step 1 of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step2_timeout`
This property specifies the timeout (in seconds) for step 2 of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step3_timeout`
This property specifies the timeout (in seconds) for step 3 of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 240

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step4_timeout`
This property specifies the timeout (in seconds) for step 4 of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 320

Range: 100 – 99999 seconds

Tunable: Any time

`cvm_stop_step_timeout`
This property specifies the timeout (in seconds) for the stop step of a reconfiguration of the VxVM component of Support for Oracle RAC.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

`vxclust_num_ports`
This property specifies the number of communications ports that the `vxclust` program uses.

Data Type: Integer

Default: 32

Range: 16 – 64

Tunable: When disabled

`vxclust_port`
This property specifies the communications port number that the `vxclust` program uses.

Data Type: Integer

Default: 5568

Range: 1024 – 65535

Tunable: When disabled

`vxconfigd_port`
This property specifies the communications port number that the VxVM component configuration daemon `vxconfigd` uses.

Data Type: Integer

Default: 5560

Range: 1024 – 65535

Tunable: When disabled

`vxkmsgd_port`
This property specifies the communications port number that the VxVM component messaging daemon `vxkmsgd` uses.

Data Type: Integer

Default: 5559

Range: 1024 – 65535

Tunable: When disabled

SUNW.rac_framework Extension Properties

`reservation_timeout`
This property specifies the timeout (in seconds) for the reservation step of a reconfiguration of Support for Oracle RAC.

Data Type: Integer

Default: 325

Range: 100 – 99999 seconds

Tunable: Any time

SUNW.rac_svm Extension Properties

`debug_level`
This property specifies the level to which debug messages from the Solaris Volume Manager for Sun Cluster component are logged. When the debug level is increased, more messages are written to the log files during reconfiguration.

Data Type: Integer

Default: 1, which logs `syslog` messages

Range: 0 – 10

Tunable: Any time

`svm_abort_step_timeout`
This property specifies the timeout (in seconds) for the abort step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_return_step_timeout`
This property specifies the timeout (in seconds) for the return step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_start_step_timeout`
This property specifies the timeout (in seconds) for the start step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step1_timeout`
This property specifies the timeout (in seconds) for step 1 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step2_timeout`
This property specifies the timeout (in seconds) for step 2 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step3_timeout`
This property specifies the timeout (in seconds) for step 3 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step4_timeout`
This property specifies the timeout (in seconds) for step 4 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 120

Range: 100 – 99999 seconds

Tunable: Any time

`svm_stop_step_timeout`
This property specifies the timeout (in seconds) for the stop step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of Support for Oracle RAC.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

SPARC: SUNW.rac_udlm Extension Properties

`failfastmode`
This property specifies the failfast mode of the node on which the Oracle UNIX Distributed Lock Manager (Oracle UDLM) (Oracle UDLM) is running. The failfast mode determines the action that is performed in response to a critical problem with this node. The possible values of this property are as follows:

- `off` – Failfast mode is disabled.
- `panic` – The node is forced to panic.

Data Type: Enum

Default: `panic`

Tunable: Any time

`num_ports`
This property specifies the number of communications ports that the Oracle UDLM uses.

Data Type: Integer

Default: 32

Range: 16 – 64

Tunable: When disabled

`oracle_config_file`
This property specifies the configuration file that the Oracle distributed lock manager (DLM) uses. This file must already exist. The file is installed when the Oracle software is installed. For more information, see the documentation for the Oracle software.

Data Type: String

Default: `/etc/opt/SUNWcluster/conf/udlm.conf`

Tunable: When disabled

`port`
This property specifies the communications port number that the Oracle UDLM uses.

Data Type: Integer

Default: 6000

Range: 1 – 65500

Tunable: When disabled

`schedclassSchedclass`
This property specifies the scheduling class of the Oracle UDLM that is passed to the `priocntl(1)` command. The possible values of this property are as follows:

- `RT` – Real-time
- `TS` – Time-sharing
- `IA` – Interactive

Data Type: Enum

Default: `RT`

Tunable: When disabled

`schedpriority`
This property specifies the scheduling priority of the Oracle UDLM that is passed to the `priocntl` command.

Data Type: Integer

Default: 11

Range: 0 – 59

Tunable: When disabled

`udlm_abort_step_timeout`
This property specifies the timeout (in seconds) for the abort step of an Oracle UDLM reconfiguration.

Data Type: Integer

Default: 325

Range: 30 – 99999 seconds

Tunable: Any time

`udlm_start_step_timeout`
This property specifies the timeout (in seconds) for the start step of an Oracle UDLM reconfiguration.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`udlm_step1_timeout`
This property specifies the timeout (in seconds) for step 1 of an Oracle UDLM reconfiguration.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`udlm_step2_timeout`
This property specifies the timeout (in seconds) for step 2 of an Oracle UDLM reconfiguration.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`uUdlm_step3_timeout`
This property specifies the timeout (in seconds) for step 3 of an Oracle UDLM reconfiguration.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`udlm_step4_timeout`
This property specifies the timeout (in seconds) for step 4 of an Oracle UDLM reconfiguration.

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`udlm_step5_timeout`
This property specifies the timeout (in seconds) for step 5 of an Oracle UDLM reconfiguration.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

SUNW.scalable_asm_diskgroup_proxy Extension Properties

`asm_diskgroups`
This property specifies the single-instance Oracle ASM disk group. If required, more than one single-instance Oracle ASM disk group can be specified as a comma separated list.

Data Type: String array

Default: Not applicable

Range: Not applicable

Tunable: When disabled

`debug_level` (integer)

Note – All SQL*Plus and srvmgr messages that the Oracle ASM disk group resource issues are written to the log file `/var/opt/SUNWscor/oracle_asm/message_log.${RESOURCE}`.

This property indicates the level to which debug messages for the Oracle ASM disk group resources are logged. When the debug level is increased, more debug messages are written to the system log `/var/adm/messages` as follows:

0	No debug messages
1	Function Begin and End messages
2	All debug messages and function Begin/End messages

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Range: 0–2

Default: 0

Tunable: Any time

`Proxy_probe_interval`(integer)
Specifies the timeout value, in seconds, that the proxy monitor uses when checking the status of the clustered Oracle ASM disk group resource for which this resource is acting as a proxy..

Range: 5–120

Default: 30

Tunable: Any time

`proxy_probe_timeout`(integer)
This property specifies the timeout value, in seconds, for the probe command.

Range: 5–120

Default: 60

Tunable: Any time

SUNW.scalable_asm_instance Extension Properties

`Child_mon_level`(integer)
Provides control over the processes that are monitored through the Process Monitor Facility (PMF). This property denotes the level to which the forked child processes are monitored. Omitting this property or setting this property to the default value is the same as omitting the `-C`option for `pmfadm`(1M). All child processes and their descendents are monitored.

Category: Optional

Default: -1

Tunable: When disabled

`debug_level`

Note – All SQL*Plus and srvmgr messages that the clustered Oracle ASM instance proxy resource issues are written to the log file `/var/opt/SUNWscor/oracle_asm/message_log.${RESOURCE}`.

This property indicates the level to which debug messages from the monitor for the clustered Oracle ASM instance proxy are logged. When the debug level is increased, more debug messages are written to the system log `/var/adm/messages` as follows:

0 No debug messages

1 Function Begin and End messages

2 All debug messages and function Begin/End messages

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Data Type: Integer

Range: 0–2

Default: 0

Tunable: Any time

`Failover_Enabled`(boolean)
Allows the resources to fail over. If this property is set to `False`, failover of the resource is disabled. You can use this property to prevent the application resource from initiating a failover of the resource group.

Note – Use the `Failover_mode` property instead of the `Failover_enabled` extension property because `Failover_mode` better controls failover behavior. For more information, see the descriptions of the `LOG_ONLY` and `RESTART_ONLY` values for `Failover_mode` in `r_properties(5)`.

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Category: Optional

Default: True

Tunable: When disabled

`Log_level`
Specifies the level, or type, of diagnostic messages that are logged by GDS. You can specify `None`, `Info`, or `Err` for this property. When you specify `None`, diagnostic messages are not logged by GDS. When you specify `Info`, both information and error messages are logged. When you specify `Err`, only error messages are logged.

Category: Optional

Default: Info

Tunable: Any time

`Network_aware`(boolean)
This property specifies whether an application uses the network.

Category: Optional

Default: `False`

Tunable: At creation

`Monitor_retry_count`
This property specifies the number of PMF restart allowed for the fault monitor.

Default: 4

Tunable: Any time

`Monitor_retry_interval`
This property specifies the number of PMF restart allowed for the fault monitor.

Default: 2

Tunable: Any time

`oracle_home`
This property specifies the full path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`oracle_sid`
This property specifies the Oracle System Identifier (SID). The Oracle SID uniquely identifies the single instance Oracle ASM on the node where the instance is running.

Data Type: String

Range: Not applicable

Default: +ASM

Tunable: When disabled

`probe_command`(string)
Specifies the command that periodically checks the health of the single instance Oracle ASM.

Category: Required

Default: `/opt/SUNWscor/oracle_asm/bin/asm_control probe -R %RS_NAME -G %RG_NAME -T %RT_NAME`

TunableNone

`Probe_timeout`(integer)
This property specifies the timeout value, in seconds, for the probe command.

Category: Optional

Default: 30 seconds

Tunable: Any time

`Start_command`(string)
Specifies the command that starts the single instance Oracle ASM.

Category: Required

Default: `/opt/SUNWscor/oracle_asm/bin/asm_control start -R %RS_NAME -G %RG_NAME -T %RT_NAME`

Tunable: None

`Stop_command`(string)
Specifies the command that stops the single instance Oracle ASM.

Category: Required

Default: `/opt/SUNWscor/oracle_asm/bin/asm_control stop -R %RS_NAME -G %RG_NAME -T %RT_NAME`

Tunable: None

`Stop_signal`(integer)
Specifies the command that stops the single instance Oracle ASM.

Category: Optional

Default: `15`

Tunable: When disabled

`Validate_command`(string)
Specifies the absolute path to the command that validates the application, although currently not used.

Category: Optional

Default: `Null`

Tunable: When disabled

SUNW.scalable_asm_instance_proxy Extension Properties

`asm_diskgroups`
This property specifies the Oracle ASM disk group. If required, more than one Oracle ASM disk group can be specified as a comma separated list.

Data Type: String array

Default: Not applicable

Range: Not applicable

Tunable: When disabled

`crs_home`
This property specifies the full path to the Oracle Clusterware home directory. The Oracle Clusterware home directory contains the binary files, log files and parameter files for the Oracle Clusterware software.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`debug_level`

Note – All SQL*Plus and srvmgr messages that the clustered ASM instance proxy resource issues are written to the log file `/var/opt/SUNWscor/oracle_asm/message_log.${RESOURCE}`.

This property indicates the level to which debug messages from the monitor for the Oracle clustered ASM instance proxy are logged. When the debug level is increased, more debug messages are written to the system log `/var/adm/messages` as follows:

0 No debug messages

1 Function Begin and End messages

2 All debug messages and function Begin/End messages

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Data Type: Integer

Range: 0–2

Default: 0

Tunable: Any time

`oracle_home`

This property specifies the full path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`oracle_sid`
This property specifies the Oracle System Identifier (SID). The Oracle SID uniquely identifies the clustered Oracle ASM database instance on the node where the instance is running.

You *must* specify a different value of the `oracle_sid` extension property for each node that can master the resource. The value for each node must correctly identify the instance that is running on the node.

Data Type: String

Range: Not applicable

Default: `NULL`

Tunable: When disabled

`proxy_probe_timeout`
This property specifies the timeout value, in seconds, that the proxy monitor uses when checking the status of the Oracle Clusterware resource for which this resource is acting as a proxy.

Data Type: Integer

Range: 5–120

Default: 60

Tunable: Any time

`proxy_probe_interval`
This property specifies the interval , in seconds, between probes of the Oracle Clusterware resource for which this resource is acting as a proxy.

Data Type: Integer

Range: 5–120

Default: 60

Tunable: Any time

SUNW.scalable_rac_listener Extension Properties

`debug_level`
This property indicates the level to which debug messages from the Oracle RAC listener component are logged. When the debug level is increased, more debug messages are written to the log files. These messages are logged to the file `/var/opt/SUNWscor/scalable_rac_listener/message_log.`*rs*, where *rs* is the name of the resource that represents the Oracle RAC listener component.

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Data Type: Integer

Range: 0–100

Default: 1, which logs `syslog` messages

Tunable: Any time

`listener_name`
This property specifies the name of the Oracle listener instance that is to be started on the node where the instance is running. This name must match the corresponding entry in the `listener.ora` configuration file.

You can specify a different value of the `listener_name` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: `LISTENER`

Tunable: When disabled

`oracle_home`
This property specifies the full path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`probe_timeout`
This property specifies the timeout value, in seconds, that the fault monitor uses when checking the status of an Oracle RAC listener.

Data Type: Integer

Range: 1–99999

Default: 300

Tunable: Any time

`user_env`
This property specifies the name of the file that contains the environment variables that are to be set before the listener starts up or shuts down. You must define all environment variables whose values differ from Oracle defaults in this file.

For example, a user's `listener.ora` file might not be located under the `/var/opt/oracle` directory or the *oracle-home*`/network/admin` directory. In this situation, the `TNS_ADMIN` environment variable must be defined.

The definition of each environment variable that is defined must follow the format *variable-name*=*value*. Each definition must start on a new line in the environment file.

You can specify a different value of the `user_env` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: Any time

SUNW.scalable_rac_server Extension Properties

`alert_log_file`
This property specifies the absolute path of the Oracle alert log file. The Oracle software logs alerts in this file. The Oracle RAC server fault monitor scans the alert log file for new alerts at the following times:

- When the RAC server fault monitor is started
- Each time that the RAC server fault monitor queries the health of the server

If an action is defined for a logged alert that the RAC server fault monitor detects, the RAC server fault monitor performs the action in response to the alert.

Preset actions for logged alerts are listed in Appendix B, "Preset Actions for DBMS Errors and Logged Alerts." To change the action that the RAC server fault monitor performs, customize the server fault monitor as explained in "Customizing the Oracle 9i RAC Server Fault Monitor" on page 145.

You can specify a different value of the `alert_log_file` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: `NULL`

Tunable: Any time

`connect_cycle`
This property specifies the number of fault monitor probe cycles that are performed before the fault monitor disconnects from the database.

You can specify a different value of the `connect_cycle` extension property for each node that can master the resource.

Data Type: Integer

Range: 0–99999

Default: 5

Tunable: Any time

`connect_string`
This property specifies the Oracle database user ID and password that the fault monitor uses to connect to the Oracle database. This property is specified as follows:

userid/*password*

userid
Specifies the Oracle database user ID that the fault monitor uses to connect to the Oracle database.

password
Specifies the password that is set for the Oracle database user *userid*.

The system administrator must define the database user ID and password for the fault monitor during the setup of Oracle RAC. To use Solaris authentication, type a slash (`/`) instead of a user ID and password.

You can specify a different value of the `connect_string` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: `NULL`

Tunable: Any time

`custom_action_file`
This property specifies the absolute path of the file that defines the custom behavior of the Oracle RAC server fault monitor.

