



Sun Cluster Data Service for Oracle RAC Guide for Solaris OS



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 820-2574-10
February 2008, Revision A

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Sun StorEdge, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Sun StorEdge, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

Contents

Preface	15
1 Installing Sun Cluster Support for Oracle RAC	19
Overview of the Installation Process for Sun Cluster Support for Oracle RAC	19
Preinstallation Considerations	20
Hardware and Software Requirements	21
Storage Management Requirements for Oracle Files	21
SPARC: Processor Architecture Requirements for Oracle Components	24
Using Oracle Data Guard With Sun Cluster Support for Oracle RAC	25
Using the Oracle Real Application Clusters Guard Option With Sun Cluster 3.2	25
Preparing the Sun Cluster Nodes	26
Before You Begin	26
▼ How to Bypass the NIS Name Service	26
▼ How to Create the DBA Group and the DBA User Accounts	27
▼ How to Configure Shared Memory for the Oracle RAC Software	30
Installing the Sun Cluster Support for Oracle RAC Packages	31
▼ How to Install the Sun Cluster Support for Oracle RAC Packages	31
SPARC: Installing the Oracle UDLM	33
▼ SPARC: How to Install the Oracle UDLM	33
2 Configuring Storage for Oracle Files	35
Summary of Configuration Tasks for Storage for Oracle Files	35
Tasks for Configuring the Sun StorEdge QFS Shared File System for Oracle Files	36
Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files	37
Tasks for Configuring VxVM for Oracle Files	38
Tasks for Configuring Hardware RAID Support for Oracle Files	39
Tasks for Configuring ASM for Oracle Files	39

Tasks for Configuring Network Appliance NAS Devices for Oracle Files	40
Tasks for Configuring the Cluster File System for Oracle Files	41
Installing Storage Management Software With Sun Cluster Support for Oracle RAC	41
Using Solaris Volume Manager for Sun Cluster	41
▼ How to Use Solaris Volume Manager for Sun Cluster	41
Using VxVM	42
▼ SPARC: How to Use VxVM	42
Using Hardware RAID Support	43
▼ How to Use Hardware RAID Support	43
Using the Sun StorEdge QFS Shared File System	45
▼ How to Install and Configure the Sun StorEdge QFS Shared File System	47
Using ASM	49
▼ How to Use ASM With Hardware RAID	49
Using the Cluster File System	51
▼ How to Use the Cluster File System	52
Registering and Configuring the RAC Framework Resource Group	53
Tools for Registering and Configuring the RAC Framework Resource Group	53
▼ How to Register and Configure the RAC Framework Resource Group by Using clsetup	53
Creating a Global Device Group for the Oracle RAC Database	57
▼ How to Create a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster for the Oracle RAC Database	58
▼ How to Create a VxVM Shared-Disk Group for the Oracle RAC Database	64
Registering and Configuring Storage Resources for Oracle Files	65
Tools for Registering and Configuring Storage Resources for Oracle Files	66
▼ How to Register and Configure Storage Resources for Oracle Files by Using clsetup	66
3 Enabling Oracle RAC to Run in a Cluster	73
Overview of Tasks for Enabling Oracle RAC to Run in a Cluster	73
Installing Oracle RAC Software	74
Installing Binary Files and Configuration Files on a Shared File System	74
Overriding Networking Defaults for Oracle 10g CRS	74
Installing CRS on a Subset of Sun Cluster Nodes	75
Next Steps	75
Creating Node-Specific Files and Directories for a Shared File System	75
▼ How to Create a Node-Specific Directory for a Shared File System	76

▼ How to Create a Node-Specific File for a Shared File System	79
Verifying the Installation of Oracle RAC	81
▼ How to Verify the Installation of Oracle 10g RAC	81
▼ How to Verify the Installation of Oracle 9i RAC	81
Creating an Oracle Database	81
▼ How to Specify the Location of Data Files on a Shared File System for Oracle 10g	82
▼ How to Specify the Location of Data Files on a Shared File System for Oracle 9i	82
Next Steps	82
Configuring Resources for Oracle RAC Database Instances	82
Tools for Registering and Configuring Resources for Oracle RAC Database Instances	83
▼ How to Enable Sun Cluster and Oracle 10g R2 CRS to Interoperate	84
▼ How to Automate the Startup and Shutdown of Oracle 9i RAC Database Instances	89
Verifying the Installation and Configuration of Sun Cluster Support for Oracle RAC	95
▼ How to Verify the Configuration of the RAC Framework Resource Group	95
▼ How to Verify the Configuration of Storage Resources for Oracle Files	96
▼ How to Verify the Configuration of Resources for Oracle 10g R2 RAC Database Instances	98
▼ How to Verify the Configuration of Resources for Oracle 9i RAC Database Instances	101
▼ How to Verify the Correct Behavior for Shutdown and Booting of the Cluster	102
4 Administering Sun Cluster Support for Oracle RAC	105
Overview of Administration Tasks for Sun Cluster Support for Oracle RAC	105
Automatically Generated Names for Sun Cluster Objects	106
Administering Oracle RAC Databases From Sun Cluster	107
Effects of State Changes to Sun Cluster Resources for Oracle 10g R2 RAC Database Instances	107
Effects of State Changes to Sun Cluster Resources for Oracle 9i RAC Database Instances	109
Tuning Sun Cluster Support for Oracle RAC	110
Guidelines for Setting Timeouts	110
SPARC: Guidelines for Setting the Communications Port Range for the Oracle UDLM	111
▼ How to Modify an Extension Property That Is Tunable Only When a Resource Is Disabled	112
Tuning the Sun Cluster Support for Oracle RAC Fault Monitors	113
Operation of the Fault Monitor for a Scalable Device Group	114
Operation of the Fault Monitor for Scalable File-System Mount Points	114

Operation of the Oracle 9i RAC Server Fault Monitor	115
Operation of the Oracle 9i RAC Listener Fault Monitor	117
Obtaining Core Files for Troubleshooting DBMS Timeouts	117
Customizing the Oracle 9i RAC Server Fault Monitor	118
Defining Custom Behavior for Errors	118
Propagating a Custom Action File to All Nodes in a Cluster	126
Specifying the Custom Action File That a Server Fault Monitor Should Use	127
▼ How to Specify the Custom Action File That a Server Fault Monitor Should Use	127
5 Troubleshooting Sun Cluster Support for Oracle RAC	129
Verifying the Status of Sun Cluster Support for Oracle RAC	129
▼ How to Verify the Status of Sun Cluster Support for Oracle RAC	129
Examples of the Status of Sun Cluster Support for Oracle RAC	130
Sources of Diagnostic Information	139
Common Problems and Their Solutions	140
Node Panic During Initialization of Sun Cluster Support for Oracle RAC	141
Failure of the ucmm daemon to Start	141
▼ How to Recover From a Failure of the UCMM or a Related Component	142
Node Panic Caused by a Timeout	143
Failure of a Node	144
Failure of a SUNW.rac_framework Resource to Start	144
▼ How to Recover From the Timing Out of the START Method	145
Failure of a Resource to Stop	146
6 Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC	147
Overview of Tasks for Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC	147
Modifying Online the Resource for a Scalable Device Group	148
▼ How to Modify Online the Resource for a Scalable Device Group	148
Upgrading Sun Cluster Support for Oracle RAC	148
Upgrading Resources in the RAC Framework Resource Group	149
▼ How to Preserve a Nondefault Value of the reservation_timeout Extension Property	151
▼ How to Remove the Resource of Type SUNW.rac_hwraid	153
Adding Storage Resources for Oracle Files	153