You can specify a different value of the `custom_action_file` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: Empty string

Tunable: Any time

`debug_level`
This property indicates the level to which debug messages from the monitor for the Oracle RAC proxy server are logged. When the debug level is increased, more debug messages are written to the log files.

The messages are logged to files in the directory `/var/opt/SUNWscor/oracle_server/proxy`*rs*. Messages for server-side components and client-side components of the proxy server resource are written to separate files:

- Messages for server-side components are written to the file `message_log.`*rs*.
- Messages for client-side components are written to the file `message_log.client.`*rs*.

In these file names and directory names, *rs* is the name of the resource that represents the Oracle RAC server component.

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Data Type: Integer

Range: 0–100

Default: 1, which logs `syslog` messages

Tunable: Any time

`oracle_home`
This property specifies the full path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`oracle_sid`
This property specifies the Oracle System Identifier (SID). The Oracle SID uniquely identifies the Oracle Real Application Cluster database instance on the node where the instance is running.

You *must* specify a different value of the `oracle_sid` extension property for each node that can master the resource. The value for each node must correctly identify the instance that is running on the node.

Data Type: String

Range: Not applicable

Default: `NULL`

Tunable: When disabled

`parameter_file`
This property specifies the full path to the Oracle parameter file. This file contains parameters that are to be set when the Oracle database is started. This property is optional. If you do not set this property, the default parameter file that is specified by Oracle is used, namely: *oracle-home*`/dbs/init`*oracle-sid*`.ora`.

oracle-home
Specifies the Oracle home directory.

oracle-sid
Specifies the Oracle system identifier of the database instance for which the file is to be used.

You can specify a different value of the `parameter_file` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: Any time

`probe_timeout`
This property specifies the timeout value, in seconds, that the fault monitor uses when checking the status of an Oracle RAC server.

Data Type: Integer

Range: 1–99999

Default: 300

Tunable: Any time

`user_env`
This property specifies the name of the file that contains the environment variables that are to be set before the database starts up or shuts down. You must define all environment variables whose values differ from Oracle defaults in this file.

For example, a user's `listener.ora` file might not be located under the `/var/opt/oracle` directory or the *oracle-home*`/network/admin` directory. In this situation, the `TNS_ADMIN` environment variable must be defined.

The definition of each environment variable that is defined must follow the format *variable-name*=*value*. Each definition must start on a new line in the environment file.

You can specify a different value of the `user_env` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: Any time

`wait_for_online`
This property specifies whether the `START` method of the Oracle RAC server resource waits for the database to be online before the `START` method exits. The permitted values for this property are as follows:

`True` Specifies that the `START` method of the Oracle RAC server resource waits for the database to be online before the `START` method exits.

`False` Specifies that the `START` method runs the commands to start the database but does not wait for the database to come online before the `START` method exits.

Data Type: Boolean

Range: Not applicable

Default: `True`

Tunable: Any time

SUNW.scalable_rac_server_proxy Extension Properties

`client_retries`
This property specifies the maximum number of attempts by the resource's remote procedure call (RPC) client to connect to the proxy daemon.

Data Type: Integer

Range: 1–25

Default: 3

Tunable: When disabled

`client_retry_interval`
This property specifies the interval, in seconds, between attempts by the resource's RPC client to connect to the proxy daemon.

Data Type: Integer

Range: 1–3600

Default: 5

Tunable: When disabled

`crs_home`
This property specifies the directory in which the Oracle Clusterware software is located.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`db_name`
This property specifies the name that uniquely identifies the specific Oracle RAC database that is associated with this resource. This identifier distinguishes the database from other databases that might run simultaneously on your system. The name of the Oracle RAC database is specified during the installation of Oracle RAC.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`debug_level`
This property indicates the level to which debug messages from the component for the Oracle RAC proxy server are logged. When the debug level is increased, more debug messages are written to the log files. These messages are logged to the file `/var/opt/SUNWscor/scalable_rac_server_proxy/message_log.`*rs*, where *rs* is the name of the resource that represents the Oracle RAC proxy server component.

You can specify a different value of the `debug_level` extension property for each node that can master the resource.

Data Type: Integer

Range: 0–100

Default: 1, which logs `syslog` messages

Tunable: Any time

`monitor_probe_interval`
This property specifies the interval, in seconds, between probes of the Oracle Clusterware resource for which this resource is acting as a proxy.

Data Type: Integer

Range: 10–3600

Default: 300

Tunable: Any time

`oracle_home`
This property specifies the full path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: When disabled

`oracle_sid`
This property specifies the Oracle System Identifier (SID). The Oracle SID uniquely identifies the Oracle RAC database instance on the node where the instance is running.

You *must* specify a different value of the `oracle_sid` extension property for each node that can master the resource. The value for each node must correctly identify the instance that is running on the node.

Data Type: String

Range: Not applicable

Default: NULL

Tunable: When disabled

`proxy_probe_timeout`
This property specifies the timeout value, in seconds, that the proxy monitor uses when checking the status of the Oracle Clusterware resource for which this resource is acting as a proxy.

Data Type: Integer

Range: 5–3600

Default: 120

Tunable: Any time

`startup_wait_count`
This property specifies the maximum number of attempts by this resource to confirm that the Oracle Clusterware software is started completely. The interval between attempts is twice the value of the `proxy_probe_timeout` extension property.

The resource requires confirmation that Oracle Clusterware software is started before attempting to start an Oracle RAC database instance. If the maximum number of attempts is exceeded, the resource does not attempt to start the database instance.

Data Type: Integer

Range: 10–600

Default: 20

Tunable: When disabled

`user_env`
This property specifies the name of the file that contains the environment variables that are to be set before the database starts up or shuts down. You must define all environment variables whose values differ from Oracle defaults in this file.

For example, a user's `listener.ora` file might not be located under the `/var/opt/oracle` directory or the *oracle-home*`/network/admin` directory. In this situation, the `TNS_ADMIN` environment variable must be defined.

The definition of each environment variable that is defined must follow the format *variable-name*=*value*. Each definition must start on a new line in the environment file.

You can specify a different value of the `user_env` extension property for each node that can master the resource.

Data Type: String

Range: Not applicable

Default: No default defined

Tunable: Any time

SUNW.ScalDeviceGroup Extension Properties

`debug_level`
This property specifies the level to which debug messages from the resource of this type are logged. When the debug level is increased, more debug messages are written to the log files.

Data type: Integer

Default: 0

Range: 0–10

Tunable: Any time

`diskgroupname`
This property specifies the name of the device group that the resource represents. You must set this property to one of the following items:

- The name of an existing Solaris Volume Manager for Sun Cluster multi-owner disk set. This name was specified in the `metaset(1M)` command with which the disk set was created.
- The name of an existing VxVM shared-disk group. This name was specified in the Veritas command with which the disk group was created.

The requirements for the device group that you specify are as follows:

- The device group must be a valid, existing multi-owner disk set or shared-disk group.
- The device group must be hosted on all nodes that can master the resource.
- The device group must be accessible from all nodes that can master the scalable device group resource.
- The device group must contain at minimum one volume.

Data type: String

Default: No default defined

Range: Not applicable

Tunable: When disabled

`logicaldevicelist`
This property specifies a comma-separated list of logical volumes that the fault monitor of the resource is to monitor. This property is optional. If you do not specify a value for this property, all logical volumes in the device group are to be monitored.

The status of the device group is derived from the statuses of the individual logical volumes that are monitored. If all monitored logical volumes are healthy, the device group is healthy. If any monitored logical volume is faulty, the device group is faulty.

The status of an individual logical volume is obtained by querying the volume's volume manager. If the status of a Solaris Volume Manager for Sun Cluster volume cannot be determined from a query, the fault monitor performs file input/output (I/O) operations to determine the status.

If a device group is discovered to be faulty, monitoring of the resource that represents the group is stopped and the resource is put into the disabled state.

Note – For mirrored disks, if one submirror is faulty, the device group is still considered to be healthy.

The requirements for each logical volume that you specify are as follows:

- The logical volume must exist.
- The logical volume must be contained in the device group that the `diskgroupname` property specifies.
- The logical volume must be accessible from all nodes that can master the scalable device group resource.

Data type: String array

Default: ""

Range: Not applicable

Tunable: Any time

`monitor_retry_count`
This property specifies the maximum number of restarts by the process monitor facility (PMF) that are allowed for the fault monitor.

Data type: Integer

Default: 4

Range: No range defined

Tunable: Any time

`monitor_retry_interval`
This property specifies the period of time in minutes during which the PMF counts restarts of the fault monitor.

Data type: Integer

Default: 2

Range: No range defined

Tunable: Any time

SUNW.ScalMountPoint Extension Properties

`debug_level`
This property specifies the level to which debug messages from the resource for a file-system mount point are logged. When the debug level is increased, more debug messages are written to the log files.

Data type: Integer

Default: 0

Range: 0–10

Tunable: Any time

`filesystemtype`
This property specifies the type of file system whose mount point the resource represents. You must specify this property. Set this property to one of the following values:

`nas`	Specifies that the file system is a file system on a qualified NAS device.
`s-qfs`	Specifies that the file system is a Sun QFS shared file system.

Data type: String

Default: No default defined

Range: Not applicable

Tunable: When disabled

`iotimeout`
This property specifies the timeout value in seconds that the fault monitor uses for file input/output (I/O) probes. To determine if the mounted file system is available, the fault monitor performs I/O operations such as opening, reading, and writing to a test file on the file system. If an I/O operation is not completed within the timeout period, the fault monitor reports an error.

Data type: Integer

Default: 300

Range: 5–300

Tunable: Any time

`monitor_retry_count`
This property specifies the maximum number of restarts by the process monitor facility (PMF) that are allowed for the fault monitor.

Data type: Integer

Default: 4

Range: No range defined

Tunable: Any time

`monitor_retry_interval`
This property specifies the period of time in minutes during which the PMF counts restarts of the fault monitor.

Data type: Integer

Default: 2

Range: No range defined

Tunable: Any time

`mountoptions`
This property specifies a comma-separated list of mount options that are to be used when the file system that the resource represents is mounted. This property is optional. If you do not specify a value for this property, mount options are obtained from the file system's table of defaults.

- For a Sun QFS shared file system, these options are obtained from the `/etc/opt/SUNWsamfs/samfs.cmd` file.
- For a file system on a qualified NAS device, these options are obtained from the `/etc/vfstab` file.

Mount options that you specify through this property override the mount options in the file system's table of defaults.

Data type: String

Default: ""

Range: Not applicable

Tunable: When disabled

`mountpointdir`
This property specifies the mount point of the file system that the resource represents. The mount point is the full path to the directory where the file system is attached to the file system hierarchy when the file system is mounted. You must specify this property.

The directory that you specify must already exist.

Data type: String

Default: No default defined

Range: Not applicable

Tunable: When disabled

`targetfilesystem`
This property specifies the file system that is to be mounted at the mount point that the `mountpointdir` extension property specifies. You must specify this property. The type of the file system must match the type that the `filesystemtype` property specifies. The format of this property depends on the type of the file system as follows:

- For a Sun QFS shared file system, set this property to the name that was assigned to the file system when the file system was created. The file system must be correctly configured. For more information, see your Sun QFS shared file system documentation.
- For a file system on a qualified NAS device, set this property to *nas-device*:*path*. The replaceable items in this format are as follows:

  nas-device
  Specifies the name of the qualified NAS device that is exporting the file system. You can optionally qualify this name with a domain.

  path
  Specifies the full path to the file system that the qualified NAS device is exporting.

  The qualified NAS device and the file system must already be configured for use with Sun Cluster. For more information, see *Oracle Solaris Cluster 3.3 With Network-Attached Storage Devices Manual*.

Data type: String

Default: No default defined

Range: Not applicable

Tunable: When disabled

SPARC: SUNW.vucmm_cvm Extension Properties

`cvm_abort_step_timeout`
This property specifies the timeout (in seconds) for the abort step of a reconfiguration of the Veritas Volume Manager (VxVM) component of the volume manager reconfiguration framework.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_return_step_timeout`
This property specifies the timeout (in seconds) for the return step of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_start_step_timeout`
This property specifies the timeout (in seconds) for the start step of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default:120

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step1_timeout`
This property specifies the timeout (in seconds) for step 1 of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default:100

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step2_timeout`
This property specifies the timeout (in seconds) for step 2 of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default: 100

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step3_timeout`
This property specifies the timeout (in seconds) for step 3 of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default: 240

Range: 30 – 99999 seconds

Tunable: Any time

`cvm_step4_timeout`
This property specifies the timeout (in seconds) for step 4 of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default: 320

Range: 100 – 99999 seconds

Tunable: Any time

`cvm_stop_step_timeout`
This property specifies the timeout (in seconds) for the stop step of a reconfiguration of the VxVM component of the volume manager reconfiguration framework.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

`vxclust_num_ports`
This property specifies the number of communications ports that the `vxclust` program uses.

Data Type: Integer

Default: 32

Range: 16 – 64

Tunable: When disabled

`vxclust_port`
This property specifies the communications port number that the `vxclust` program uses.

Data Type: Integer

Default: 5568

Range: 1024 – 65535

Tunable: When disabled

`vxconfigd_port`
This property specifies the communications port number that the VxVM component configuration daemon `vxconfigd` uses.

Data Type: Integer

Default: 5560

Range: 1024 – 65535

Tunable: When disabled

`vxkmsgd_port`
This property specifies the communications port number that the VxVM component messaging daemon `vxkmsgd` uses.

Data Type: Integer

Default: 5559

Range: 1024 – 65535

Tunable: When disabled

SUNW.vucmm_framework Extension Properties

`reservation_timeout`
This property specifies the timeout (in seconds) for the reservation step of a reconfiguration of the framework.

Data Type: Integer

Default: 325

Range: 100 – 99999 seconds

Tunable: Any time

SUNW.vucmm_svm Extension Properties

`debug_level`
This property specifies the level to which debug messages from the Solaris Volume Manager for Sun Cluster component are logged. When the debug level is increased, more messages are written to the log files during reconfiguration.

Data Type: Integer

Default: 1, which logs `syslog` messages

Range: 0 – 10

Tunable: Any time

`svm_abort_step_timeout`
This property specifies the timeout (in seconds) for the abort step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_return_step_timeout`
This property specifies the timeout (in seconds) for the return step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_start_step_timeout`
This property specifies the timeout (in seconds) for the start step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step1_timeout`
This property specifies the timeout (in seconds) for step 1 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step2_timeout`
This property specifies the timeout (in seconds) for step 2 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step3_timeout`
This property specifies the timeout (in seconds) for step 3 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 30 – 99999 seconds

Tunable: Any time

`svm_step4_timeout`
This property specifies the timeout (in seconds) for step 4 of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 120

Range: 100 – 99999 seconds

Tunable: Any time

`svm_stop_step_timeout`
This property specifies the timeout (in seconds) for the stop step of a reconfiguration of the Solaris Volume Manager for Sun Cluster module of the volume manager reconfiguration framework.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

SUNW.wait_zc_boot Extension Properties

`zcname` This property specifies the name of the zone cluster that needs to be booted before the dependent resource.

Data type: String

Default: No default defined

Range: Not applicable

Tunable: When disabled

◆ ◆ ◆ APPENDIX D

Command-Line Alternatives

Oracle Solaris Cluster maintenance commands enable you to automate the creation, modification, and removal of the framework resource groups by using scripts. Automating this process reduces the time for propagating the same configuration information to many nodes in a cluster.

This appendix contains the following sections:

Setting Support for Oracle RAC Extension Properties

The procedures in the sections that follow contain instructions for registering and configuring resources. These instructions explain how to set *only* extension properties that Support for Oracle RAC requires you to set. Optionally, you can set additional extension properties to override their default values. For more information, see the following sections:

- "Tuning Support for Oracle RAC" on page 136
- Appendix C, "Support for Oracle RAC Extension Properties"

Registering and Configuring the Framework Resource Groups by Using Oracle Solaris Cluster Maintenance Commands

The tasks in this section are an alternative for the resource-configuration steps in "How to Register and Configure the RAC Framework Resource Group by Using `clsetup`" on page 66. Instructions include configuring the multiple-owner volume-manager framework, which currently you cannot configure by using `clsetup`. The following information is in this section:

Overview of the Framework Resource Groups

This section describes the following framework resource groups:

RAC Framework Resource Group

The RAC framework resource group enables Oracle RAC to run with Oracle Solaris Cluster. This resource group contains an instance of the following single-instance resource types:

- `SUNW.rac_framework`, which represents the framework that enables Oracle RAC to be managed by using Oracle Solaris Cluster commands
- SPARC: `SUNW.rac_udlm`, which represents the Oracle UNIX Distributed Lock Manager (Oracle UDLM) (Oracle UDLM) component of Support for Oracle RAC

In addition, the RAC framework resource group can contain an instance of a single-instance resource type that represents the volume manager that you are using for Oracle files, if any. This is the legacy configuration that is used in releases before the Sun Cluster 3.2 11/09 release, before the introduction of the `SUNW.vucmm_framework` resource group to contain multiple-owner volume-manager resources.

Note – This legacy use of the `SUNW.rac_framework` resource group to contain volume-manager resources is still supported in this release, but might be deprecated in a future Oracle Solaris Cluster release.

- Solaris Volume Manager for Sun Cluster is represented by the `SUNW.rac_svm` resource type.
- SPARC: VxVM with the cluster feature is represented by the `SUNW.rac_cvm` resource type.

Note – The resource types that are defined for the RAC framework resource group do *not* enable the Resource Group Manager (RGM) to manage instances of Oracle RAC.

Multiple-Owner Volume-Manager Framework Resource Group

The multiple-owner volume-manager framework resource group enables a multiple-owner shared-storage feature to be used by Oracle RAC.

The multiple-owner volume-manager framework resource group is based on the `SUNW.vucmm_framework` resource type. This resource group contains the volume manager resources for the multiple-owner volume-manager framework, `SUNW.vucmm_svm` or `SUNW.vucmm_cvm`.

When a multiple-owner volume-manager framework resource group is configured, you do not also put a `SUNW.rac_svm` or `SUNW.rac_cvm` resource in the `SUNW.rac_framework` resource group.

▼ How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands

Perform this procedure on only one node of the global cluster.

1 Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.

2 Create a scalable Oracle RAC resource group.

Note – If you are performing the steps in this procedure to register and configure the RAC resource framework in a zone cluster and RAC support is not also required in the global cluster, you do not need to also create a RAC framework resource group in the global cluster. In that case, skip this step and proceed to Step 7.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -n nodelist \
  -p maximum_primaries=num-in-list \
```

```
  -p desired_primaries=num-in-list \
  [-p rg_description="description"] \
  -p rg_mode=Scalable rac-fmwk-rg
```

`-n nodelist=`*nodelist*
: Specifies a comma-separated list of cluster nodes on which Support for Oracle RAC is to be enabled. The Support for Oracle RAC software packages must be installed on each node in this list.

`-p maximum_primaries=`*num-in-list*
: Specifies the number of nodes on which Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.

`-p desired_primaries=`*num-in-list*
: Specifies the number of nodes on which Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.

`-p rg_description="`*description*"
: Specifies an optional brief description of the resource group. This description is displayed when you use Oracle Solaris Cluster maintenance commands to obtain information about the resource group.

`-p rg_mode=Scalable`
: Specifies that the resource group is scalable.

rac-fmwk-rg
: Specifies the name that you are assigning to the Oracle RAC resource group.

3 Register the `SUNW.rac_framework` resource type.

```
# clresourcetype register SUNW.rac_framework
```

4 Add an instance of the `SUNW.rac_framework` resource type to the resource group that you created in Step 2.

```
# clresource create -g rac-fmwk-rg -t SUNW.rac_framework rac-fmwk-rs
```

`-g` *rac-fmwk-rg* — Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 2.

rac-fmwk-rs — Specifies the name that you are assigning to the `SUNW.rac_framework` resource.

5 SPARC: Register the `SUNW.rac_udlm` resource type.

```
# clresourcetype register SUNW.rac_udlm
```

6 **SPARC: Add an instance of the `SUNW.rac_udlm` resource type to the resource group that you created in Step 2.**

Ensure that this instance depends on the `SUNW.rac_framework` resource that you created in Step 4.

```
# clresource create -g resource-group \
 -t SUNW.rac_udlm \
 -p resource_dependencies=rac-fmwk-rs  rac-udlm-rs
```

`-g` *rac-fmwk-rg*
Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 2.

`-p resource_dependencies=`*rac-fmwk-rs*
Specifies that this instance depends on the `SUNW.rac_framework` resource that you created in Step 4.

rac-udlm-rs
Specifies the name that you are assigning to the `SUNW.rac_udlm` resource.

7 **Create a scalable multiple-owner volume-manager framework resource group, if one does not already exist.**

If you do not intend to create a multiple-owner volume-manager framework resource group, skip to Step 10.

```
# clresourcegroup create -n nodelist -S vucmm-fmwk-rg
```

`-n nodelist=`*nodelist*
Specifies the same node list that you configured for the scalable Oracle RAC resource group.

vucmm-fmwk-rg
Specifies the name that you are assigning to the multiple-owner volume-manager framework resource group.

8 **Register the `SUNW.vucmm_framework` resource type.**

If you did not create a multiple-owner volume-manager framework resource group, skip to Step 10.

```
# clresourcetype register SUNW.vucmm_framework
```

9 **Add an instance of the `SUNW.vucmm_framework` resource type to the resource group that you created in Step 7.**

If you did not create a multiple-owner volume-manager framework resource group, skip to Step 10.

```
# clresource create -g vucmm-fmwk-rg -t SUNW.vucmm_framework vucmm-fmwk-rs
```

`-g` *vucmm-fmwk-rg* Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 7.

vucmm-fmwk-rs Specifies the name that you are assigning to the `SUNW.vucmm_framework` resource.

10 Register and add an instance of the resource type that represents the volume manager that you are using for Oracle files, if any.

If you are not using a volume manager, omit this step.

- **If you are using Solaris Volume Manager for Sun Cluster, register and add the instance as follows:**

  a. Register the resource type.

  ```
  # clresourcetype register svm-rt
  ```

  svm-rt
  Specifies the resource type you register.
  - If you are using a `SUNW.vucmm_framework` based resource group, register the `SUNW.vucmm_svm` resource type.
  - If you are using only a `SUNW.rac_framework` based resource group, register the `SUNW.rac_svm` resource type.

  b. Add an instance of the resource type to the resource group to contain the volume manager resource.

  Ensure that this instance depends on the framework resource that you created.

  ```
  # clresource create -g fmwk-rg \
    -t svm-rt \
    -p resource_dependencies=fmwk-rs svm-rs
  ```

  `-g` *fmwk-rg*
  Specifies the resource group to which you are adding the resource.
  - If you are using a `SUNW.vucmm_framework` based resource group, specify the resource group that you created in Step 7.
  - If you are using only a `SUNW.rac_framework` based resource group, specify the resource group that you created in Step 2.

  `-p resource_dependencies=`*fmwk-rs*
  Specifies that this instance depends on the framework resource that you created.
  - If you are using a `SUNW.vucmm_framework` based resource group, specify the resource group that you created in Step 9.
  - If you are using only a `SUNW.rac_framework` based resource group, specify the resource group that you created in Step 4.

  svm-rs
  Specifies the name that you are assigning to the `SUNW.vucmm_svm` or `SUNW.rac_svm` resource.

- **SPARC: If you are using VxVM with the cluster feature, register and add the instance as follows.**

  a. **Register the resource type.**

  ```
  # clresourcetype register cvm-rt
  ```

  cvm-rt
  Specifies the resource type you register.
  - If you are using a `SUNW.vucmm_framework` based resource group, register the `SUNW.vucmm_cvm` resource type.
  - If you are using only a `SUNW.rac_framework` based resource group, register the `SUNW.rac_cvm` resource type.

  b. **Add an instance of the resource type to the resource group that you created.**

  Ensure that this instance depends on the framework resource that you created.

  ```
  # clresource create -g fmwk-rg \
    -t cvm-rt \
    -p resource_dependencies=fmwk-rs  cvm-rs
  ```

  `-g` *fmwk-rg*
  Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created.
  - If you are using a `SUNW.vucmm_framework` based resource group, specify the resource group that you created in Step 7.
  - If you are using only a `SUNW.rac_framework` based resource group, specify the resource group that you created in Step 2.

  `-p resource_dependencies=`*fmwk-rs*
  Specifies that this instance depends on the framework resource that you created.
  - If you are using a `SUNW.vucmm_framework` based resource group, specify the resource group that you created in Step 9.
  - If you are using only a `SUNW.rac_framework` based resource group, specify the resource group that you created in Step 4.

  cvm-rs
  Specifies the name that you are assigning to the `SUNW.vucmm_cvm` or `SUNW.rac_cvm` resource.

11 Bring online and in a managed state the RAC framework resource group, the multiple-owner volume-manager framework resource group if used, and their resources.

```
# clresourcegroup online -emM rac-fmwk-rg [vucmm-fmwk-rg]
```

rac-fmwk-rg Specifies that the RAC resource group that you created in Step 2 is to be moved to the `MANAGED` state and brought online.

vucmm-fmwk-rg Specifies that the multiple-owner volume-manager framework resource group that you created in Step 7 is to be moved to the `MANAGED` state and brought online.

How to Register and Configure the RAC Framework Resource Group in a Zone Cluster by Using Oracle Solaris Cluster Maintenance Commands

Perform the steps in this procedure to register and configure the RAC framework resource group in a zone cluster for the Sun QFS shared file system with Solaris Volume Manager. For this configuration, you need to create the RAC framework resource group in both the global cluster and zone cluster.

Note – When a step in the procedure requires running the Oracle Solaris Cluster commands in a zone cluster, you should run the command from the global cluster and use the `-Z` option to specify the zone cluster.

Before You Begin Perform the steps to register and configure the RAC framework resource group *rac-fmwk-rg*, with resources *rac-fmwk-rs* and *rac-svm-rs* in the global cluster.

Note – For information on registering and configuring the RAC framework resource group in the global cluster, see "How to Register and Configure the Framework Resource Groups in the Global Cluster by Using Oracle Solaris Cluster Maintenance Commands" on page 303.

---

1. **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2. **Create a scalable Oracle RAC resource group.**

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -Z zcname -n nodelist \
 -p maximum_primaries=num-in-list \
 -p desired_primaries=num-in-list \
 [-p rg_description="description"] \
 -p rg_mode=Scalable rac-fmwk-rg
```

3 **Register the `SUNW.rac_framework` resource type.**

```
# clresourcetype register -Z zcname SUNW.rac_framework
```

4 **Add an instance of the `SUNW.rac_framework` resource type to the resource group that you created in Step 2.**

```
# clresource create -Z zcname -g rac-fmwk-rg \
-t SUNW.rac_framework rac-fmwk-rs
```

`-g` *rac-fmwk-rg* — Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 2.

rac-fmwk-rs — Specifies the name that you are assigning to the `SUNW.rac_framework` resource.

5 **SPARC: Register the `SUNW.rac_udlm` resource type.**

```
# clresourcetype register -Z zcname SUNW.rac_udlm
```

6 **SPARC: Add an instance of the `SUNW.rac_udlm` resource type to the resource group that you created in Step 2.**

Ensure that this instance depends on the `SUNW.rac_framework` resource that you created in Step 4.

```
# clresource create -Z zcname -g resource-group \
 -t SUNW.rac_udlm \
 -p resource_dependencies=rac-fmwk-rs  rac-udlm-rs
```

`-g` *rac-fmwk-rg*
Specifies the Oracle RAC resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 2.

`-p resource_dependencies=`*rac-fmwk-rs*
Specifies that this instance depends on the `SUNW.rac_framework` resource that you created in Step 4.

rac-udlm-rs
Specifies the name that you are assigning to the `SUNW.rac_udlm` resource.

7 **Bring online and in a managed state the RAC framework resource group and its resources.**

```
# clresourcegroup online -Z zcname -emM rac-fmwk-rg
```

Registering and Configuring Oracle ASM Resource Groups (CLI)

The following information is in this section:

- "How to Register and Configure Oracle ASM Resource Groups in a Zone Cluster (CLI)" on page 313

▼ How to Register and Configure Oracle ASM Resource Groups in the Global Cluster (CLI)

Before You Begin

- Ensure that the Oracle Clusterware framework resource is created and that dependencies are configured between the RAC framework resource and the Oracle Clusterware framework resource.
- Ensure that the RAC framework resource group, the multiple-owner volume-manager framework resource group if used, and their resources are online.

1 Register the ASM resource types for the data service.

a. Register the scalable ASM instance proxy resource type.

```
# clresourcetype register SUNW.scalable_asm_instance_proxy
```

b. Register the ASM disk-group resource type.

- **For For Oracle 10*g* and 11*g* release 1 only, use the `SUNW.asm_diskgroup` resource type.**

  ```
  # clresourcetype register SUNW.asm_diskgroup
  ```

- **For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type.**

  ```
  # clresourcetype register SUNW.scalable_asm_diskgroup_proxy
  ```

2 Create resource groups *asm-inst-rg* and *asm-dg-rg*.

```
# clresourcegroup create -S asm-inst-rg asm-dg-rg
```

asm-inst-rg
Specifies the name of the ASM instance resource group.

asm-dg-rg
Specifies the name of the ASM disk-group resource group.

3 Set a strong positive affinity on *rac-fmwk-rg* by *asm-inst-rg*.

```
# clresourcegroup set -p rg_affinities=++rac-fmwk-rg asm-inst-rg
```

4 Set a strong positive affinity on *asm-inst-rg* by *asm-dg-rg*.

```
# clresourcegroup set -p rg_affinities=++asm-inst-rg asm-dg-rg
```

5 If the Oracle Oracle ASM home is located on a cluster file system or a Sun QFS shared file system, create a storage resource and bring it online.

- **For a cluster file system, create a `SUNW.HAStoragePlus` resource.**

```
# clresourcetype register SUNW.HAStoragePlus

# clresource create -g asm-inst-rg \
-t SUNW.HAStoragePlus \
-p FileSystemMountpoints=clusterfilesystem \
asm-stor-rs
# clresourcegroup online -eM asm-inst-rg
```

`-p FileSystemMountpoints=`*clusterfilesystem*
Sets the cluster file system.

asm-stor-rs
Specifies the name of the Oracle ASM storage resource that you are creating.