Adding Resources for Interoperation With Oracle 10g R2 CRS	154
Upgrading to Sun Cluster 3.2 Resource Types for Oracle 9i RAC Database Instances	155
▼ How to Take Offline Resource Groups for Oracle RAC Database Components and Restart the Components	155
▼ How to Modify the Configuration of Sun Cluster Resources for Oracle 9i RAC Database Instances	156
Extending an Existing Configuration of Sun Cluster Support for Oracle RAC	157
▼ How to Add Sun Cluster Support for Oracle RAC to Selected Nodes	158
▼ How to Add a Volume Manager Resource to the RAC Framework Resource Group	165
Removing Sun Cluster Support for Oracle RAC	167
▼ How to Remove Sun Cluster Support for Oracle RAC From a Cluster	167
▼ How to Remove Sun Cluster Support for Oracle RAC From Selected Nodes	173
A Sample Configurations of This Data Service	185
Sample Oracle 10g Configurations	186
Sample Oracle 9i Configurations	188
B Preset Actions for DBMS Errors and Logged Alerts	193
C Sun Cluster Support for Oracle RAC Extension Properties	201
SUNW.rac_framework Extension Properties	201
SPARC: SUNW.rac_udlm Extension Properties	202
SUNW.rac_svm Extension Properties	205
SPARC: SUNW.rac_cvm Extension Properties	207
SUNW.scalable_rac_server Extension Properties	210
SUNW.scalable_rac_listener Extension Properties	214
SUNW.scalable_rac_server_proxy Extension Properties	216
SUNW.crs_framework Extension Properties	219
SUNW.ScalDeviceGroup Extension Properties	219
SUNW.ScalMountPoint Extension Properties	221
D Command-Line Alternatives	225
Setting Sun Cluster Support for Oracle RAC Extension Properties	225
Registering and Configuring the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands	226

Overview of the RAC Framework Resource Group	226
▼ How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands	226
Creating Storage Management Resources by Using Sun Cluster Maintenance Commands ...	229
Resources for Scalable Device Groups and Scalable File-System Mount Points	230
Resources for the Sun StorEdge QFS Metadata Server	230
▼ How to Create a Resource for a Scalable Device Group	231
▼ How to Register and Configure Resources for the Sun StorEdge QFS Metadata Server ...	232
▼ How to Create a Resource for a File-System Mount Point	233
Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands	235
▼ How to Create Sun Cluster Resources for Interoperation With Oracle 10g	236
▼ How to Create an Oracle CRS Resource for Interoperation With Sun Cluster	240
Registering and Configuring Sun Cluster Resources for Interoperation With Oracle 9i by Using Sun Cluster Maintenance Commands	242
Oracle 9i RAC Server Resources	242
Oracle 9i Listener Resources	242
Logical Hostname Resources for Oracle 9i Listener Resources	243
▼ How to Register and Configure Sun Cluster Resources for Interoperation With Oracle 9i	244
 Index	 253

Figures

FIGURE A-1	Configuration of Oracle 10g With Solaris Volume Manager for Sun Cluster .	186
FIGURE A-2	Configuration of Oracle 10g With Solaris Volume Manager for Sun Cluster and Sun StorEdge QFS Shared File System	187
FIGURE A-3	Configuration of Oracle 10g With Sun StorEdge QFS Shared File System and Hardware RAID	188
FIGURE A-4	Configuration of Oracle 9i With Solaris Volume Manager for Sun Cluster	189
FIGURE A-5	Configuration of Oracle 9i With Solaris Volume Manager for Sun Cluster and Sun StorEdge QFS Shared File System	190
FIGURE A-6	Configuration of Oracle 9i With Sun StorEdge QFS Shared File System and Hardware RAID	191
FIGURE D-1	Proxy Resources for Configurations With a Volume Manager	236
FIGURE D-2	Proxy Resources for Configurations With a Shared File System	237

Tables

TABLE 1-1	Tasks for Installing Sun Cluster Support for Oracle RAC	19
TABLE 1-2	Storage Management Schemes for Oracle Files	22
TABLE 2-1	Tasks for Configuring the Sun StorEdge QFS Shared File System for Oracle Files	36
TABLE 2-2	Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files	37
TABLE 2-3	Tasks for Configuring VxVM for Oracle Files	38
TABLE 2-4	Tasks for Configuring Hardware RAID Support for Oracle Files	39
TABLE 2-5	Tasks for Configuring ASM for Oracle Files	39
TABLE 2-6	Tasks for Configuring Network Appliance NAS Devices for Oracle Files	40
TABLE 2-7	Tasks for Configuring the Cluster File System for Oracle Files	41
TABLE 3-1	Tasks for Enabling Oracle RAC to Run in a Cluster	73
TABLE 4-1	Administration Tasks for Sun Cluster Support for Oracle RAC	105
TABLE 4-2	Propagation of State Changes Between Sun Cluster Resources and Oracle CRS Resources	108
TABLE 4-3	Comparisons of States for Sun Cluster Resources and Oracle CRS Resources	109
TABLE 4-4	Resource Types for Sun Cluster Support for Oracle RAC Fault Monitors	113
TABLE 6-1	Tasks for Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC	147
TABLE 6-2	Changes to RAC Framework Resource Types	149
TABLE 6-3	Information for Registering the New Version of RAC Framework Resource Types	150
TABLE 6-4	Information for Migrating Existing Instances of RAC Framework Resource Types	150
TABLE B-1	Preset Actions for DBMS Errors	193
TABLE B-2	Preset Actions for Logged Alerts	200

Examples

EXAMPLE 1-1	Creating the DBA Group and the DBA User Accounts	29
EXAMPLE 2-1	Creating a Multi-Owner Disk Set in Solaris Volume Manager for Sun Cluster	62
EXAMPLE 3-1	Creating Node-Specific Directories	77
EXAMPLE 3-2	Creating Node-Specific Files	80
EXAMPLE 4-1	Setting the VxVM Component Reconfiguration Step 4 Timeout	110
EXAMPLE 4-2	Setting the Reservation Step Timeout	111
EXAMPLE 4-3	Setting the Communications Port Number for the Oracle UDLM	111
EXAMPLE 4-4	Changing the Response to a DBMS Error to Restart	122
EXAMPLE 4-5	Ignoring a DBMS Error	122
EXAMPLE 4-6	Changing the Response to a Logged Alert	124
EXAMPLE 4-7	Changing the Maximum Number of Consecutive Timed-Out Probes	125
EXAMPLE 5-1	Status of a Faulty RAC Framework Resource Group	131
EXAMPLE 5-2	Status of a Faulty RAC Database Resource Group	134
EXAMPLE 5-3	Status of an Operational Oracle RAC Configuration	137
EXAMPLE 6-1	Editing an Instance of the SUNW.rac_udlm Resource Type	151
EXAMPLE 6-2	Preserving a Nondefault Value of the reservation_timeout Extension Property	152
EXAMPLE 6-3	Adding Sun Cluster Support for Oracle RAC to Selected Nodes	162
EXAMPLE 6-4	Removing Sun Cluster Support for Oracle RAC From a Cluster	171
EXAMPLE 6-5	Removing Sun Cluster Support for Oracle RAC From Selected Nodes	180
EXAMPLE D-1	Registering and Configuring Sun Cluster Resources for Interoperation With Oracle 9i	249

Preface

Sun Cluster Data Service for Oracle RAC Guide for Solaris OS explains how to install and configure Sun™ Cluster Support for Oracle RAC on both SPARC® based systems and x86 based systems.

Note – This Sun Cluster release supports systems that use the SPARC™ and x86 families of processor architectures: UltraSPARC, SPARC64, and AMD64. In this document, the label x86 refers to systems that use the AMD64 family of processor architectures.

This document is intended for system administrators with extensive knowledge of Sun 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 Solaris™ Operating System (Solaris OS) and expertise with the volume-manager software that is used with Sun Cluster software.

Note – Sun Cluster software runs on two platforms, SPARC and x86. The information in this document pertains to both platforms unless otherwise specified in a special chapter, section, note, bulleted item, figure, table, or example.

Using UNIX Commands

This document contains information about commands that are specific to installing and configuring Sun 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 Solaris Operating System
- 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 <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>machine_name% you have mail.</code>
AaBbCc123	What you type, contrasted with onscreen computer output	<code>machine_name% su</code> Password:
<i>aabbcc123</i>	Placeholder: replace with a real name or value	The command to remove a file is <i>rm filename</i> .
<i>AaBbCc123</i>	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the <i>User's Guide</i> . <i>A cache</i> is a copy that is stored locally. Do <i>not</i> 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 the C shell, Bourne shell, and Korn shell.

TABLE P-2 Shell Prompts

Shell	Prompt
C shell	<code>machine_name%</code>
C shell for superuser	<code>machine_name#</code>
Bourne shell and Korn shell	<code>\$</code>
Bourne shell and Korn shell for superuser	<code>#</code>

Related Documentation

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

Topic	Documentation
Data service administration	<i>Sun Cluster Data Services Planning and Administration Guide for Solaris OS</i> Individual data service guides
Concepts	<i>Sun Cluster Concepts Guide for Solaris OS</i>
Overview	<i>Sun Cluster Overview for Solaris OS</i>
Software installation	<i>Sun Cluster Software Installation Guide for Solaris OS</i>
System administration	<i>Sun Cluster System Administration Guide for Solaris OS</i>
Hardware administration	<i>Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS</i> Individual hardware administration guides
Data service development	<i>Sun Cluster Data Services Developer's Guide for Solaris OS</i>
Error messages	<i>Sun Cluster Error Messages Guide for Solaris OS</i>
Command and function reference	<i>Sun Cluster Reference Manual for Solaris OS</i>

For a complete list of Sun Cluster documentation, see the release notes for your release of Sun 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 – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun 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. Sun 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

The Sun web site provides information about the following additional resources:

- [Documentation](http://www.sun.com/documentation/) (<http://www.sun.com/documentation/>)
- [Support](http://www.sun.com/support/) (<http://www.sun.com/support/>)
- [Training](http://www.sun.com/training/) (<http://www.sun.com/training/>)

Getting Help

If you have problems installing or using Sun 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 Solaris Operating System (for example, Solaris 10)
- The release number of Sun Cluster (for example, Sun Cluster 3.2)

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

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

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

Installing Sun Cluster Support for Oracle RAC

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

- “Overview of the Installation Process for Sun Cluster Support for Oracle RAC” on page 19
- “Preinstallation Considerations” on page 20
- “Preparing the Sun Cluster Nodes” on page 26
- “Installing the Sun Cluster Support for Oracle RAC Packages” on page 31
- “SPARC: Installing the Oracle UDLM” on page 33

Overview of the Installation Process for Sun Cluster 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 Sun Cluster Support for Oracle RAC

Task	Instructions
Plan your installation	“Preinstallation Considerations” on page 20
Prepare the Sun Cluster nodes	“Preparing the Sun Cluster Nodes” on page 26
Install data service packages	“Installing the Sun Cluster Support for Oracle RAC Packages” on page 31
SPARC: Install the UNIX Distributed Lock Manager	“SPARC: Installing the Oracle UDLM” on page 33

Preinstallation Considerations

Oracle Real Application Clusters (RAC) is an application that can run on more than one node concurrently. Sun Cluster Support for Oracle RAC enables you to run Oracle RAC on Sun Cluster nodes and to manage Oracle RAC by using Sun Cluster commands.

Note – In earlier versions of Oracle, this application is referred to as Oracle Parallel Server. In this book, references to Oracle RAC also apply to Oracle Parallel Server unless this book explicitly states otherwise.

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

- **The RAC framework.** These resources enable Oracle RAC to run with Sun Cluster software. The resources also enable reconfiguration parameters to be set by using Sun Cluster commands. You *must* configure resources for the RAC framework. For more information, see [“Registering and Configuring the RAC Framework Resource Group” on page 53](#).
- **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 65](#).
- **Oracle RAC database instances.** The features that these resources provide depend on the version of Oracle that you are using:
 - **Oracle 10g R2.** These resource types enable Oracle Cluster Ready Services (CRS) and Sun Cluster to interoperate. These resource types do *not* provide fault monitoring and automatic fault recovery for Oracle 10g R2RAC. The Oracle CRS software provides this functionality.
 - **Oracle 9i.** 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 for Oracle RAC database instances is optional. For more information, see [“Configuring Resources for Oracle RAC Database Instances” on page 82](#).

Note – If you are using the Solaris 10 OS, install and configure this data service to run only in the global zone. At publication of this document, this data service is not supported in non-global zones. For updated information about supported configurations of this data service, contact your Sun service representative.

Hardware and Software Requirements

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

Sun Cluster Framework Requirements

Sun Cluster Support for Oracle RAC requires a functioning cluster with the initial cluster framework already installed. See *Sun Cluster Software Installation Guide for Solaris OS* 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 Sun Cluster 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 Operating System, Sun Cluster, Oracle, and your volume manager. If you need to install any Sun Cluster Support for Oracle RAC patches, you must apply these patches after you install the data service packages.

Storage Management Requirements for Oracle Files

Sun Cluster 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 StorEdge QFS	Network Appliance NAS Devices	ASM	Cluster File System	Local Disks
RDBMS binary files	-	-	-	+	+	-	+	+
CRS 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	+	+	+	+	+	-	+	-
CRS 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 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. Solaris Volume Manager for Sun Cluster with Oracle Parallel Server is *not* supported.

- 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 StorEdge QFS shared file system with hardware RAID support or with Solaris Volume Manager for Sun Cluster
- Network Appliance network-attached storage (NAS) devices
- Oracle Automatic Storage Management (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
- A shared file system from the following list:
 - The Sun StorEdge QFS shared file system
 - The cluster file system
 - A file system on a Network Appliance NAS device

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:

- The Sun StorEdge QFS shared file system
- The cluster file system

If you use the 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 Network Appliance 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.

SPARC: Processor Architecture Requirements for Oracle Components

Before you decide which architecture to use for the Oracle UDLM 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 Sun Cluster Support for Oracle RAC

You can use Sun Cluster Support for Oracle RAC with Oracle Data Guard. To configure Sun Cluster 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 Sun Cluster 3.2

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 Sun Cluster 3.2, note the points in the subsections that follow before you install Sun Cluster 3.2.

Hostname Restrictions

If you use the Oracle Real Application Clusters Guard option with Sun Cluster 3.2, the following restrictions apply to hostnames that you use in your cluster:

- Hostnames cannot contain special characters.
- You cannot change the hostnames after you install Sun Cluster 3.2.

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

Sun Cluster Command Usage Restrictions

If you use the Oracle Real Application Clusters Guard option with Sun Cluster 3.2, do *not* use Sun Cluster commands to perform the following operations:

- Manipulating the state of resources that Oracle Real Application Clusters Guard installs. Using Sun Cluster commands for this purpose might cause failures.
- Querying the state of the resources that Oracle Real Application Clusters Guard installs. 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 Sun Cluster Nodes

Preparing the Sun Cluster nodes modifies the configuration of the operating system to enable Oracle RAC to run on Sun Cluster nodes. Preparing the Sun 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC to fail during startup.

Before You Begin

Before you prepare the Sun 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC refers to the NIS name service, unavailability of the name service might cause the Sun Cluster Support for Oracle RAC data service to fail.

Bypassing the NIS name service ensures that the Sun Cluster Support for Oracle RAC data service does not refer to the NIS name service when the data service sets the user identifier (ID). The Sun Cluster 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 Sun Cluster 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 27.

▼ How to Create the DBA Group and the DBA User Accounts

In installations of Oracle RAC with Sun Cluster, 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 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 Sun Cluster 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 Sun Cluster 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”](#) on page 30.

▼ How to Configure Shared Memory for the Oracle RAC Software

Note – Perform this task *only* if you are using the Solaris 10 OS. If you are using the Solaris 9 OS, perform the task for configuring kernel parameters in your Oracle RAC documentation.

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 Sun Cluster Support for Oracle RAC”](#) on page 141.

For detailed instructions, see “Shutting Down and Booting a Single Cluster Node” in *Sun Cluster System Administration Guide for Solaris OS*.

Next Steps Go to [“Installing the Sun Cluster Support for Oracle RAC Packages”](#) on page 31.

Installing the Sun Cluster Support for Oracle RAC Packages

If you did not install the Sun Cluster Support for Oracle RAC packages during your initial Sun Cluster installation, perform this procedure to install the packages. To install the packages, use the Sun Java™ Enterprise System Installation Wizard.

▼ How to Install the Sun Cluster Support for Oracle RAC Packages

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

You can run the Sun Java Enterprise System Installation Wizard 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 Sun Java Availability Suite DVD-ROM.

If you intend to run the Sun Java Enterprise System Installation Wizard 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 Sun Java Availability Suite DVD-ROM 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 Sun Java Enterprise System 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 Sun Java Enterprise System Installation Wizard.

```
# ./installer
```

5 When you are prompted, accept the license agreement.

If any Sun Java Enterprise System components are installed, you are prompted to select whether to upgrade the components or install new software.

6 From the list of Sun Cluster agents in 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 Sun Java Enterprise System 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 Sun Java Enterprise System Installation Wizard with the CLI, omit this step.