- **For a Sun QFS shared file system, create a `SUNW.qfs` resource.**

```
# clresourcetype register SUNW.qfs
# clresourcegroup create qfs-rg

# clresource create -g qfs-rg \
-t SUNW.qfs \
-p QFSFileSystem=qfs-mp \
qfs-rs

# clresourcegroup online -eM qfs-rg
```

qfs-rg
Specifies the name of the QFS resource group.

`-p QFSFileSystem=`*qfs-mp*
Sets the mount point of the QFS shared file system that is used for the Oracle Oracle ASM home.

qfs-rs
Specifies the name of the QFS resource.

6 If you use a Sun QFS shared file system for the Oracle Oracle ASM home, create a scalable mount-point resource, set the resource dependency, and bring the resource group online.

```
# clresourcetype register SUNW.ScalMountPoint
# clresourcegroup create -S scal-mp-rg

# clresource create -g scal-mp-rg \
-t SUNW.ScalMountPoint \
-p mountpointdir=qfs-mp \
-p filesystemtype=s-qfs \
-p targetfilesystem=qfs-fs \
-p resource_dependencies_restart=qfs-rs \
qfs-mp-rs

# clresourcegroup online -eM scal-mp-rg
```

`-p mountpointdir=`*qfs-mp*
Sets the QFS mount point.

`-p targetfilesystem=`*qfs-fs*
Sets the QFS shared file system.

scal-mp-rg
Specifies the name of the scalable mount-point resource group.

7 Create a `SUNW.scalable_asm_instance_proxy` resource and set the resource dependencies.

- **For a cluster file system, perform the following commands:**

```
# clresource create -g asm-inst-rg \
-t SUNW.scalable_asm_instance_proxy \
-p ORACLE_HOME=oracle-asm-home \
-p CRS_HOME=crs_home \
-p "ORACLE_SID{node1}"=asm-instance1 \
-p "ORACLE_SID{node2}"=asm-instance2 \
-p resource_dependencies_restart=asm-stor-rs \
-p resource_dependencies_offline_restart=crs-fmwk-rs \
-d asm-inst-rs
```

`-t SUNW.scalable_asm_instance_proxy`
Specifies the type of the resource to add.

`-p ORACLE_HOME=`*oracle-asm-home*
Sets the path to the Oracle Oracle ASM home directory.

`-p CRS_HOME=`*crs-home*
Sets the path to the Oracle Clusterware home directory.

`-p ORACLE_SID=`*asm-instance*
Sets the Oracle Oracle ASM system identifier.

`-d` *asm-inst-rs*
Specifies the name of the Oracle ASM instance resource that you create.

- **For a QFS shared file system, perform the following commands:**

```
# clresource create -g asm-inst-rg \
-t SUNW.scalable_asm_instance_proxy \
-p ORACLE_HOME=oracle-asm-home \
-p CRS_HOME=crs_home \
-p "ORACLE_SID{node1}"=asm-instance1 \
-p "ORACLE_SID{node2}"=asm-instance2 \
-p resource_dependencies_offline_restart=crs-fmwk-rs,qfs-mp-rs \
-d asm-inst-rs
```

qfs-mp-rs
Specifies the name of the scalable mount-point resource.

- **For a local file system, perform the following commands:**

```
# clresource create -g asm-inst-rg \
-t SUNW.scalable_asm_instance_proxy \
-p ORACLE_HOME=oracle-asm-home \
-p CRS_HOME=crs_home \
-p "ORACLE_SID{node1}"=asm-instance1 \
-p "ORACLE_SID{node2}"=asm-instance2 \
-p resource_dependencies_offline_restart=crs-fmwk-rs \
-d asm-inst-rs
```

8 **Add an ASM disk-group resource type to the *asm-dg-rg* resource group.**

- **For Oracle 10*g* and 11*g* release 1 only, use the `SUNW.asm_diskgroup` resource type.**

```
# clresource create -g asm-dg-rg -t SUNW.asm_diskgroup \
-p asm_diskgroups=dg[,dg...] \
-p resource_dependencies_offline_restart=asm-inst-rs,asm-stor-rs \
-d asm-dg-rs
```

`-t SUNW.asm_diskgroup`
Specifies the type of the resource to add.

`-p asm_diskgroups=`*dg*
Specifies the ASM disk group.

`-d` *asm-dg-rs*
Specifies the name of the resource that you are creating.

- **For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type.**

```
# clresource create -g asm-dg-rg -t SUNW.scalable_asm_diskgroup_proxy \
-p asm_diskgroups=dg[,dg...] \
-p resource_dependencies_offline_restart=asm-inst-rs,asm-stor-rs \
-d asm-dg-rs
```

9 **Bring online the *asm-inst-rg* resource group in a managed state on a cluster node.**

```
# clresourcegroup online -eM asm-inst-rg
```

10 **Bring online the *asm-dg-rg* resource group in a managed state on a cluster node.**

```
# clresourcegroup online -eM asm-dg-rg
```

11 **Verify the Oracle ASM configuration.**

```
# clresource status +
```

▼ How to Register and Configure Oracle ASM Resource Groups in a Zone Cluster (CLI)

Before You Begin

- Ensure that the Oracle Clusterware framework resource is created and that dependencies are configured between the RAC framework resource and the Oracle Clusterware framework resource.

- Ensure that the RAC framework resource group, the multiple-owner volume-manager framework resource group if used, and their resources are online.

Perform all steps from the global zone.

1 Register the ASM resource types for the data service.

a. Register the scalable ASM instance proxy resource type.

```
# clresourcetype register -Z zcname SUNW.scalable_asm_instance_proxy
```

b. Register the ASM disk-group resource type.

- **For For Oracle 10*g* and 11*g* release 1 only, use the `SUNW.asm_diskgroup` resource type.**

  ```
  # clresourcetype register -Z zcname SUNW.asm_diskgroup
  ```

- **For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type.**

  ```
  # clresourcetype register -Z zcname SUNW.scalable_asm_diskgroup_proxy
  ```

2 Create resource groups *asm-inst-rg* and *asm-dg-rg*.

```
# clresourcegroup create -Z zcname -S asm-inst-rg asm-dg-rg
```

asm-inst-rg
Specifies the name of the Oracle ASM instance resource group.

asm-dg-rg
Specifies the name of the Oracle ASM disk-group resource group.

3 Set a strong positive affinity on *rac-fmwk-rg* by *asm-inst-rg*.

```
# clresourcegroup set -Z zcname -p rg_affinities=++rac-fmwk-rg asm-inst-rg
```

4 Set a strong positive affinity on *asm-inst-rg* by *asm-dg-rg*.

```
# clresourcegroup set -Z zcname -p rg_affinities=++asm-inst-rg asm-dg-rg
```

5 If the Oracle Oracle ASM home is located on a Sun QFS shared file system, create a storage resource.

a. Ensure that the file system is available for use from within the zone cluster.

```
# clzonecluster configure zcname
clzc:zcname> add fs
clzc:zcname:fs> set dir=mountpoint
clzc:zcname:fs> set special=QFSfilesystemname
clzc:zcname:fs> set type=samfs
clzc:zcname:fs> end
clzc:zcname> verify
clzc:zcname> commit
clzc:zcname> exit
```

b. Create a storage resource and bring it online.

Create a `SUNW.qfs` resource with a dependency on a `SUNW.wait_zc_boot` resource.

```
# clresourcetype register SUNW.wait_zc_boot
# clresourcetype register SUNW.qfs

# clresourcegroup create -S scal-wait-zc-rg
# clresourcegroup create qfs-rg

# clresource create -g scal-wait-zc-rg \
-t SUNW.wait_zc_boot \
-p zcname=zcname \
wait-zc-rs

# clresource create -g qfs-rg \
-t SUNW.qfs \
-p QFSFileSystem=qfs-mp \
-p resource_dependencies=wait-zc-rs
qfs-rs

# clresourcegroup online -eM scal-wait-zc-rg
# clresourcegroup online -eM qfs-rg
```

scal-wait-zc-rg
Specifies the name of the `SUNW.wait_zc_boot` resource group.

qfs-rg
Specifies the name of the QFS resource group.

wait-zc-rs
Specifies the name of the `SUNW.wait_zc_boot` resource.

`-p QFSFileSystem=`*qfs-mp*
Sets the mount point of the QFS shared file system that is used for the Oracle Oracle ASM home.

c. Create a scalable mount-point resource, set the resource dependency, and bring the resource group online.

```
# clresourcetype register -Z zcname SUNW.ScalMountPoint
# clresourcegroup create -Z zcname -S scal-mp-rg

# clresource create -Z zcname -g scal-mp-rg \
-t SUNW.ScalMountPoint \
-p mountpointdir=qfs-mp \
-p filesystemtype=s-qfs \
-p targetfilesystem=qfs-fs \
-p resource_dependencies_restart=global:qfs-rs \
qfs-mp-rs

# clresourcegroup online -Z zcname -eM scal-mp-rg
```

scal-mp-rg
Specifies the name of the scalable mount-point resource group.

`-p targetfilesystem=`*qfs-fs*
Specifies the name of the QFS shared file system.

qfs-rs
Specifies the name of the QFS resource.

qfs-mp-rs
Specifies the name of the scalable mount-point resource.

6 Create a `SUNW.scalable_asm_instance_proxy` resource and set the resource dependencies.

- **For a QFS shared file system, perform the following command:**

```
# clresource create -Z zcname -g asm-inst-rg \
-t SUNW.scalable_asm_instance_proxy \
-p ORACLE_HOME=oracle-asm-home \
-p CRS_HOME=crs_home \
-p "ORACLE_SID{node1}"=asm-instance1 \
-p "ORACLE_SID{node2}"=asm-instance2 \
-p resource_dependencies_offline_restart=crs-fmwk-rs,qfs-mp-rs \
-d asm-inst-rs
```

- **For a local file system, perform the following command:**

```
# clresource create -Z zcname -g asm-inst-rg \
-t SUNW.scalable_asm_instance_proxy \
-p ORACLE_HOME=oracle-asm-home \
-p CRS_HOME=crs_home \
-p "ORACLE_SID{node1}"=asm-instance1 \
-p "ORACLE_SID{node2}"=asm-instance2 \
-p resource_dependencies_offline_restart=crs-fmwk-rs \
-d asm-inst-rs
```

`-g` *asm-inst-rg*
Specifies the name of the resource group into which the resources are to be placed.

`-t SUNW.scalable_asm_instance_proxy`
Specifies the type of the resource to add.

`-p ORACLE_HOME=`*Oracle-asm-home*
Sets the path to the Oracle Oracle ASM home directory.

`-p ORACLE_SID=`*asm-instance*
Sets the Oracle Oracle ASM system identifier.

`-d` *asm-inst-rs*
Specifies the name of the resource that you are creating.

7 Add an ASM disk-group resource type to the *asm-dg-rg* resource group.

- **For Oracle 10*g* and 11*g* release 1 only, use the `SUNW.asm_diskgroup` resource type.**

```
# clresource create -Z zcname -g asm-dg-rg -t SUNW.asm_diskgroup \
-p asm_diskgroups=dg[,dg...] \
-p resource_dependencies_offline_restart=asm-inst-rs,asm-stor-rs \
-d asm-dg-rs
```

`-t SUNW.asm_diskgroup`
Specifies the type of the resource to add.

`-p asm_diskgroups=`*dg*
Specifies the ASM disk group.

`-d` *asm-dg-rs*
Specifies the name of the resource that you are creating.

- **For Oracle 11*g* release 2 only, use the `SUNW.scalable_asm_diskgroup_proxy` resource type.**

```
# clresource create -Z zcname -g asm-dg-rg -t SUNW.scalable_asm_diskgroup_proxy \
-p asm_diskgroups=dg[,dg...] \
-p resource_dependencies_offline_restart=asm-inst-rs,asm-stor-rs \
-d asm-dg-rs
```

8 **Bring online the *asm-inst-rg* resource group in a managed state on a cluster node.**

```
# clresourcegroup online -Z zcname -eM asm-inst-rg
```

9 **Bring online the *asm-dg-rg* resource group in a managed state on a cluster node.**

```
# clresourcegroup online -Z zcname -eM asm-dg-rg
```

10 **Verify the Oracle ASM configuration.**

```
# clresource status -Z zcname +
```

Creating Storage Management Resources by Using Oracle Solaris Cluster Maintenance Commands

The tasks in this section are alternatives for the resource-configuration steps in "How to Register and Configure Storage Resources for Oracle Files by Using `clsetup`" on page 84. The following information is in this section:

The following resources to represent storage for Oracle files are required:

- Resources for scalable device groups and scalable file-system mount points
- Resources for the Sun QFS metadata server

Resources for Scalable Device Groups and Scalable File-System Mount Points

If you are using Solaris Volume Manager for Sun Cluster or VxVM, configure storage resources as follows:

- Create one scalable resource group to contain all resources for scalable device groups.
- Create one resource for each Solaris Volume Manager for Sun Cluster multi-owner disk set or VxVM shared-disk group that you are using for Oracle files.

If you are using Sun QFS or qualified NAS devices, configure storage resources as follows:

- Create one scalable resource group to contain all resources for scalable file-system mount points.
- Create one resource for each Sun QFS shared file system or NFS file system on a qualified NAS device that you are using for Oracle files.

The resource that represents a Sun QFS shared file system can start only if the file system's Sun QFS metadata server is running. Similarly, the resource that represents a Sun QFS shared file system can stop only if the file system's Sun QFS metadata server is stopped. To meet this requirement, configure a resource for each Sun QFS metadata server. For more information, see "Resources for the Sun QFS Metadata Server" on page 318.

Resources for the Sun QFS Metadata Server

If you are using the Sun QFS shared file system, create one resource for each Sun QFS metadata server. The configuration of resource groups for these resources depends on the version of Oracle that you are using:

- "Configuration of Sun QFS Resource Groups With Oracle 9*i* and Oracle 10*g* Release 2 or 11*g*" on page 318
- "Configuration of Sun QFS Resource Groups With Oracle 10*g* Release 1" on page 319

Configuration of Sun QFS Resource Groups With Oracle 9*i* and Oracle 10*g* Release 2 or 11*g*

If you are using Oracle 9*i*, Oracle 10*g* release 2, or Oracle 11*g*, the configuration of resource groups depends on the number of file systems in your configuration.

- If your configuration contains a small number of file systems, create one resource group for all resources for the Sun QFS metadata server.
- If your configuration contains a large number of file systems, configure resources for the Sun QFSmetadata server in several resource groups as follows:
  - Distribute resources among the resource groups to ensure optimum load balancing.
  - Select a different primary node for each resource group to prevent all the resource groups from being online on the same node simultaneously.

Configuration of Sun QFS Resource Groups With Oracle 10*g* Release 1

If you are using Oracle 10*g*, Oracle Clusterware manages Oracle RAC database instances. These database instances must be started *only* after all shared file systems are mounted.

You might use multiple file systems for database files and related files. For more information, see "Sun QFS File Systems for Database Files and Related Files" on page 56. In this situation, ensure that the file system that contains the Oracle Clusterware voting disk is mounted *only* after the file systems for other database files have been mounted. This behavior ensures that, when a node is booted, Oracle Clusterware resources are started only after all Sun QFS file systems are mounted.