- 11 Exit the Sun Java Enterprise System Installation Wizard.**

- 12 Unload the Sun Java Availability Suite DVD-ROM 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 33
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 27](#).

▼ SPARC: How to Install the Oracle UDLM

Note – You must install the Oracle UDLM software on the local disk of each node.

- 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.”](#)

Configuring Storage for Oracle Files

This chapter explains how to configure storage for Oracle files.

- “Summary of Configuration Tasks for Storage for Oracle Files” on page 35
- “Installing Storage Management Software With Sun Cluster Support for Oracle RAC” on page 41
- “Registering and Configuring the RAC Framework Resource Group” on page 53
- “Creating a Global Device Group for the Oracle RAC Database” on page 57
- “Registering and Configuring Storage Resources for Oracle Files” on page 65

Summary of Configuration Tasks for Storage for Oracle Files

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

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

The following table summarizes the tasks for configuring the Sun StorEdge QFS shared 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 2-1 Tasks for Configuring the Sun StorEdge QFS Shared File System for Oracle Files

Task	Instructions
Install and configure the Sun StorEdge QFS shared file system	“Using the Sun StorEdge QFS Shared File System” on page 45
Install and configure the other storage management scheme that you are using with the Sun StorEdge QFS shared file system	<p>If you are using Solaris Volume Manager for Sun Cluster, see “Using Solaris Volume Manager for Sun Cluster” on page 41.</p> <p>If you are using hardware RAID support, see “Using Hardware RAID Support” on page 43.</p>
Register and configure the RAC framework resource group	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>
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 58
Register and configure storage resources for Oracle files	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring Storage Resources for Oracle Files” on page 65.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “Creating Storage Management Resources by Using Sun Cluster Maintenance Commands” on page 229.</p>

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

The following table summarizes 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 2-2 Tasks for Configuring Solaris Volume Manager for Sun Cluster for Oracle Files

Task	Instructions
Configure Solaris Volume Manager for Sun Cluster	“Using Solaris Volume Manager for Sun Cluster” on page 41
Register and configure the RAC framework resource group	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>
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 58
Register and configure storage resources for Oracle files	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring Storage Resources for Oracle Files” on page 65.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “Creating Storage Management Resources by Using Sun Cluster Maintenance Commands” on page 229.</p>

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 2-3 Tasks for Configuring VxVM for Oracle Files

Task	Instructions
Install and configure VxVM	“Using VxVM” on page 42
Register and configure the RAC framework resource group	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>
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 64
Register and configure storage resources for Oracle files	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring Storage Resources for Oracle Files” on page 65.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “Creating Storage Management Resources by Using Sun Cluster Maintenance Commands” on page 229.</p>

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.

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

TABLE 2-4 Tasks for Configuring Hardware RAID Support for Oracle Files

Task	Instructions
Install and configure hardware RAID support	“Using Hardware RAID Support” on page 43
Register and configure the RAC framework resource group	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>

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.

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

TABLE 2-5 Tasks for Configuring ASM for Oracle Files

Task	Instructions
Install and configure ASM	“Using ASM” on page 49
Register and configure the RAC framework resource group	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>

Tasks for Configuring Network Appliance NAS Devices for Oracle Files

The following table summarizes the tasks for configuring Network Appliance NAS devices 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 2-6 Tasks for Configuring Network Appliance NAS Devices for Oracle Files

Task	Instructions
Install and configure the Network Appliance NAS devices	<i>Sun Cluster 3.1 - 3.2 With Network-Attached Storage Devices Manual for Solaris OS</i>
Register and configure the RAC framework resource group	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>
Register and configure storage resources for Oracle files	<p>If you are using the <code>clsetup</code> utility for this task, see “Registering and Configuring Storage Resources for Oracle Files” on page 65.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “Creating Storage Management Resources by Using Sun Cluster Maintenance Commands” on page 229.</p>

Tasks for Configuring the 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 2-7 Tasks for Configuring the Cluster File System for Oracle Files

Task	Instructions
Install and configure the cluster file system	“Using the Cluster File System” on page 51
Register and configure the RAC framework resource group	<p>If you are using the <code>csetup</code> utility for this task, see “Registering and Configuring the RAC Framework Resource Group” on page 53.</p> <p>If you are using the Sun Cluster maintenance commands for this task, see “How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226.</p>

Installing Storage Management Software With Sun Cluster 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 21](#).

Note – For information about how to install and configure Network Appliance NAS devices with Sun Cluster Support for Oracle RAC, see *Sun Cluster 3.1 - 3.2 With Network-Attached Storage Devices Manual for Solaris OS*.

Using Solaris Volume Manager for Sun 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 21](#).

▼ How to Use Solaris Volume Manager for Sun Cluster

To use the Solaris Volume Manager for Sun Cluster software with Sun Cluster Support for Oracle RAC, perform the following tasks.

1 Ensure that you are using Solaris 9/04, Solaris 10, or compatible versions.

Solaris Volume Manager for Sun Cluster is installed during the installation of the Solaris Operating System.

2 Configure the Solaris Volume Manager for Sun Cluster software on the cluster nodes.

For more information, see “Configuring Solaris Volume Manager Software” in *Sun Cluster Software Installation Guide for Solaris OS*.

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 “[Registering and Configuring the RAC Framework Resource Group](#)” on page 53.

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 21.

▼ SPARC: How to Use VxVM

To use the VxVM software with Sun Cluster 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 *Sun Cluster Software Installation Guide for Solaris OS* 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 “[Registering and Configuring the RAC Framework Resource Group](#)” on page 53.

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 21](#).

Sun Cluster provides hardware RAID support for several storage devices. For example, you can use Sun StorEdge™ SE9960 disk arrays with hardware RAID support and without volume manager software. To use this combination, configure raw device identities (`/dev/did/rdisk*`) 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

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

1 Create LUNs on the disk arrays.

See the Sun 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/rdisk/c0t2d0
d2              phys-schost-1:/dev/rdisk/c0t3d0
d3              phys-schost-2:/dev/rdisk/c4t4d0
d3              phys-schost-1:/dev/rdisk/c1t5d0
d4              phys-schost-2:/dev/rdisk/c3t5d0
d4              phys-schost-1:/dev/rdisk/c2t5d0
d5              phys-schost-2:/dev/rdisk/c4t4d1
d5              phys-schost-1:/dev/rdisk/c1t5d1
d6              phys-schost-2:/dev/rdisk/c3t5d1
d6              phys-schost-1:/dev/rdisk/c2t5d1
d7              phys-schost-2:/dev/rdisk/c0t2d0
d8              phys-schost-2:/dev/rdisk/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/rdisk/c2t5d0
Replication:                     none
default_fencing:                 global
```

5 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 Gbytes of disk space in slice `s0`.

6 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 `sN` 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/rdisk/d4`. If you choose to use slice `s0` on these devices, specify the raw device `/dev/did/rdisk/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 [“Registering and Configuring the RAC Framework Resource Group” on page 53](#).

Using the Sun StorEdge QFS Shared File System

You must use the Sun StorEdge 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 StorEdge QFS Shared File Systems

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

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

Sun StorEdge 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 Cluster Ready Services (CRS) binary files

Sun StorEdge 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 StorEdge QFS Metadata Server” on page 230](#).

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 CRS voting disk

Optimizing the Performance of the Sun StorEdge 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 StorEdge 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 StorEdge 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 StorEdge QFS software is installed on all nodes where Sun Cluster Support for Oracle RAC is to run.

For information about how to install Sun StorEdge QFS, see *Sun StorEdge QFS Installation and Upgrade Guide, Version 4, Update 5*.

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

For information about how to create a Sun StorEdge QFS file system, see *Sun StorEdge QFS Configuration and Administration Guide, Version 4, Update 5*.

For each Sun StorEdge 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
sync_meta=1
mh_write
qwrite
forcedirectio
nstreams=1024
rdlease=300    Set this value for optimum performance.
wrlease=300    Set this value for optimum performance.
aplease=300    Set this value for optimum performance.
```

<i>fs-name</i>	Specifies the name that uniquely identifies the file system.
<i>width</i>	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). <i>width</i> 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 StorEdge 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 Change the ownership of each file system that you are using for Oracle files 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 27.

```
# 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.

5 Grant to the owner of each file system whose ownership you changed in [Step 4](#) read access and write access to the file system.