If you are using Oracle 10*g* release 1, the configuration of resource groups must ensure that Oracle Solaris Cluster mounts the file systems in the required order. To meet this requirement, configure resource groups for the metadata servers of the file systems as follows:

- Create the resources for the metadata servers in separate resource groups.
- Set the resource group for the file system that contains the Oracle Clusterware voting disk to depend on the other metadata resource groups.

▼ How to Create a Resource for a Scalable Device Group in the Global Cluster

Perform this procedure on only one node of the cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **Create a scalable resource group to contain the scalable device group resource.**

Set a strong positive affinity by the resource group for the RAC framework resource group.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -p nodelist=nodelist \
-p desired_primaries=num-in-list \
-p maximum_primaries=num-in-list \
-p rg_affinities=++rac-fmwk-rg \
[-p rg_description="description"] \
-p rg_mode=Scalable \
scal-dg-rg
```

3 **Register the `SUNW.ScalDeviceGroup` resource type.**

```
# clresourcetype register SUNW.ScalDeviceGroup
```

4 **For each scalable device group that you are using for Oracle files, add an instance of the `SUNW.ScalDeviceGroup` resource type to the resource group that you created in Step 2.**

Set a strong dependency for the instance of `SUNW.ScalDeviceGroup` on the resource in the RAC framework resource group that represents the volume manager for the device group. Limit the scope of this dependency to only the node where the `SUNW.ScalDeviceGroup` resource is running.

```
# clresource create -t SUNW.ScalDeviceGroup -g scal-dg-rg \
-p resource_dependencies=fm-vol-mgr-rs{local_node} \
-p diskgroupname=disk-group  scal-dg-rs
```

5 **Bring online and in a managed state the resource group that you created in Step 2.**

```
# clresourcegroup online -emM scal-dg-rg
```

▼ How to Create a Resource for a Scalable Device Group in a Zone Cluster

Perform this procedure from the global cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **Create a scalable resource group to contain the scalable device group resource.**

Set a strong positive affinity by the resource group for the RAC framework resource group.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -Z zcname -p nodelist=nodelist \
-p desired_primaries=num-in-list \
-p maximum_primaries=num-in-list \
-p rg_affinities=++rac-fmwk-rg \
[-p rg_description="description"] \
-p rg_mode=Scalable \
scal-dg-rg
```

3 **Register the `SUNW.ScalDeviceGroup` resource type.**

```
# clresourcetype register -Z zcname SUNW.ScalDeviceGroup
```

4 **For each scalable device group that you are using for Oracle files, add an instance of the `SUNW.ScalDeviceGroup` resource type to the resource group that you created in Step 2.**

Set a strong dependency for the instance of `SUNW.ScalDeviceGroup` on the resource in the RAC framework resource group that represents the volume manager for the device group. Limit the scope of this dependency to only the node where the `SUNW.ScalDeviceGroup` resource is running.

```
# clresource create -Z zcname -t SUNW.ScalDeviceGroup -g scal-dg-rg \
-p resource_dependencies=fm-vol-mgr-rs{local_node} \
-p diskgroupname=disk-group  scal-dg-rs
```

5 **Bring online and in a managed state the resource group that you created in Step 2.**

```
# clresourcegroup online -Z zcname-emM scal-dg-rg
```

▼ How to Register and Configure Resources for the Sun QFS Metadata Server in the Global Cluster

Perform this task *only* if you are using the Sun QFS shared file system.

Perform this procedure on only one node of the cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **Create a failover resource group to contain the resources for the Sun QFS metadata server.**

If you are also using a volume manager, set a strong positive affinity by the resource group for the resource group that contains the volume manager's scalable device-group resource. This resource group is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

```
# clresourcegroup create -n nodelist \
[-p rg_affinities=++scal-dg-rg] \
[-p rg_description="description"] \
qfs-mds-rg
```

3 **Register the `SUNW.qfs` resource type.**

```
# clresourcetype register SUNW.qfs
```

4 **For each Sun QFS shared file system that you are using, add an instance of the `SUNW.qfs` resource type to the resource group that you created in Step 2.**

Each instance of `SUNW.qfs` represents the metadata server of the file system.

If you are also using a volume manager, set a strong dependency by the instance of `SUNW.qfs` on the resource for the scalable device group that is to store the file system . This resource is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

```
# clresource create -t SUNW.qfs -g qfs-mds-rg \
-p qfsfilesystem=path \
[-p resource_dependencies=scal-dg-rs] \
qfs-mds-rs
```

5 **Bring online and in a managed state the resource group that you created in Step 2.**

```
# clresourcegroup online -emM qfs-mds-rg
```

▼ How to Register and Configure Resources for the Sun QFS Metadata Server for a Zone Cluster

Perform the steps in this procedure to register and configure resources for the Sun QFS metadata server for a zone cluster.

Perform these steps in the global cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **Create a scalable resource group to contain the `SUNW.wait_zc_boot` resource in the global cluster.**

```
# clresourcegroup create -n nodelist \
-p rg_mode=Scalable \
-p maximum_primaries=num-in-list \
-p desired_primaries=num-in-list \
```

```
[-p rg_mode=Scalable \
zc-wait-rg
```

3 **Register the `SUNW.wait_zc_boot` resource type.**

```
# clresourcetype register SUNW.wait_zc_boot
```

4 **Add an instance of the `SUNW.wait_zc_boot` resource type to the resource group that you created in Step 2.**

```
# clresource create -g zc-wait-rg -t SUNW.wait_zc_boot \
-p ZCName=zcname  zc-wait-rs
```

5 **Bring online and in a managed state the resource group that you created in Step 2.**

```
# clresourcegroup online -emM zc-wait-rg
```

6 **Create a failover resource group to contain the resources for the Sun QFS metadata server.**

Set a strong positive affinity by the resource group for the resource group that contains the `SUNW.wait_zc_boot` resource that is configured for the zone cluster.

If you also use a volume manager, set a strong positive affinity by the resource group for the resource group that contains the volume manager's scalable device-group resource. This resource group is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

```
# clresourcegroup create -n nodelist \
-p rg_affinities=++wait-zc-rg[,++scal-dg-rg] \
[-p rg_description="description"] \
qfs-mds-rg
```

7 **Register the `SUNW.qfs` resource type.**

```
# clresourcetype register SUNW.qfs
```

8 **Add an instance of the `SUNW.qfs` resource type to the resource group that you created in Step 6 for each Sun QFS shared file system that you are using.**

Each instance of `SUNW.qfs` represents the metadata server of the file system.

Set a strong dependency by the instance of `SUNW.qfs` on the `SUNW.wait_zc_boot` resource that is configured for the zone cluster.

If you are also using a volume manager, set a strong dependency by the instance of `SUNW.qfs` on the resource for the scalable device group that is to store the file system. This resource is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

```
# clresource create -t SUNW.qfs -g qfs-mds-rg \
-p qfsfilesystem=path
\-p resource_dependencies=zc-wait-rs[,scal-dg-rs] \
qfs-mds-rs
```

9 **Bring online and in a managed state the resource group that you created in Step 6.**

```
# clresourcegroup online -emM qfs-mds-rg
```

▼ How to Create a Resource for a File-System Mount Point in the Global Cluster

Perform this procedure on only one node of the cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **Create a scalable resource group to contain the resource for a scalable file-system mount point.**

If you are also using a volume manager, set a strong positive affinity by the resource group for the resource group that contains the volume manager's scalable device-group resource. This resource group is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -n nodelist \
-p desired_primaries=num-in-list \
-p maximum_primaries=num-in-list \
[-p rg_affinities=++scal-dg-rg] \
[-p rg_description="description"] \
-p rg_mode=Scalable scal-mp-rg
```

3 **Register the `SUNW.ScalMountPoint` resource type.**

```
# clresourcetype register SUNW.ScalMountPoint
```

4 **For each shared file system that requires a scalable file-system mount point resource, add an instance of the `SUNW.ScalMountPoint` resource type to the resource group that you created in Step 2.**

- **For each Sun QFS shared file system, type the following command:**

  Set a strong dependency by the instance of `SUNW.ScalMountPoint` on the resource for the Sun QFS metadata server for the file system. The resource for the Sun QFS metadata server set is created in "How to Register and Configure Resources for the Sun QFS Metadata Server in the Global Cluster" on page 321.

  If you are also using a volume manager, set an offline-restart dependency by the instance of `SUNW.ScalMountPoint` on the resource for the scalable device group that is to store the file system. This resource is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

  ```
  # clresource create -t SUNW.ScalMountPoint -g scal-mp-rg \
  -p resource_dependencies=qfs-mds-rs \
  [-p resource_dependencies_offline_restart=scal-dg-rs] \
  ```

```
-p mountpointdir=mp-path \
-p filesystemtype=s-qfs \
-p targetfilesystem=fs-name qfs-mp-rs
```

- **For each file system on a qualified NAS device, type the following command:**

  If you are also using a volume manager, set an offline-restart dependency by the instance of `SUNW.ScalMountPoint` on the resource for the scalable device group that is to store the file system. This resource is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

  ```
  # clresource create -t SUNW.ScalMountPoint -g scal-mp-rg \
  [-p resource_dependencies_offline_restart=scal-dg-rs] \
  -p mountpointdir=mp-path \
  -p filesystemtype=nas \
  -p targetfilesystem=nas-device:fs-name nas-mp-rs
  ```

5 **Bring online and in a managed state the resource group that you created in Step 2.**

```
# clresourcegroup online -emM scal-mp-rg
```

How to Create a Resource for a File-System Mount Point in Zone Cluster

Perform the steps in this procedure to create a resource for a file-system mount point in a zone cluster. For RAC configurations with the Sun QFS shared file system on Solaris Volume Manager for Sun Cluster and the Sun QFS shared file system on hardware RAID, you create a scalable resource group to contain all the scalable mount point resources in a zone cluster.

Note – The node list is the list of zone-cluster nodes.

Perform this procedure from the global cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 **Create a scalable resource group to contain the resource for a scalable file-system mount point in zone cluster.**

If you are also using a volume manager, set a strong positive affinity by the resource group for the resource group that contains the volume manager's scalable device-group resource. This resource group is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -Z zcname zcnodelist \
-p desired_primaries=num-in-list \
-p maximum_primaries=num-in-list \
[-p rg_affinities=++global:scal-dg-rg] \
[-p rg_description="description"] \
-p rg_mode=Scalable scal-mp-rg
```

3 Register the `SUNW.ScalMountPoint` resource type.

```
# clresourcetype register -Z zcname SUNW.ScalMountPoint
```

4 For each Sun QFS shared file system that requires a scalable file-system mount point resource, add an instance of the `SUNW.ScalMountPoint` resource type to the resource group that you created in Step 2.

```
# clresource create -Z zcname -t SUNW.ScalMountPoint -d -g scal-mp-rg \
-p resource_dependencies=global:qfs-mds-rs \
[-y resource_dependencies_offline_restart=global:scal-dg-rs \]
-x mountpointdir=mp-path \
-x filesystemtype=s-qfs \
-x targetfilesystem=fs-name qfs-mp-rs
```

- **Set a strong dependency by the instance of `SUNW.ScalMountPoint` on the resource for the Sun QFS metadata server for the file system.**

  The resource for the Sun QFS metadata server set is created in "How to Register and Configure Resources for the Sun QFS Metadata Server in the Global Cluster" on page 321.

- **If you are also using a volume manager, set an offline-restart dependency by the instance of `SUNW.ScalMountPoint` on the resource for the scalable device group that is to store the file system.**

  This resource is created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.

5 Bring online and in a managed state the resource group that you created in Step 2.

```
# clresourcegroup online -Z zcname -emM scal-mp-rg
```

Creating Resources for Interoperation With Oracle 10*g* or 11*g* by Using Oracle Solaris Cluster Maintenance Commands

The task in this section is an alternative for the resource-configuration steps in "How to Enable Oracle Solaris Cluster and Oracle 10*g* Release 2 or 11*g* Oracle Clusterware to Interoperate" on page 109. The following information is in this section:

Resources for interoperation with Oracle 10*g* release 2 or 11*g* enable you to administer Oracle RAC database instances by using Oracle Solaris Cluster interfaces. These resources also ensure that dependencies by Oracle Clusterware resources on Oracle Solaris Cluster resources are met. These resources enable the high-availability frameworks that are provided by Oracle Solaris Cluster software and Oracle Clusterware to interoperate.

The following resources for interoperation with Oracle 10*g* or 11*g* are required:

- A Oracle Solaris Cluster resource to act as a proxy for the Oracle RAC database
- A Oracle Solaris Cluster resource to represent the Oracle Clusterware framework
- Oracle Clusterware resources to represent scalable device groups
- Oracle Clusterware resources to represent scalable file-system mount points

You must assign to an Oracle Clusterware resource that represents an Oracle Solaris Cluster resource a name in the following form:

```
sun.node.sc-rs
```

node Specifies the name of the node where the Oracle Clusterware resource is to run.

sc-rs Specifies the name of the Oracle Solaris Cluster resource that the Oracle Clusterware resource represents.