```
# 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 [“Registering and Configuring the RAC Framework Resource Group”](#) on page 53.

Using ASM

For information about the types of Oracle files that you can store by using ASM, see “[Storage Management Requirements for Oracle Files](#)” on page 21.

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

- **Hardware RAID.** For more information, see “[How to Use ASM With Hardware RAID](#)” on page 49.
- **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 58.
- **VxVM.** For more information, see “[How to Create a VxVM Shared-Disk Group for the Oracle RAC Database](#)” on page 64.

▼ How to Use ASM With Hardware RAID

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- 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/rdisk/c3t216000C0FF084E77d0
d5                  phys-schost-1:/dev/rdisk/c5t216000C0FF084E77d0
d5                  phys-schost-2:/dev/rdisk/c4t216000C0FF084E77d0
d5                  phys-schost-4:/dev/rdisk/c2t216000C0FF084E77d0
d6                  phys-schost-3:/dev/rdisk/c4t216000C0FF284E44d0
d6                  phys-schost-1:/dev/rdisk/c6t216000C0FF284E44d0
d6                  phys-schost-2:/dev/rdisk/c5t216000C0FF284E44d0
d6                  phys-schost-4:/dev/rdisk/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 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/rdisk/d5
Full Device Path:                phys-schost-1:/dev/rdisk/c5t216000C0FF084E77d0
Replication:                    none
default_fencing:                global

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

4 Create or modify a slice on each DID device to contain the disk-space allocation for the 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 ASM disk group, you might choose to allocate 100 Gbytes of disk space in slice `s0`.

5 Change the ownership and permissions of the raw devices that you are using for ASM to allow access by ASM to these devices.

To specify the raw device, append `sN` to the DID device name that you obtained in [Step 3](#), where `N` is the slice number.

For example, the `cldevice` output in [Step 3](#) identifies that the raw DIDs that correspond to the disk are `/dev/did/rdisk/d5` and `/dev/did/rdisk/d6`. If you choose to use slice `s0` on these devices, specify the raw devices `/dev/did/rdisk/d5s0` and `/dev/did/rdisk/d6s0`.

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

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

For example, to use the `/dev/did/` path for the ASM disk group, add the value `/dev/did/rdisk/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/rdisk/*'
```

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 [“Registering and Configuring the RAC Framework Resource Group” on page 53](#).

Using the 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 *Sun Cluster Software Installation Guide for Solaris OS*
- “Creating Cluster File Systems” in *Sun Cluster Software Installation Guide for Solaris OS*

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

Types of Oracle Files That You Can Store on the 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 CRS 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 CRS 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 the 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 *Sun Cluster Concepts Guide for Solaris OS*.

▼ How to Use the Cluster File System

1 Create and mount the cluster file system.

See “Creating Cluster File Systems” in *Sun Cluster Software Installation Guide for Solaris OS* 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 CRS 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 CRS 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 [“Registering and Configuring the RAC Framework Resource Group” on page 53](#).

Registering and Configuring the RAC Framework Resource Group

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

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

Tools for Registering and Configuring the RAC Framework Resource Group

Sun Cluster provides the following tools for registering and configuring the RAC framework resource group:

- **The `clsetup(1CL)` utility.** For more information, see [“How to Register and Configure the RAC Framework Resource Group by Using `clsetup`” on page 53](#).
- **Sun Cluster Manager.** For more information, see the Sun Cluster Manager online help.
- **Sun Cluster maintenance commands.** For more information, see [Appendix D, “Command-Line Alternatives”](#).

The `clsetup` utility and Sun 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.

▼ 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 Sun Cluster Support for Oracle RAC. Perform this procedure from one node only.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Note – The following instructions explain how to perform this operation by using the `clsetup` utility.

Before You Begin Ensure that the following prerequisites are met:

- All preinstallation tasks for Oracle RAC are completed.
- The Sun Cluster nodes are prepared.
- The data services packages are installed.
- 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 following information:

- The names of the nodes where you require Sun Cluster Support for Oracle RAC to run
- 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 number that corresponds to the option for data services and press Return.

The Data Services menu is displayed.

4 Type the number that corresponds to the option for configuring Sun Cluster Support for Oracle RAC and press Return.

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

5 Press Return to continue.

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

6 Type the number that corresponds to the option for performing the initial configuration of Sun Cluster Support for Oracle RAC and press Return.

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

- 7 Type the number that corresponds to the option for the RAC framework resource group and press Return.**

The `clsetup` utility displays the list of prerequisites for performing this task.
- 8 Verify that the prerequisites are met, and press Return.**

The `clsetup` utility displays a list of the cluster nodes on which the Sun Cluster Support for Oracle RAC packages are installed.
- 9 Select the nodes where you require Sun Cluster 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 numbers that correspond to the nodes 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 numbers that correspond to the nodes 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.
- 10 To confirm your selection of nodes, type `d` and press Return.**

The `clsetup` utility displays a list of storage management schemes for Oracle files.
- 11 Type the numbers that correspond to the storage management schemes that you are using for Oracle files and press Return.**
- 12 To confirm your selection of storage management schemes, type `d` and press Return.**

The `clsetup` utility displays the names of the Sun Cluster objects that the utility will create.
- 13 If you require a different name for any Sun Cluster objects, change each name as follows.**
 - a. Type the number that corresponds to the name that you are changing 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 Sun Cluster objects that the utility will create.

14 To confirm your selection of Sun Cluster object names, type d and press Return.

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

15 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.

16 Press Return to continue.

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

17 (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.

18 Determine if 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.

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

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

```
# clresourcegroup online 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: <code>SUN.rac_framework</code>	None.	RAC framework resource.
Resource name: <code>rac-framework-rs</code>		
Resource group: <code>rac-framework-rg</code>		

Resource Name, Resource Type, and Resource Group	Dependencies	Description
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.
Resource type: SUNW.rac_svm Resource name: rac-svm-rs Resource group: rac-framework-rg	Strong dependency on the RAC framework resource.	Solaris Volume Manager for Sun Cluster resource. Created only if Solaris Volume Manager for Sun Cluster was selected.
Resource type: SUNW.rac_cvm Resource name: rac-cvm-rs Resource group: rac-framework-rg	Strong dependency on the RAC 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 58
VxVM with the cluster feature	“How to Create a VxVM Shared-Disk Group for the Oracle RAC Database” on page 64
None	“Registering and Configuring Storage Resources for Oracle Files” on page 65

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](#).
- If you are using VxVM, create a [VxVM shared-disk group](#).

▼ 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 StorEdge QFS shared file system, or 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*.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Before You Begin Note the following points.

- Ensure that the required Sun Cluster Support for Oracle RAC software packages are installed on each node. For more information, see [“Installing the Sun Cluster Support for Oracle RAC Packages” on page 31](#).
- Unless you are using the Sun StorEdge QFS shared file system, do not create any file systems in the multi-owner disk set. In configurations without the Sun StorEdge 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 <i>setname</i> | 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 <i>nodelist</i> | Specifies a space-separated list of nodes that are to be added to the disk set. The Sun Cluster Support for Oracle RAC software packages <i>must</i> 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/dN`, 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 StorEdge 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 StorEdge 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 `dV`, 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 `dV`, 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.

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 StorEdge 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](#) 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 27.

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

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/dV`
- For raw devices: `/dev/md/setname/rdisk/dV`

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 StorEdge QFS shared file system only) Grant to the owner of each volume whose ownership you changed in [Step 8](#) read access and write access to the volume.