For example, the name of the Oracle Clusterware resource for node `pclus1` that represents the Oracle Solaris Cluster resource `scal-dg-rs` must be as follows:

```
sun.pclus1.scal-dg-rs
```

FIGURE D–1 Proxy Resources for Configurations With a Volume Manager

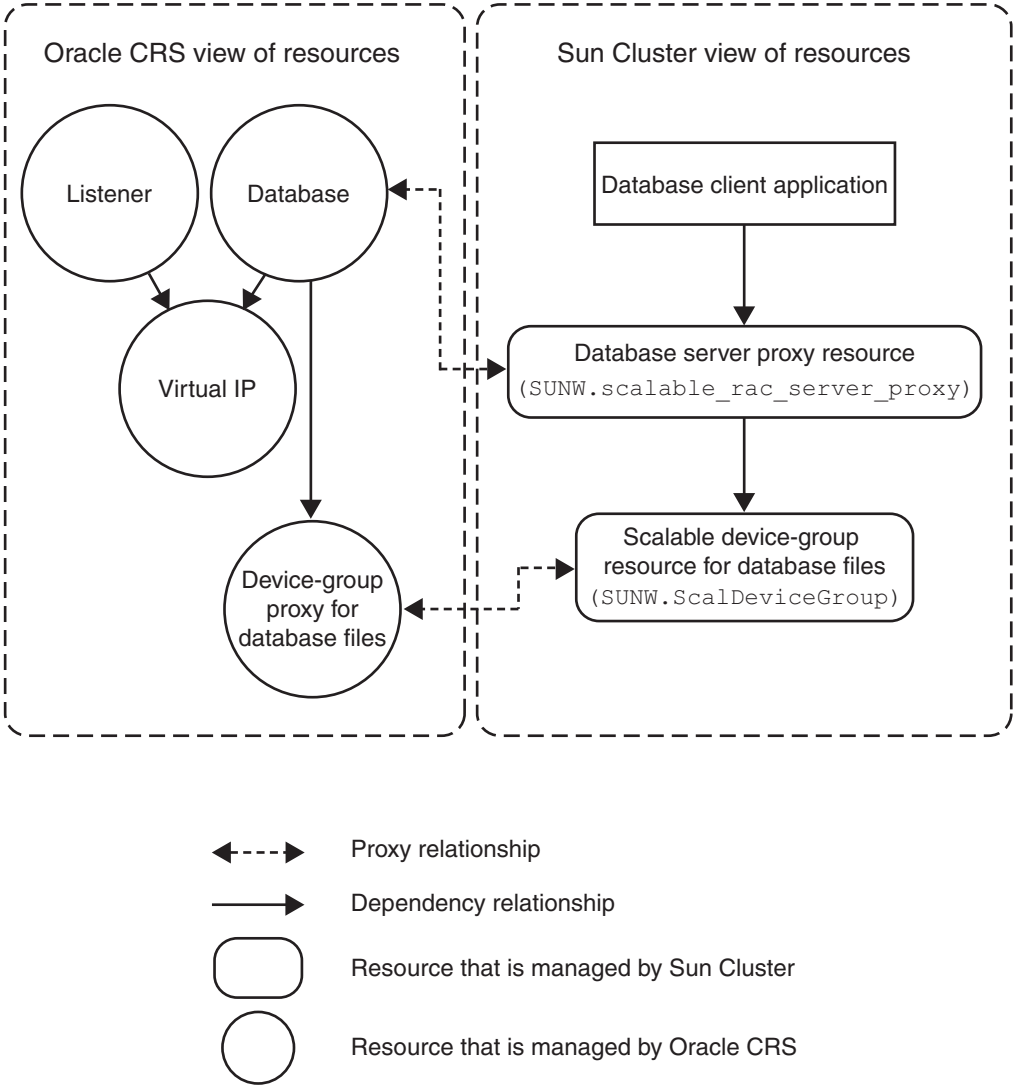
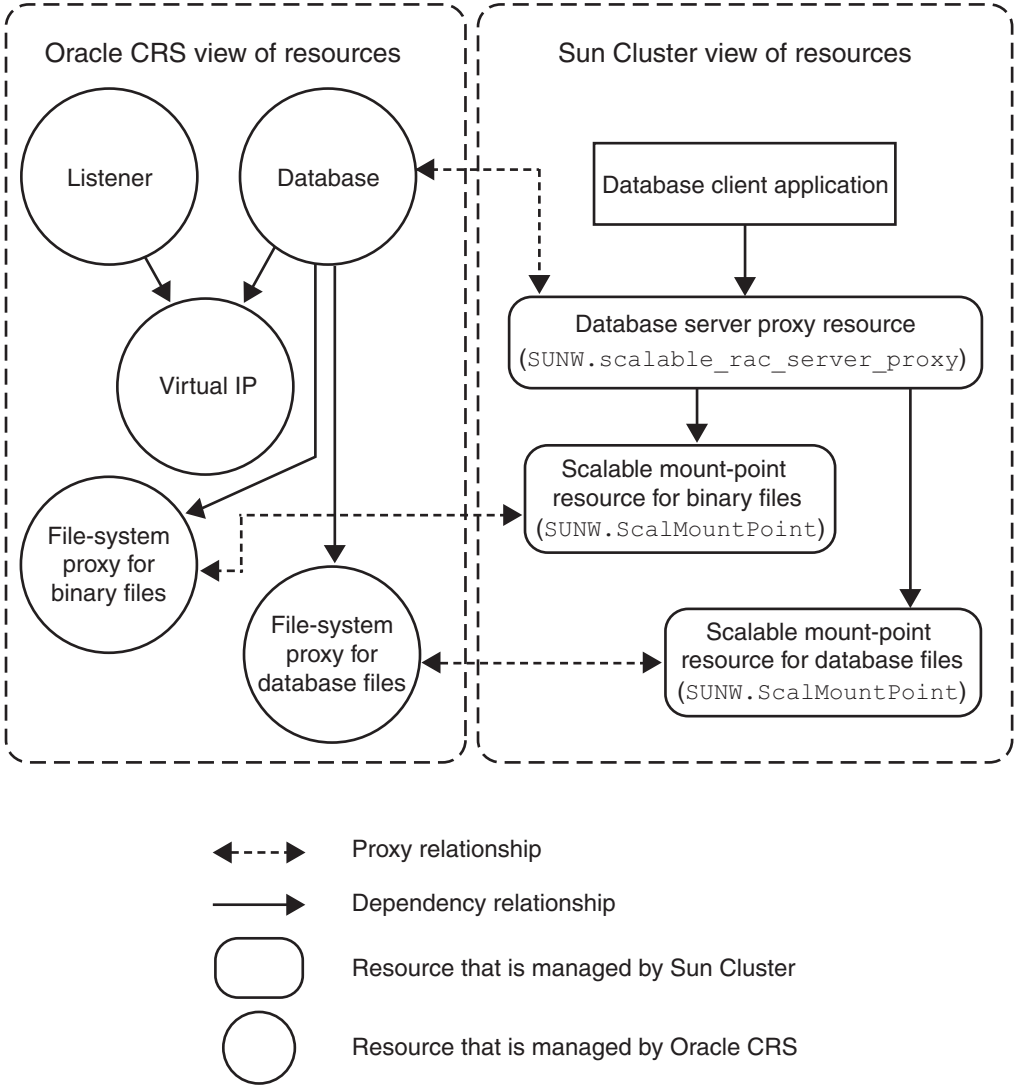


FIGURE D–2 Proxy Resources for Configurations With a Shared File System



▼ How to Create Oracle Solaris Cluster Resources for Interoperation With Oracle 10*g* or 11*g*

Perform this procedure on only one node of the cluster.

1 **Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.**

2 Register the `SUNW.crs_framework` resource type.

```
# clresourcetype register SUNW.crs_framework
```

3 Add an instance of the `SUNW.crs_framework` resource type to the RAC framework resource group.

For information about this resource group, see "Registering and Configuring the RAC Framework Resource Group" on page 65.

Set a strong dependency by the instance of `SUNW.crs_framework` on the instance of `SUNW.rac_framework` in the RAC framework resource group.

You might have configured a storage resource for the storage that you are using for database files. In this situation, set an offline-restart dependency by the instance of `SUNW.crs_framework` on the storage resource. Limit the scope of this dependency to only the node where the storage resource is running.

- If you are using a volume manager for database files, set the dependency on the resource that you created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.
- If you are using a file system for database files, set the dependency on the resource that you created in "How to Create a Resource for a File-System Mount Point in the Global Cluster" on page 324.

You might have configured a storage resource for the file system that you are using for binary files. In this situation, set an offline-restart dependency by the instance of `SUNW.crs_framework` on the storage resource. Limit the scope of this dependency to only the node where the storage resource is running. Set the dependency on the resource that you created in "How to Create a Resource for a File-System Mount Point in the Global Cluster" on page 324.

```
# clresource create -t SUNW.crs_framework \
-g rac-fmwk-rg \
-p resource_dependencies=rac-fmwk-rs \
[-p resource_dependencies_offline_restart=db-storage-rs{local_node} \
[,bin-storage-rs{local_node}]] \
crs-fmwk-rs
```

4 Create a scalable resource group to contain the proxy resource for the Oracle RAC database server.

Set a strong positive affinity by the scalable resource group for the RAC framework resource group.

You might have configured a storage resource for the storage that you are using for database files. In this situation, set a strong positive affinity by the scalable resource group for the resource group that contains the storage resource for database files.

- If you are using a volume manager for database files, set a strong positive for the resource group that you created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.
- If you are using a file system for database files, set a strong positive affinity for the resource group that you created in "How to Create a Resource for a File-System Mount Point in the Global Cluster" on page 324.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -n nodelist \
-p maximum_primaries=num-in-list \
-p desired_primaries=num-in-list \
-p rg_affinities=++rac-fmwk-rg[,db-storage-rg] \
[-p rg_description="description"] \
-p rg_mode=Scalable \
rac-db-rg
```

5 **Register the `SUNW.scalable_rac_server_proxy` resource type.**

```
# clresourcetype register SUNW.scalable_rac_server_proxy
```

6 **Add an instance of the `SUNW.scalable_rac_server_proxy` resource type to the resource group that you created in Step 4.**

Set a strong dependency by the instance of `SUNW.scalable_rac_server_proxy` on the instance of `SUNW.rac_framework` in the RAC framework resource group.

Set an offline-restart dependency by the instance of `SUNW.scalable_rac_server_proxy` on the instance of `SUNW.crs_framework` that you created in Step 3.

You might have configured a storage resource for the storage that you are using for database files. In this situation, set an offline-restart dependency by the instance of `SUNW.scalable_rac_server_proxy` on the storage resource. Limit the scope of this dependency to only the node where the storage resource is running.

- If you are using a volume manager for database files, set the dependency on the resource that you created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.
- If you are using a file system for database files, set the dependency on the resource that you created in "How to Create a Resource for a File-System Mount Point in the Global Cluster" on page 324.

Set a different value of the `oracle_sid` extension property for each node that can master the resource.

```
# clresource create -g rac-db-rg \
-t SUNW.scalable_rac_server_proxy \
-p resource_dependencies=rac-fmwk-rs \
-p resource_dependencies_offline_restart=crs-fmk-rs[, db-storage-rs] \
-p oracle_home=ora-home \
-p crs_home=crs-home \
-p db_name=db-name \
-p oracle_sid{node1-id}=sid-node1 \
[ -p oracle_sid{node2-id}=sid-node2...] \
rac-srvr-proxy-rs
```

7 Bring online the resource group that you created in Step 4.

```
# clresourcegroup online -emM rac-db-rg
```

▼ How to Create Oracle Solaris Cluster Resources in a Zone Cluster for Interoperation With Oracle 10*g* or 11*g*

Perform this procedure on only one node of the cluster.

Note – When a step in the procedure requires running the Oracle Solaris Cluster commands in a zone cluster, you should run the command from the global cluster and use the `-Z` option to specify the zone cluster.

---

1 Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.

2 Register the `SUNW.crs_framework` resource type.

```
# clresourcetype register -Z zcname SUNW.crs_framework
```

3 Add an instance of the `SUNW.crs_framework` resource type to the RAC framework resource group.

For information about this resource group, see "Registering and Configuring the RAC Framework Resource Group" on page 65.

Set a strong dependency by the instance of `SUNW.crs_framework` on the instance of `SUNW.rac_framework` in the RAC framework resource group.

You might have configured a storage resource for the storage that you are using for database files. In this situation, set an offline-restart dependency by the instance of `SUNW.crs_framework` on the storage resource. Limit the scope of this dependency to only the node where the storage resource is running.

- If you are using a volume manager for database files, set the dependency on the resource that you created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.
- If you are using a file system for database files, set the dependency on the resource that you created in "How to Create a Resource for a File-System Mount Point in Zone Cluster" on page 325.

You might have configured a storage resource for the file system that you are using for binary files. In this situation, set an offline-restart dependency by the instance of `SUNW.crs_framework` on the storage resource. Limit the scope of this dependency to only the node where the storage resource is running. Set the dependency on the resource that you created in "How to Create a Resource for a File-System Mount Point in Zone Cluster" on page 325.

```
# clresource create -Z zcname -t SUNW.crs_framework \
-g rac-fmwk-rg \
-p resource_dependencies=rac-fmwk-rs \
[-p resource_dependencies_offline_restart=db-storage-rs{local_node} \
[,bin-storage-rs{local_node}]] \
crs-fmwk-rs
```

4 Create a scalable resource group to contain the proxy resource for the Oracle RAC database server.

Set a strong positive affinity by the scalable resource group for the RAC framework resource group.

You might have configured a storage resource for the storage that you are using for database files. In this situation, set a strong positive affinity by the scalable resource group for the resource group that contains the storage resource for database files.

- If you are using a volume manager for database files, set a strong positive affinity for the resource group that you created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.
- If you are using a file system for database files, set a strong positive affinity for the resource group that you created in "How to Create a Resource for a File-System Mount Point in Zone Cluster" on page 325.

Tip – If you require Support for Oracle RAC to run on all cluster nodes, specify the `-S` option in the command that follows and omit the options `-n`, `-p maximum_primaries`, `-p desired_primaries`, and `-p rg_mode`.

```
# clresourcegroup create -Z zcname -n nodelist \
-p maximum_primaries=num-in-list \
-p desired_primaries=num-in-list \
-p rg_affinities=++rac-fmwk-rg[,db-storage-rg] \
[-p rg_description="description"] \
-p rg_mode=Scalable \
rac-db-rg
```

5 Register the `SUNW.scalable_rac_server_proxy` resource type.

```
# clresourcetype register -Z zcname SUNW.scalable_rac_server_proxy
```

6 Add an instance of the `SUNW.scalable_rac_server_proxy` resource type to the resource group that you created in Step 4.

Set a strong dependency by the instance of `SUNW.scalable_rac_server_proxy` on the instance of `SUNW.rac_framework` in the RAC framework resource group.

Set an offline-restart dependency by the instance of `SUNW.scalable_rac_server_proxy` on the instance of `SUNW.crs_framework` that you created in Step 3.

You might have configured a storage resource for the storage that you are using for database files. In this situation, set an offline-restart dependency by the instance of `SUNW.scalable_rac_server_proxy` on the storage resource. Limit the scope of this dependency to only the node where the storage resource is running.

- If you are using a volume manager for database files, set the dependency on the resource that you created in "How to Create a Resource for a Scalable Device Group in the Global Cluster" on page 319.
- If you are using a file system for database files, set the dependency on the resource that you created in "How to Create a Resource for a File-System Mount Point in Zone Cluster" on page 325.

Set a different value of the `oracle_sid` extension property for each node that can master the resource.

```
# clresource create -Z zcname -g rac-db-rg \
-t SUNW.scalable_rac_server_proxy \
-p resource_dependencies=rac-fmwk-rs \
-p resource_dependencies_offline_restart=crs-fmk-rs \
[, db-storage-rs, bin-storage-rs] \
-p oracle_home=ora-home \
-p crs_home=crs-home \
-p db_name=db-name \
-p oracle_sid{node1-id}=sid-node1 \
[ -p oracle_sid{node2-id}=sid-node2...] \
rac-srvr-proxy-rs
```

7 Bring online the resource group that you created in Step 4.

```
# clresourcegroup online -Z zcname -emM rac-db-rg
```

▼ How to Create an Oracle Clusterware Resource for Interoperation With Sun Cluster

Oracle Clusterware resources are similar to Oracle Solaris Cluster resources. Oracle Clusterware resources represent items that the Oracle Clusterware manage in a similar way to how Oracle Solaris Cluster resources represent items that the Oracle Solaris Cluster RGM manages.

Depending on your configuration, some Oracle components that are represented as Oracle Clusterware resources might depend on file systems and global devices that Oracle Solaris Cluster manages. For example, if you are using file systems and global devices for Oracle files, the Oracle RAC database and the Oracle listener might depend on these file systems and global devices.

Create an Oracle Clusterware resource for each Oracle Solaris Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend. The Oracle Clusterware resources that you create track the status of their associated Oracle Solaris Cluster resources. The Oracle Clusterware resources also ensure the orderly startup of Oracle Clusterware resources.

Perform this task on each cluster node where Support for Oracle RAC is to run.

Note – Some steps in this procedure require you to use Oracle Clusterware commands. In these steps, the syntax of the command for Oracle release 10*g* release 2 or 11*g* is provided. If you are you are using a version of Oracle other than 10*g* release 2 or 11*g*, see your Oracle documentation for the correct command syntax.

Note – To create an Oracle Clusterware resource in a zone cluster, you should perform the steps in this procedure in that zone cluster.

---

1 On the node where you are performing this task, global-cluster node for a global cluster or zone-cluster node for a zone cluster, become superuser.

2 If the `/var/cluster/ucmm/profile` directory does not exist, create it.

Profiles for Oracle Clusterware resources are created in this directory.

```
# mkdir -p /var/cluster/ucmm/profile
```

3 Create a profile for the Oracle Clusterware resource.

```
# crs-home/bin/crs_profile \
-create sun.node.sc-rs \
-t application -d "description" \
```

```
-dir /var/cluster/ucmm/profile \
-a /opt/SUNWscor/dsconfig/bin/scproxy_crs_action \
-p restricted -h node -f -o st=1800
```

4. **Register the Oracle Clusterware resource for which you created a profile in Step 3.**

   ```
   # crs-home/bin/crs_register sun.node.sc-rs \
   -dir /var/cluster/ucmm/profile
   ```

5. **Ensure that the Oracle Solaris Cluster resource for which the Oracle Clusterware resource is a proxy is online.**

   a. **Obtain the state of the Oracle Solaris Cluster resource.**

      ```
      # clresource status sc-rs
      ```

   b. **If the state of the Oracle Solaris Cluster resource is *not* online, bring online the resource group that contains the Oracle Solaris Cluster resource.**

      If the state of the Oracle Solaris Cluster resource is online, omit this step.

      ```
      # clresourcegroup online -emM sc-rg
      ```

6. **Start the Oracle Clusterware resource that you registered in Step 4.**

   ```
   # crs-home/bin/crs_start sun.node.sc-rs
   ```

7. **Add the Oracle Clusterware resource that you registered in Step 4 to the list of resources that the dependent Oracle Clusterware resource requires.**

   a. **If the dependent Oracle Clusterware resource is the Oracle RAC database instance, obtain the name of the instance.**

      ```
      # crs-home/bin/srvctl config database -d db-name | grep node
      ```

   b. **Obtain the list of resources that the dependent Oracle Clusterware resource requires.**

      ```
      # crs-home/bin/crs_stat -p depend-crs-rs | grep REQUIRED_RESOURCES
      ```

   c. **Append the name of the Oracle Clusterware resource to the list that you obtained in Step b.**

      ```
      # crs-home/bin/crs_register depend-crs-rs \
      -update -r "existing-list sun.node.sc-rs"
      ```

▼ How to Create an Oracle Clusterware Oracle ASM Resource for Interoperation With Sun Cluster Software

Depending on your configuration, some Oracle components that are represented as Oracle ASM resources might depend on file systems and global devices that Oracle Solaris Cluster manages. Create an Oracle Clusterware Oracle ASM resource for each Oracle Solaris Cluster resource for scalable device groups and scalable file-system mount points on which Oracle

components depend. The Oracle Clusterware Oracle ASM resources that you create track the status of their associated Oracle Solaris Cluster resources. The Oracle Clusterware Oracle ASM resources also ensure the orderly startup of Oracle Clusterware Oracle ASM resources.

Note – Some steps in this procedure require you to use Oracle Clusterware commands. In these steps, the syntax of the command for Oracle 10*g* release 2 or 11*g* is provided. If you are using a version of Oracle other than 10*g* release 2 or 11*g*, see your Oracle documentation for the correct command syntax.

Note – To create an Oracle Clusterware Oracle ASM resource in a zone cluster, perform the steps in this procedure from that zone cluster.

---

1 Become superuser.

2 On each node where Support for Oracle RAC is to run, create the `/var/cluster/ucmm/profile` directory, if it does not already exist.

```
# mkdir -p /var/cluster/ucmm/profile
```

Profiles for Oracle Clusterware resources are created in this directory.

3 Create a profile for the Oracle Clusterware Oracle ASM disk-group resource.

```
# crs-home/bin/crs_profile \
-create sun.node.asm-dg-rs \
-t application -d "description" \
-dir /var/cluster/ucmm/profile \
-a /opt/SUNWscor/dsconfig/bin/scproxy_crs_action \
-p restricted -h node -f -o ra=60,st=20
```

asm-dg-rs
Specifies the name of the Oracle Clusterware Oracle ASM disk-group resource.

4 Register the Oracle Clusterware Oracle ASM disk-group resource for which you created a profile in Step 3.

```
# crs-home/bin/crs_register sun.node.asm-dg-rs \
-dir /var/cluster/ucmm/profile
```

5 Ensure that the Oracle Solaris Cluster resource, for which the Oracle Clusterware Oracle ASM disk-group resource is a proxy, is online.

a. Obtain the state of the Oracle Solaris Cluster resource.

```
# clresource status asm-dg-rs
```

b. If the state of the Oracle Solaris Cluster resource is *not* online, bring online the resource group that contains the Oracle Solaris Cluster resource.

If the state of the Oracle Solaris Cluster resource is online, omit this step.

```
# clresourcegroup online -emM asm-dg-rg
```

6 Start the Oracle Clusterware Oracle ASM disk-group resource that you registered in Step 4.

```
# crs-home/bin/crs_start sun.node.asm-dg-rs
```

7 On each node where Support for Oracle RAC is to run, add a dependency for the Oracle Clusterware Oracle ASM disk-group resource.

Add the Oracle Clusterware Oracle ASM disk-group resource that you registered in Step 4 to the list of resources that the dependent Oracle Clusterware Oracle ASM disk-group resource requires.

a. Obtain the name of the Oracle Clusterware RAC instance resource.

```
# crs-home/bin/srvctl config database -d db-name | grep node
```

b. List the dependencies for the Oracle Clusterware RAC instance resource.

```
# crs-home/bin/crs_stat -p ora.db-name.instance.inst | grep REQUIRED_RESOURCES
```

c. Add the name of the Oracle Clusterware Oracle ASM disk-group resource to the list that you obtained in Step b.

```
# crs-home/bin/crs_register ora.db-name.instance.inst \
-update -r "existing-list sun.node.asm-dg-rs"
```

8 On each node where Support for Oracle RAC is to run, add a dependency for the Oracle Clusterware Oracle ASM instance resource.

Add the Oracle Clusterware Oracle ASM instance resource as a depended-on resource that the Oracle Clusterware Oracle ASM disk group requires.

a. Obtain the name for the Oracle Clusterware Oracle ASM instance resource.

```
# crs-home/bin/crs_stat -p | grep -i asm | grep node
```

b. Add the Oracle Clusterware Oracle ASM instance resource as a depended-on resource to the Oracle Clusterware Oracle ASM disk-group resource.

```
# crs-home/bin/crs_register -p sun.node.asm-dg-rs
-update -r crs-asm-instance
```

Registering and Configuring Oracle Solaris Cluster Resources for Interoperation With Oracle 9*i* by Using Oracle Solaris Cluster Maintenance Commands

The task in this section is an alternative for the resource-configuration steps in "How to Automate the Startup and Shutdown of Oracle 9*i* RAC Database Instances" on page 115. The following information is in this section:

Resources for interoperation with Oracle 9*i* enable you to administer Oracle RAC database instances by using Oracle Solaris Cluster interfaces. These resources also provide fault monitoring and automatic fault recovery for Oracle RAC. The automatic fault recovery that this data service provides supplements the automatic fault recovery that the Oracle RAC software provides.

The following resources for interoperation with Oracle 9*i* are required:

- Oracle RAC server resources
- Oracle listener resources
- Logical hostname resources

Oracle 9*i* RAC Server Resources

Note – If you are using Oracle 10*g* or 11*g*, no Oracle RAC server resources are required. For more information, see "Creating Resources for Interoperation With Oracle 10*g* or 11*g* by Using Oracle Solaris Cluster Maintenance Commands" on page 327.

You require one scalable resource group for each Oracle RAC database. Each resource group contains the Oracle RAC server resource that represents all instances of the database in the cluster. Ensure that this scalable resource group is mastered on all nodes where Oracle RAC is to run.

Oracle 9*i* Listener Resources

Note – If you are using Oracle 10*g* or 11*g*, no Oracle listener resources are required. For more information, see "Creating Resources for Interoperation With Oracle 10*g* or 11*g* by Using Oracle Solaris Cluster Maintenance Commands" on page 327.

If your configuration of Oracle RAC requires Oracle listeners, configure each listener to serve only one Oracle RAC database instance. This configuration provides the highest availability and scalability, and the easiest management.

Note – Not all configurations of Oracle RAC require Oracle listeners. For example, if the Oracle RAC database server and the database client are running on the same machine, no Oracle listeners are required.

If your configuration includes Oracle listeners, configure one scalable resource to represent all listeners that serve a specific Oracle RAC database. Configure the listener resource as follows:

- Configure the listener resource and the RAC server resource in the same resource group.
- Set the listener resource's listener name to a unique value for each node.
- Set the RAC server resource to depend on the listener resource.

Logical Hostname Resources for Oracle 9*i* Listener Resources

Note – If you are using Oracle 10*g* or 11*g*, no `LogicalHostname` resources are required.

To ensure that Oracle listeners can continue to access the database after failure of an instance on a node, each node requires a logical hostname resource. On each node, the scalable Oracle listener listens on an IP address that is represented by the logical hostname resource.

If a cluster node that is running an instance of Oracle RAC fails, an operation that a client application attempted might be required to time out before the operation is attempted again on another instance. If the Transmission Control Protocol/Internet Protocol (TCP/IP) network timeout is high, the client application might require a significant length of time to detect the failure. Typically, client applications require between three and nine minutes to detect such failures.

In such situations, client applications can connect to listener resources that are listening on an address that is represented by the Oracle Solaris Cluster logical hostname resource. If a node fails, the resource group that contains the logical hostname resource fails over to another surviving node on which Oracle RAC is running. The failover of the logical hostname resource enables new connections to be directed to the other instance of Oracle RAC.

Configure `LogicalHostname` resources for each listener resource as follows:

- Create one logical hostname resource for each listener that the listener resource represents.
- Configure each logical hostname resource in a separate resource group.
- Set the listener resource to depend on the logical hostname resources for all listeners that the listener resource represents.
- Ensure that each node is the primary node of one resource group.
- Ensure that the logical hostname resource is failed back to the primary node when the database instance on the primary node recovers after a failure.

▼ How to Register and Configure Oracle Solaris Cluster Resources in a Global Cluster for Interoperation With Oracle 9*i*

The `SUNW.scalable_rac_server` resource type represents the Oracle RAC server in a Sun Cluster configuration.

Oracle RAC server instances should be started only after the RAC framework is enabled on a cluster node. You ensure that this requirement is met by creating the following affinities and dependencies:

- A strong positive affinity between the Oracle RAC server resource group and the RAC framework resource group
- A dependency between the Oracle RAC server resource and the RAC framework resource

Perform this procedure on only one node of the cluster.

1 Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.

2 Create the logical hostname resources to represent the IP addresses on which the Oracle listeners are to listen.

Each node where Support for Oracle RAC can run requires a logical hostname resource. Create each logical hostname resource as follows:

a. Create a failover resource group to contain the logical hostname resource.

Set the properties of the resource group as follows:

- Specify the node for which you are creating the logical hostname resource as the primary node.
- Specify the remaining nodes where Support for Oracle RAC can run as potential primary nodes.
- Choose an order for the potential primary nodes that ensures that the logical hostname resources are distributed equally throughout the cluster.
- Ensure that the resource group is failed back to the primary node when the database instance on the primary node recovers after a failure.

```
# clresourcegroup create -n nodelist -p failback=true \
[-p rg_description="description"] \
lh-name-rg
```

`-n` *nodelist*	Specifies a comma-separated list of names of the nodes that can master this resource group. Ensure that the node for which you are creating the logical hostname resource appears first in the list. Choose an order for the remaining nodes that ensures that the logical hostname resources are distributed equally throughout the cluster.
`-p rg_description=`"*description*"	Specifies an optional brief description of the resource group. This description is displayed when you use Oracle Solaris Cluster maintenance commands to obtain information about the resource group.
lh-name-rg	Specifies your choice of name to assign to the resource group.

b. Add a logical hostname resource to the resource group that you created in Step a.

```
# clreslogicalhostname create -h lh-name -g lh-name-rg lh-name-rs
```

`-h` *lh-name*	Specifies the logical hostname that this resource is to make available. An entry for this logical hostname must exist in the name service database.
`-g`*lh-name-rg*	Specifies that you are adding the resource to the resource group that you created in Step 2.
lh-name-rs	Specifies your choice of name to assign to the logical hostname resource.

3 Create a scalable resource group to contain the Oracle RAC server resource and Oracle listener resource.

```
# clresourcegroup create -n nodelist \
-p maximum_primaries=num-in-list \
-p desired_primaries=num-in-list \
-p rg_affinities=++rac-fmwk-rg \
[-p rg_description="description"] \
-p rg_mode=Scalable rac-db-rg
```

`-n` *nodelist*
Specifies a comma-separated list of cluster nodes on which Support for Oracle RAC is to be enabled. The Support for Oracle RAC software packages must be installed on each node in this list.

`-p maximum_primaries=`*num-in-list*
Specifies the number of nodes on which Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.

`-p desired_primaries=`*num-in-list*
Specifies the number of nodes on which Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.

`-p rg_affinities=++`*rac-fmwk-rg*
Creates a strong positive affinity to the RAC framework resource group. If the RAC framework resource group was created by using the `clsetup` utility, the RAC framework resource group is named `rac-framework-rg`.

`-p rg_description="`*description*"
Specifies an optional brief description of the resource group. This description is displayed when you use Oracle Solaris Cluster maintenance commands to obtain information about the resource group.

`-p rg_mode=Scalable`
Specifies that the resource group is scalable.

rac-db-rg
Specifies the name that you are assigning to the resource group.

4 Register the `SUNW.scalable_rac_listener` resource type.

```
# clresourcetype register SUNW.scalable_rac_listener
```

5 Add an instance of the `SUNW.scalable_rac_listener` resource type to the resource group that you created in Step 3.

When you create this resource, specify the following information about the resource:

- The name of the Oracle listener on each node where Oracle RAC is to run. This name must match the corresponding entry in the `listener.ora` file for the node.

- The Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

```
# clresource create -g rac-db-rg \
-t SUNW.scalable_rac_listener \
-p resource_dependencies_weak=lh-rs-list \
[-p resource_dependencies=db-bin-rs] \
-p listener_name{node}=listener[...] \
-p oracle_home=ora-home \
rac-lsnr-rs
```

`-g` *rac-db-rg*
: Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 3.

`[-p resource_dependencies=`*db-bin-rs*`]`
: Specifies that this Oracle listener resource has a strong dependency on the storage resource for binary files. Specify this dependency only if you are using the Sun QFS shared file system or a qualified NAS device for Oracle binary files. The storage resource for Oracle binary files is created when you perform the tasks in "Registering and Configuring Storage Resources for Oracle Files" on page 82.

`-p listener_name{`*node*`}=`*ora-sid*
: Specifies the name of the Oracle listener instance on node *node*. This name must match the corresponding entry in the `listener.ora` file.

`-p resource_dependencies_weak=`*lh-rs-list*
: Specifies a comma-separated list of resources on which this resource is to have a weak dependency. The list must contain all the logical hostname resources that you created in Step 2.

`-p oracle_home=`*ora-home*
: Specifies the path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

rac-lsnr-rs
: Specifies the name that you are assigning to the `SUNW.scalable_rac_listener` resource.

6 Register the `SUNW.scalable_rac_server` resource type.