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

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

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

For example, to use the `/dev/md/setname/rdisk/d` path for the ASM disk group, add the value `/dev/md/*/rdisk/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/*/rdisk/d*'
```

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

For more information, see your Oracle documentation.

Example 2-1 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 StorEdge QFS shared file system. This example does not show the creation of the Sun StorEdge 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 oradg Multi-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 65.

▼ 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 ASM to use.

Before You Begin Note the following points.

- Ensure that the required Sun Cluster Support for Oracle RAC software packages are installed on each node. For more information, see [“Installing the Sun Cluster Support for Oracle RAC Packages”](#) on page 31.
- 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 ASM, specify the raw devices that you are using for the ASM disk group.**
To specify the devices, modify the `ASM_DISKSTRING` ASM instance-initialization parameter.
For example, to use the `/dev/md/setname/rdisk/d` path for the ASM disk group, add the value `/dev/md/*/rdisk/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/*/rdisk/d*'`
If you are using mirrored devices, specify external redundancy in the ASM configuration.
For more information, see your Oracle documentation.

Next Steps Go to [“Registering and Configuring Storage Resources for Oracle Files”](#) on page 65.

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 storage resources for the following types of global device groups:

- Solaris Volume Manager for Sun Cluster multi-owner disk sets
- VxVM shared-disk groups

Configure storage resources for the following types of shared file systems:

- A Sun StorEdge QFS shared file system
- A file system on a Network Appliance NAS device

Tools for Registering and Configuring Storage Resources for Oracle Files

Sun Cluster provides the following tools for registering and configuring storage resources for Oracle files:

- **The `clsetup(1CL)` utility.** For more information, see [“How to Register and Configure Storage Resources for Oracle Files by Using `clsetup`”](#) on page 66.
- **Sun Cluster Manager.** For more information, see the Sun Cluster Manager online help.
- **Sun Cluster maintenance commands.** For more information, see [“Creating Storage Management Resources by Using Sun Cluster Maintenance Commands”](#) on page 229.

The `clsetup` utility and Sun 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`

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 53.
- Required volumes, global device groups, and file systems are created. For more information, see the following sections:
 - [“Installing Storage Management Software With Sun Cluster Support for Oracle RAC”](#) on page 41
 - [“Creating a Global Device Group for the Oracle RAC Database”](#) on page 57
- 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 Type the number that corresponds to the option for data services and press Return.

The Data Services menu is displayed.

4 Type the number that corresponds to the option for configuring Sun Cluster Support for Oracle RAC and press Return.

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

5 Press Return to continue.

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

6 Type the number that corresponds to the option for performing the initial configuration of Sun Cluster Support for Oracle RAC and press Return.

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

7 Type the number that corresponds to the option for storage resources for Oracle files and press Return.

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

8 Verify that the prerequisites are met, and press Return.

The response of the `clsetup` utility depends on how the RAC framework resource group was configured.

- **By using the `clsetup` wizard or the Sun Cluster Manager wizard.** 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.
- **By using the `scsetup` utility or Sun Cluster maintenance commands.** The `clsetup` utility displays a list of storage management schemes for Oracle files.

- 9 If you are prompted to select storage management schemes for Oracle files, select the schemes.**
If you are prompted for resources for scalable device groups, omit this step.

 - a. Type the numbers that correspond to the storage management schemes that you are using for Oracle files and press Return.**
 - b. To confirm your selection of storage management schemes, type `d` and press Return.**
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.
- 10 If no suitable resources exist, or if no resource exists for a device group that you are using, add a resource to the list.**
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. Type `c` and press Return.**
The `clsetup` utility displays a list of the scalable device groups that are configured on the cluster.
 - b. Type a comma-separated or space-separated list of numbers that correspond to the device groups that you are using for Oracle files and press Return.**
 - c. 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.
- 11 If a suitable existing resource that you intend to use is not listed, type `r` to refresh the list.**
- 12 When the list contains resources for all the device groups that you are using, type the numbers that correspond to the resources that you require.**
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.
- 13 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.
- 14 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. Type `c` and 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.

15 If a suitable existing resource that you intend to use is not listed, type `r` to refresh the list.

16 When the list contains resources for all the file-system mount points that you are using, type the numbers that correspond to the resources that you require.

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 Sun Cluster objects that the utility will create or add to your configuration.

18 If you need to modify a Sun Cluster object that the utility will create, modify the object as follows:

a. Type the number that corresponds to the Sun 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 number that corresponds to 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.

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

19 When you have modified all the Sun Cluster objects that you need to change, type d.

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 Sun Cluster 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 if 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
<p>Resource type: SUNW.ScalDeviceGroup</p> <p>Resource name: <code>scaldg-name-rs</code>, where dg-name is the name of the device group that the resource represents</p> <p>Resource group: <code>scaldg-rg</code></p>	<p>Strong dependency on the resource in the RAC 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.</p>	<p>Scalable device-group resource. One resource is created for each scalable device group that you are using for Oracle files.</p>
<p>Resource type: SUNW.qfs</p> <p>Resource name: <code>qfs-mp-dir-rs</code>, where mp-dir is the mount point of the file system, with / replaced by –</p> <p>Resource group: <code>qfsmds-rg</code></p>	<p>Strong dependency on the scalable device-group resource, if any.</p> <p>If you are using Sun StorEdge QFS without a volume manager, this resource does not depend on any other resources.</p>	<p>Resource for the Sun StorEdge QFS metadata server. One resource is created for each Sun StorEdge QFS shared file system that you are using for Oracle files.</p>
<p>Resource type: SUNW.ScalMountPoint</p> <p>Resource name: <code>scal-mp-dir-rs</code>, where mp-dir is the mount point of the file system, with / replaced by –</p> <p>Resource group: <code>scalmnt-rg</code></p>	<p>Strong dependency on the resource for the Sun StorEdge QFS metadata server, if any.</p> <p>Offline-restart dependency on the scalable device-group resource, if any.</p> <p>If you are using a file system on a Network Appliance NAS device without a volume manager, this resource does not depend on any other resources.</p>	<p>Scalable file-system mount point resource. One resource is created for each shared file system that you are using for Oracle files.</p>

Next Steps Go to [Chapter 3, “Enabling Oracle RAC to Run in a Cluster.”](#)

Enabling Oracle RAC to Run in a Cluster

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

- “Overview of Tasks for Enabling Oracle RAC to Run in a Cluster” on page 73
- “Installing Oracle RAC Software” on page 74
- “Creating Node-Specific Files and Directories for a Shared File System” on page 75
- “Verifying the Installation of Oracle RAC” on page 81
- “Creating an Oracle Database” on page 81
- “Configuring Resources for Oracle RAC Database Instances” on page 82
- “Verifying the Installation and Configuration of Sun Cluster Support for Oracle RAC” on page 95

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

Table 3-1 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 3-1 Tasks for Enabling Oracle RAC to Run in a Cluster

Task	Instructions
Install the Oracle RAC software	“Installing Oracle RAC Software” on page 74
Create node-specific files and directories that the Sun Cluster Support for Oracle RAC software requires	“Creating Node-Specific Files and Directories for a Shared File System” on page 75
Verify the installation of the Oracle RAC software	“Verifying the Installation of Oracle RAC” on page 81
Create your Oracle database	“Creating an Oracle Database” on page 81

TABLE 3-1 Tasks for Enabling Oracle RAC to Run in a Cluster (Continued)

Task	Instructions
(Not required for Oracle 10g R1) Configure resources for Oracle RAC database instances	“Configuring Resources for Oracle RAC Database Instances” on page 82
Verify the Sun Cluster Support for Oracle RAC installation and configuration	“Verifying the Installation and Configuration of Sun Cluster Support for Oracle RAC” on page 95

Installing Oracle RAC Software

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 StorEdge QFS shared file system
- The cluster file system
- A file system on a Network Appliance 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 10g CRS

By default, the network interfaces panel of the Oracle 10g Universal Installer shows all interfaces as private. If you are installing Oracle 10g RAC for use with Sun Cluster, 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 CRS on a Subset of Sun Cluster Nodes

By default, the Oracle installer installs CRS on all nodes in a cluster. Instructions for installing CRS on a subset of Sun Cluster nodes are available at the [Oracle MetaLink web site](http://metalink.oracle.com/) (<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 75
Local disks of each node	“Verifying the Installation of Oracle RAC” on page 81

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.

▼ 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/srvn/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 3-1 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/srvn/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
```

<code>-s</code>	Specifies that the link is a symbolic link.
<code>local-file</code>	Specifies that the file that you copied in Step 2 is the source of the link.
<code>global-file</code>	Specifies that the global version of the file that you removed in Step a is the target of the link.

Example 3-2 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 81.

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 RAC database instances can be started and stopped automatically.

▼ How to Verify the Installation of Oracle 10g 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 9i 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 81.

Creating an Oracle Database

Perform this task to configure and create the initial Oracle database in a Sun 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 10g

- 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.

▼ How to Specify the Location of Data Files on a Shared File System for Oracle 9i

- 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 82.](#)

Configuring Resources for Oracle RAC Database Instances

Resources for Oracle RAC database instances enable the database instances to be administered from Sun Cluster.