```
# clresourcetype register SUNW.scalable_rac_server
```

7 Add an instance of the `SUNW.scalable_rac_server` resource type to the resource group that you created in Step 3.

When you create this resource, specify the following information about the resource:

- The Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.
- The Oracle system identifier on each node where Oracle RAC is to run. This identifier is the name of the Oracle database instance on the node.

- The full path to the alert log file on each node where Oracle RAC is to run.

```
# clresource create -g rac-db-rg \
-t SUNW.scalable_rac_server \
-p resource_dependencies=rac-fmwk-rs \
-p resource_dependencies_offline_restart=[db-storage-rs][,db-bin-rs] \
-p resource_dependencies_weak=rac-lsnr-rs \
-p oracle_home=ora-home \
-p connect_string=string \
-p oracle_sid{node}=ora-sid[...] \
-p alert_log_file{node}=al-file[...] \
rac-srvr-rs
```

`-g` *rac-db-rg*
: Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 3.

`-p resource_dependencies=`*rac-fmwk-rs*
: Specifies the resources on which this Oracle RAC server resource has a strong dependency.

  You must specify the RAC framework resource. If the RAC framework resource group is created by using the `clsetup` utility or Oracle Solaris Cluster Manager, this resource is named `rac-framework-rs`.

  If you are using a volume manager or the Sun QFS shared file system for database files, you must also specify the storage resource for database files.

  If you are using the Sun QFS shared file system for Oracle binary files, you must also specify the storage resource for binary files.

  The storage resources for Oracle files are created when you perform the tasks in "Registering and Configuring Storage Resources for Oracle Files" on page 82.

`-p resource_dependencies_weak=`*rac-lsnr-rs*
: Specifies a weak dependency by this Oracle RAC server resource on the Oracle listener resource that you created in Step 5.

`-p oracle_sid{`*node*`}=`*ora-sid*
: Specifies the Oracle system identifier on node *node*. This identifier is the name of the Oracle database instance on the node. You must set a different value for this property on each node where Oracle RAC is to run.

`-p oracle_home=`*ora-home*
: Specifies the path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

`-p connect_string=`*string*
: Specifies the Oracle database user ID and password that the fault monitor uses to connect to the Oracle database. *string* is specified as follows:

  userid/*password*

userid
Specifies the Oracle database user ID that the fault monitor uses to connect to the Oracle database.

password
Specifies the password that is set for the Oracle database user *userid*.

The database user ID and password are defined during the setup of Oracle RAC. To use Solaris authentication, type a slash (`/`) instead of a user ID and password.

rac-srvr-rs
Specifies the name that you are assigning to the `SUNW.scalable_rac_server` resource.

8 Bring online the resource group that you created in Step 3.

```
# clresourcegroup online -emM rac-db-rg
```

rac-db-rg Specifies that a resource group that you created in Step 3 is to be moved to the `MANAGED` state and brought online.

Example D–1 Registering and Configuring Oracle Solaris Cluster Resources for Interoperation With Oracle 9*i*

This example shows the sequence of operations that is required to register and configure Oracle Solaris Cluster resources for interoperation with Oracle 9*i* on a two-node cluster.

The example makes the following assumptions:

- Support for Oracle RAC is to run on all cluster nodes.
- The C shell is used.
- A RAC framework resource group that is named `rac-framework-rg` exists and contains a resource of type `SUNW.rac_framework` that is named `rac_framework-rs`.
- A resource of type `SUNW.ScalDeviceGroup` that is named `db-storage-rs` represents the Solaris Volume Manager for Sun Cluster multi-owner disk set where the Oracle database files are stored.
- The Oracle binary files are installed on the cluster file system, for which no storage resource is required.

1. To create the logical hostname resource for node `phys-schost-1`, the following commands are run:

```
# clresourcegroup create -n phys-schost-1,phys-schost-2 -p failback=true \
-p rg_description="Logical hostname schost-1 RG" \
schost-1-rg
# clreslogicalhostname create -h schost-1 -g schost-1-rg schost-1
```

2. To create the logical hostname resource for node `phys-schost-2`, the following commands are run:

```
# clresourcegroup create -n phys-schost-2,phys-schost-1 -p failback=true \
-p rg_description="Logical hostname schost-2 RG" \
schost-2-rg
# clreslogicalhostname create -h schost-2 -g schost-2-rg schost-2
```

3. To create a scalable resource group to contain the Oracle RAC server resource and Oracle listener resource, the following command is run:

```
# clresourcegroup create -S \
-p rg_affinities=++rac_framework-rg \
-p rg_description="RAC 9i server and listener RG" \
rac-db-rg
```

4. To register the `SUNW.scalable_rac_listener` resource type, the following command is run:

```
# clresourcetype register SUNW.scalable_rac_listener
```

5. To add an instance of the `SUNW.scalable_rac_listener` resource type to the `rac-db-rg` resource group, the following command is run:

```
# clresource create -g rac-db-rg \
-t SUNW.scalable_rac_listener \
-p resource_dependencies_weak=schost-1,schost-2 \
-p listener_name\{phys-schost-1\}=LISTENER1 \
-p listener_name\{phys-schost-2\}=LISTENER2 \
-p oracle_home=/home/oracle/product/9.2.0 \
scalable_rac_listener-rs
```

   A different value of the `listener_name` extension property is set for each node that can master the resource.

6. To register the `SUNW.scalable_rac_server` resource type, the following command is run:

```
# clresourcetype register SUNW.scalable_rac_server
```

7. To add an instance of the `SUNW.scalable_rac_listener` resource type to the `rac-db-rg` resource group, the following command is run:

```
# clresource create -g rac-db-rg \
-t SUNW.scalable_rac_server \
-p resource_dependencies=rac_framework-rs, db-storage-rs \
-p resource_dependencies_weak=scalable_rac_listener-rs \
-p oracle_home=/home/oracle/product/9.2.0 \
-p connect_string=scooter/t!g3r \
-p oracle_sid\{phys-schost-1\}=V920RAC1 \
-p oracle_sid\{phys-schost-2\}=V920RAC2 \
-p alert_log_file\{phys-schost-1\}=/home/oracle/9.2.0/rdbms/log/alert_V920RAC1.log \
-p alert_log_file\{phys-schost-2\}=/home/oracle/9.2.0/rdbms/log/alert_V920RAC2.log \
scalable_rac_server-rs
```

   A different value of the following extension properties is set for each node that can master the resource:

   - `alert_log_file`
   - `oracle_sid`

8. To bring online the resource group that contains the Oracle RAC server resource and Oracle listener resource, the following command is run:

```
# clresourcegroup online -emM rac-db-rg
```

Next Steps Go to "Verifying the Installation and Configuration of Support for Oracle RAC" on page 120.

How to Register and Configure Oracle Solaris Cluster Resources in a Zone Cluster for Interoperation With Oracle 9*i*

Perform the steps in this procedure to register and configure Oracle Solaris Cluster resources in a zone cluster for interoperation with Oracle 9*i*.

1 Become superuser or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorizations.

2 Create the logical hostname resources to represent the IP addresses on which the Oracle listeners are to listen.

Each node where Support for Oracle RAC can run requires a logical hostname resource. Create each logical hostname resource as follows:

a. Create a failover resource group to contain the logical hostname resource.

Set the properties of the resource group as follows:

- Specify the node for which you are creating the logical hostname resource as the primary node.
- Specify the remaining nodes where Support for Oracle RAC can run as potential primary nodes.
- Choose an order for the potential primary nodes that ensures that the logical hostname resources are distributed equally throughout the cluster.
- Ensure that the resource group is failed back to the primary node when the database instance on the primary node recovers after a failure.

```
# clresourcegroup create -Z zcname -n nodelist -p failback=true \
[-p rg_description="description"] \
lh-name-rg
```

`-n` *nodelist* — Specifies a comma-separated list of names of the nodes that can master this resource group. Ensure that the node for which you are creating the logical hostname resource appears first in the list. Choose an

	order for the remaining nodes that ensures that the logical hostname resources are distributed equally throughout the cluster.
`-p rg_description="`*description*`"`	Specifies an optional brief description of the resource group. This description is displayed when you use Oracle Solaris Cluster maintenance commands to obtain information about the resource group.
lh-name-rg	Specifies your choice of name to assign to the resource group.

b. Add a logical hostname resource to the resource group that you created in Step a.

```
# clreslogicalhostname create -Z zcname -h lh-name -g lh-name-rg lh-name-rs
```

`-h` *lh-name*	Specifies the logical hostname that this resource is to make available. An entry for this logical hostname must exist in the name service database.
`-g` *lh-name-rg*	Specifies that you are adding the resource to the resource group that you created in Step 2.
lh-name-rs	Specifies your choice of name to assign to the logical hostname resource.

3 Create a scalable resource group to contain the Oracle RAC server resource and Oracle listener resource.

```
# clresourcegroup create -Z zcname -n nodelist \
-p maximum_primaries=num-in-list \
-p desired_primaries=num-in-list \
-p rg_affinities=++rac-fmwk-rg \
[-p rg_description="description"] \
-p rg_mode=Scalable rac-db-rg
```

`-n` *nodelist*
Specifies a comma-separated list of cluster nodes on which Support for Oracle RAC is to be enabled. The Support for Oracle RAC software packages must be installed on each node in this list.

`-p maximum_primaries=`*num-in-list*
Specifies the number of nodes on which Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.

`-p desired_primaries=`*num-in-list*
Specifies the number of nodes on which Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.

`-p rg_affinities=++`*rac-fmwk-rg*
Creates a strong positive affinity to the RAC framework resource group. If the RAC framework resource group was created by using the `clsetup` utility, the RAC framework resource group is named `rac-framework-rg`.

`-p rg_description="`*description*`"`
Specifies an optional brief description of the resource group. This description is displayed when you use Oracle Solaris Cluster maintenance commands to obtain information about the resource group.

`-p rg_mode=Scalable`
Specifies that the resource group is scalable.

rac-db-rg
Specifies the name that you are assigning to the resource group.

4 Register the `SUNW.scalable_rac_listener` resource type.

```
# clresourcetype register -Z zcname SUNW.scalable_rac_listener
```

5 Add an instance of the `SUNW.scalable_rac_listener` resource type to the resource group that you created in Step 3.

When you create this resource, specify the following information about the resource:

- The name of the Oracle listener on each node where Oracle RAC is to run. This name must match the corresponding entry in the `listener.ora` file for the node.
- The Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

```
# clresource create -Z zcname -g rac-db-rg \
-t SUNW.scalable_rac_listener \
-p resource_dependencies_weak=lh-rs-list \
[-p resource_dependencies=db-bin-rs] \
-p listener_name{node}=listener[...] \
-p oracle_home=ora-home \
rac-lsnr-rs
```

`-g` *rac-db-rg*
Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 3.

`[-p resource_dependencies=`*db-bin-rs*`]`
Specifies that this Oracle listener resource has a strong dependency on the storage resource for binary files. Specify this dependency only if you are using the Sun QFS shared file system for Oracle binary files. The storage resource for Oracle binary files is created when you perform the tasks in "Registering and Configuring Storage Resources for Oracle Files" on page 82.

`-p listener_name{`*node*`}=`*ora-sid*
Specifies the name of the Oracle listener instance on node *node*. This name must match the corresponding entry in the `listener.ora` file.

`-p resource_dependencies_weak=`*lh-rs-list*
Specifies a comma-separated list of resources on which this resource is to have a weak dependency. The list must contain all the logical hostname resources that you created in Step 2.

`-p oracle_home=`*ora-home*
Specifies the path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

rac-lsnr-rs
Specifies the name that you are assigning to the `SUNW.scalable_rac_listener` resource.

6 Register the `SUNW.scalable_rac_server` resource type.

```
# clresourcetype register -Z zcname SUNW.scalable_rac_server
```

7 Add an instance of the `SUNW.scalable_rac_server` resource type to the resource group that you created in Step 3.

When you create this resource, specify the following information about the resource:

- The Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.
- The Oracle system identifier on each node where Oracle RAC is to run. This identifier is the name of the Oracle database instance on the node.
- The full path to the alert log file on each node where Oracle RAC is to run.

```
# clresource create -Z zcname -g rac-db-rg \
-t SUNW.scalable_rac_server \
-p resource_dependencies=rac-fmwk-rs \
-p resource_dependencies_offline_restart=[db-storage-rs][,db-bin-rs] \
-p resource_dependencies_weak=rac-lsnr-rs \
-p oracle_home=ora-home \
-p connect_string=string \
-p oracle_sid{node}=ora-sid[...] \
-p alert_log_file{node}=al-file[...] \
rac-srvr-rs
```

`-g` *rac-db-rg*
Specifies the resource group to which you are adding the resource. This resource group must be the resource group that you created in Step 3.

`-p resource_dependencies=`*rac-fmwk-rs*
Specifies the resources on which this Oracle RAC server resource has a strong dependency.

You must specify the RAC framework resource. If the RAC framework resource group is created by using the `clsetup` utility or Oracle Solaris Cluster Manager, this resource is named `rac-framework-rs`.

If you are using a volume manager or the Sun QFS shared file system for database files, you must also specify the storage resource for database files.

If you are using the Sun QFS shared file system for Oracle binary files, you must also specify the storage resource for binary files.

The storage resources for Oracle files are created when you perform the tasks in "Registering and Configuring Storage Resources for Oracle Files" on page 82.

`-p resource_dependencies_weak=`*rac-lsnr-rs*
Specifies a weak dependency by this Oracle RAC server resource on the Oracle listener resource that you created in Step 5.

`-p oracle_sid{`*node*`}=`*ora-sid*
Specifies the Oracle system identifier on node *node*. This identifier is the name of the Oracle database instance on the node. You must set a different value for this property on each node where Oracle RAC is to run.

`-p oracle_home=`*ora-home*
Specifies the path to the Oracle home directory. The Oracle home directory contains the binary files, log files, and parameter files for the Oracle software.

`-p connect_string=`*string*
Specifies the Oracle database user ID and password that the fault monitor uses to connect to the Oracle database. *string* is specified as follows:

userid/*password*

userid
Specifies the Oracle database user ID that the fault monitor uses to connect to the Oracle database.

password
Specifies the password that is set for the Oracle database user *userid*.

The database user ID and password are defined during the setup of Oracle RAC. To use Solaris authentication, type a slash (`/`) instead of a user ID and password.

rac-srvr-rs
Specifies the name that you are assigning to the `SUNW.scalable_rac_server` resource.

8 Bring online the resource group that you created in Step 3

```
# clresourcegroup online -Z zcname -emM rac-db-rg
```

rac-db-rg Specifies that a resource group that you created in Step 3 is to be moved to the `MANAGED` state and brought online.

Index

D

E

F

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z