If you are using Oracle 10g R2 RAC, these resources also enable Sun Cluster and Oracle Cluster Ready Services (CRS) to interoperate.

The software architectures of Oracle 9i, Oracle 10g R1, and Oracle 10g R2 are different. As a result of these differences, the resources for Oracle RAC database instances that Sun Cluster requires depend on the version of Oracle that you are using.

Note – If you are using Oracle 10g R1, omit this task. In Oracle 10g R1, Oracle CRS and Sun Cluster cannot interoperate. Instead, Oracle CRS start and shut down Oracle RAC database instances.

Note – You can also use the Oracle 10g R2 RAC Plug-in for Sun N1 Service Provisioning System to deploy Oracle 10g R2 RAC software in a Sun Cluster configuration. Follow instructions in the documentation at

<http://wikis.sun.com/display/SunCluster/Oracle+RAC+10g+R2+Plug-in>.

Tools for Registering and Configuring Resources for Oracle RAC Database Instances

Sun Cluster provides the following tools for registering and configuring resources for Oracle RAC database instances:

- **The `clsetup(1CL)` utility.** For more information, see the following sections:
 - “How to Enable Sun Cluster and Oracle 10g R2 CRS to Interoperate” on page 84
 - “How to Automate the Startup and Shutdown of Oracle 9i RAC Database Instances” on page 89
- **Sun Cluster Manager.** For more information, see the Sun Cluster Manager online help.
- **Sun Cluster maintenance commands.** For more information, see the following sections:
 - “Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands” on page 235
 - “Registering and Configuring Sun Cluster Resources for Interoperation With Oracle 9i by Using Sun Cluster Maintenance Commands” on page 242

The `clsetup` utility and Sun 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 Sun Cluster and Oracle 10g R2 CRS to Interoperate

When you perform this task, the `clsetup` utility creates the following resources for interoperation with Oracle 10g R2:

- A Sun Cluster resource to act as a proxy for the Oracle RAC database
- A Sun Cluster resource to represent the Oracle CRS framework
- Oracle CRS resources to represent scalable device groups
- Oracle CRS 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 CRS 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 Sun 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 Type the number that corresponds to the option for data services and press Return.

The Data Services menu is displayed.

4 Type the number that corresponds to the option for configuring Sun Cluster Support for Oracle RAC and press Return.

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

5 Press Return to continue.

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

6 Type the number that corresponds to the option for performing the initial configuration of Sun Cluster Support for Oracle RAC and press Return.

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

7 Type the number that corresponds to the option for resources for Oracle RAC database instances and press Return.

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

8 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.

9 Type the number that corresponds to the option for Oracle 10g R2 and press Return.

The `clsetup` utility displays a list of Oracle CRS home directories that exist on the cluster.

10 Specify the Oracle CRS home directory for your installation of the Oracle CRS 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 names of Oracle RAC databases that are configured on the cluster.

- **If the directory is not listed, specify the directory explicitly.**

a. Type e and press Return.

The `clsetup` utility prompts you for the Oracle CRS home directory.

b. Type the full path to the Oracle CRS home directory and press Return.

The `clsetup` utility displays a list of names of Oracle RAC databases that are configured on the cluster.

11 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 RAC database 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.

12 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.

13 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 Sun 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 65](#).

- **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 Sun 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 65](#).

- 14 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 Sun Cluster objects that the utility will create.

- 15 If you require a different name for any Sun Cluster objects, change each name as follows.**

- a. **Type the number that corresponds to the name that you are changing 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 Sun Cluster objects that the utility will create.

16 To confirm your selection of Sun Cluster object names, press Return.

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

19 (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 Sun Cluster resources that the `clsetup` utility creates when you complete this task.

Resource Type, Resource Name, and Resource Group	Dependencies	Description
Resource type: <code>SUNW.crs_framework</code> Resource name: <code>crs_framework-rs</code> Resource group: <code>rac-framework-rg</code>	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 CRS framework resource.
Resource type: <code>SUNW.scalable_rac_server_proxy</code> Resource name: <code>rac_server_proxy-rs</code> Resource group: <code>rac_server_proxy-rg</code>	Strong dependency on the RAC framework resource. Offline-restart dependency on the Oracle CRS framework resource.	Proxy resource for the Oracle RAC database server.

The `clsetup` utility also creates an Oracle CRS resource for each Sun Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend.

The name of each Oracle CRS 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 CRS resource is to run.
- *sc-rs* specifies the name of the Sun Cluster resource that the Oracle CRS resource represents. This resource is created when the task in [“Registering and Configuring Storage Resources for Oracle Files” on page 65](#) is performed.

The Oracle CRS resource for the database instance on *node* depends on the Oracle CRS resource that the `clsetup` utility creates for that node. Oracle CRS resources for database instances are created during the installation and configuration of the Oracle RAC

The name of the Oracle CRS 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 CRS resource represents.
- *sid* specifies the Oracle SID of the database instance that the Oracle CRS resource represents.

Next Steps Go to [“Verifying the Installation and Configuration of Sun Cluster Support for Oracle RAC” on page 95](#).

▼ How to Automate the Startup and Shutdown of Oracle 9i 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 Sun 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 Type the number that corresponds to the option for data services and press Return.

The Data Services menu is displayed.

4 Type the number that corresponds to the option for configuring Sun Cluster Support for Oracle RAC and press Return.

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

5 Press Return to continue.

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

6 Type the number that corresponds to the option for performing the initial configuration of Sun Cluster Support for Oracle RAC and press Return.

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

7 Type the number that corresponds to the option for resources for Oracle RAC database instances and press Return.

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

8 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.

9 Type the number that corresponds to the option for Oracle 9i and press Return.

- 10 Type the number that corresponds to the option to configure both the listener and the server.**
The `clsetup` utility displays a list of Oracle home directories that exist on the cluster.

- 11 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.

- 12 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 Sun 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 Sun Cluster resources that the utility will create.

13 If you require a different value for any Sun Cluster resource properties, change each property as follows.

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 number that corresponds to the property that you are changing 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 Sun Cluster resource properties, press Return.

The response of the `clsetup` utility depends on whether you have specified the Sun Cluster resource properties for all nodes where Oracle RAC is to run.

- If you have *not* specified Sun 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 Sun Cluster resource properties for all nodes, the `clsetup` utility displays a list of logical hostname resources that are configured.

14 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 number that corresponds to the resource that you are selecting 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 Sun 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 65.

▪ **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 number that corresponds to the resource that you are creating 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 Sun 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 65.
- 15 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 Sun Cluster objects that the utility will create.
- 16 If you require a different name for any Sun Cluster objects, change each name as follows.**
- a. **Type the number that corresponds to the name that you are changing 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 Sun Cluster objects that the utility will create.
- 17 To confirm your selection of Sun Cluster object names, press Return.**
- The `clsetup` utility displays information about the Sun Cluster configuration that the utility will create.
- 18 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.
- 19 Press Return to continue.**
- The `clsetup` utility returns you to the list of options for configuring Sun Cluster Support for Oracle RAC.
- 20 (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
<p>Resource type: <code>SUNW.LogicalHostname</code></p> <p>Resource name: <i>lh-name</i>, where <i>lh-name</i> is the logical hostname that you specified when you created the resource</p> <p>Resource group: <code>rac-lhN-rg</code>, where <i>N</i> 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.</p>	None.	Logical hostname resource. One logical hostname resource is created for each node where Oracle RAC is to run.
<p>Resource type: <code>SUNW.scalable_rac_listener</code></p> <p>Resource name: <code>rac-listener-rs</code></p> <p>Resource group: <i>ora-sid-rg</i>, where <i>ora-sid</i> is the SID on the primary node without any numbers in the SID</p>	Strong dependency on each logical hostname resource.	Scalable RAC listener resource.
<p>Resource type: <code>SUNW.scalable_rac_server</code></p> <p>Resource name: <i>ora-sid-rs</i>, where <i>ora-sid</i> is the SID on the primary node without any numbers in the SID</p> <p>Resource group: <i>ora-sid-rg</i>, where <i>ora-sid</i> is the SID on the primary node without any numbers in the SID</p>	<p>Strong dependency on the RAC framework resource.</p> <p>Weak dependency on the scalable RAC listener resource.</p> <p>Offline-restart dependencies on all scalable device-group resources for Oracle files, if any.</p> <p>Offline-restart dependencies on all scalable file-system mount-point resources for Oracle files.</p> <p>If you are not using a shared file system or a volume manager, this resource has no offline-restart dependencies on any other resources.</p>	Scalable RAC server resource.

Next Steps Go to [“Verifying the Installation and Configuration of Sun Cluster Support for Oracle RAC”](#) on page 95.

Verifying the Installation and Configuration of Sun Cluster Support for Oracle RAC

After you install, register, and configure Sun Cluster Support for Oracle RAC, verify the installation and configuration. Verifying the Sun Cluster 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 *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Verifying the installation and configuration of Sun Cluster Support for Oracle RAC involves the following tasks:

1. [Verifying the configuration of the RAC Framework resource group](#)
2. [Verifying the configuration of storage resources for Oracle files](#)
3. (Oracle 10g R2 and Oracle 9i only) [Verifying the configuration of resources for Oracle RAC database instances](#)

Note – If you are using Oracle 10g R1, 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 10g R2.** Perform the task in “[How to Verify the Configuration of Resources for Oracle 10g R2 RAC Database Instances](#)” on page 98.
 - **Oracle 9i.** Perform the task in “[How to Verify the Configuration of Resources for Oracle 9i RAC Database Instances](#)” on page 101.
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 53 is performed.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- 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-fmwk-rg` Specifies the name of the RAC framework resource group.

- 4 **Verify that all resource groups for Oracle RAC are online.**

```
# clresourcegroup status
```

▼ How to Verify the Configuration of Storage Resources for Oracle Files

Storage resources for Oracle files are created when the task in “[Registering and Configuring Storage Resources for Oracle Files](#)” on page 65 is performed.

Perform this task only if your configuration of storage management schemes requires storage resources for Oracle files.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- 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 10g R2, the status of the RAC framework resource group is pending online blocked.
 - If you are using Oracle 10g R1 or Oracle 9i, the status of the RAC framework resource group remains online.

```
# clresourcegroup status
```

5 (Oracle 10g R2 only) Verify that the Oracle CRS are offline.

To verify that the Oracle CRS are offline, consult the system messages file to find messages to indicate that CRS 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 10g R2 only) On each node, restart the Oracle CRS.

```
# /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 10g R2 only) On each node, verify that the Oracle CRS are online.

Use the Oracle command `crstat` for this purpose.

```
# crs-home/bin/crs_stat
```

crs-home Specifies the Oracle CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

Note – The startup of Oracle CRS on all nodes might require several minutes to complete. If you perform this step before the Oracle CRS 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 CRS 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 10g R2 RAC Database Instances

Resources for Oracle 10g R2 RAC database instances are created when the tasks in one of the following sections are performed:

- [“How to Enable Sun Cluster and Oracle 10g R2 CRS to Interoperate” on page 84](#)
- [“Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands” on page 235](#)

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands

are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 CRS resources to represent Sun 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 CRS resources to represent Sun 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 CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

b. For each Oracle CRS resource that represents an Oracle RAC database instance, confirm that the required dependencies are set.

Each Oracle CRS 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 CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

crs-resource Specifies the name of an Oracle CRS 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 CRS framework resource.

```
# clresource disable -t SUNW.crs_framework +
```

This step might require several minutes to complete.

5 Verify that the Oracle CRS are offline.**a. Attempt to obtain the status of Oracle CRS 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 CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

b. Consult the system messages file to find messages to indicate that CRS are being shut down.**6 Verify that the following resources are offline on all nodes:**

- The CRS framework resource
- The RAC database resource

```
# clresource status -t SUNW.crs_framework,SUNW.scalable_rac_server_proxy +
```

7 On each node, restart the Oracle CRS.

```
# /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 CRS are online.

Use the Oracle command `crstat` for this purpose.

```
# crs-home/bin/crs_stat
```

crs-home Specifies the Oracle CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

Note – The startup of Oracle CRS on all nodes might require several minutes to complete. If you perform this step before the Oracle CRS 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 CRS is online on all nodes.

9 Enable the CRS 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 CRS framework resource

- The RAC database resource

```
# clresource status -t SUNW.crs_framework,SUNW.scalable_rac_server_proxy +
```

Note – Enabling the CRS 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 9i RAC Database Instances

Resources for Oracle 9i 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 9i RAC Database Instances” on page 89](#)
- [“Registering and Configuring Sun Cluster Resources for Interoperation With Oracle 9i by Using Sun Cluster Maintenance Commands” on page 242](#)

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- 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 Sun Cluster Support for Oracle RAC is correct, Sun 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 Sun Cluster Support for Oracle RAC on a production cluster where other data services are running, omit this task.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

1 Shut down the cluster.

For instructions for performing this task, see “How to Shut Down a Cluster” in *Sun Cluster System Administration Guide for Solaris OS*.

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 *Sun Cluster System Administration Guide for Solaris OS*.

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.

Administering Sun Cluster Support for Oracle RAC

This chapter explains how to administer Sun Cluster Support for Oracle RAC on your Sun Cluster nodes.

- “Overview of Administration Tasks for Sun Cluster Support for Oracle RAC” on page 105
- “Automatically Generated Names for Sun Cluster Objects” on page 106
- “Administering Oracle RAC Databases From Sun Cluster” on page 107
- “Tuning Sun Cluster Support for Oracle RAC” on page 110
- “Tuning the Sun Cluster Support for Oracle RAC Fault Monitors” on page 113
- “Customizing the Oracle 9i RAC Server Fault Monitor” on page 118

Overview of Administration Tasks for Sun Cluster Support for Oracle RAC

Table 4–1 summarizes the administration tasks for Sun Cluster Support for Oracle RAC.

Perform these tasks whenever they are required.

TABLE 4-1 Administration Tasks for Sun Cluster Support for Oracle RAC

Task	Instructions
Administer Oracle RAC databases from Sun Cluster	“Administering Oracle RAC Databases From Sun Cluster” on page 107
Tune Sun Cluster Support for Oracle RAC extension properties	“Tuning Sun Cluster Support for Oracle RAC” on page 110
Tune Sun Cluster Support for Oracle RAC fault monitors	“Tuning the Sun Cluster Support for Oracle RAC Fault Monitors” on page 113

TABLE 4-1 Administration Tasks for Sun Cluster Support for Oracle RAC (Continued)

Task	Instructions
Customize the Oracle 9i RAC server fault monitor	“Customizing the Oracle 9i RAC Server Fault Monitor” on page 118
Troubleshoot Sun Cluster Support for Oracle RAC	Chapter 5, “Troubleshooting Sun Cluster Support for Oracle RAC,”

Examples in this chapter provide the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster 3.1 - 3.2 Hardware Administration Manual for Solaris OS*.

Automatically Generated Names for Sun Cluster Objects

When the `clsetup` utility or Sun 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 Sun Cluster Manager, see the following table for these names.

Resource Type	Resource Name
SUNW.rac_svm	rac-svm-rs
SUNW.rac_cvm	rac-cvm-rs
SUNW.rac_udlm	rac-udlm-rs
SUNW.rac_framework	rac-framework-rs
SUNW.scalable_rac_server	<i>ora-sid</i> -rs, where <i>ora-sid</i> 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	<i>scal-dg-name</i> -rs, where <i>dg-name</i> is the name of the device group that the resource represents
SUNW.ScalMountPoint	<i>scal-mp-dir</i> -rs, where <i>mp-dir</i> is the mount point of the file system, with / replaced by –
SUNW.qfs	<i>qfs-mp-dir</i> -rs, where <i>mp-dir</i> is the mount point of the file system, with / replaced by –

Resource Type	Resource Name
SUNW.LogicalHostname	<i>lh-name</i> , where <i>lh-name</i> is the logical hostname that you specified when you created the resource

Administering Oracle RAC Databases From Sun Cluster

Administering Oracle RAC databases from Sun Cluster involves using Sun Cluster administration tools to modify the states of Sun 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 82.

The software architectures of Oracle 9i, Oracle 10g R1, and Oracle 10g R2 are different. As a result of these differences, the resources for Oracle RAC database instances that Sun Cluster requires depend on the version of Oracle that you are using. Consequently, the administration of Oracle RAC databases from Sun Cluster also depends on the version of Oracle that you are using.

Note – If you are using Oracle 10g R1, you cannot administer Oracle RAC databases from Sun Cluster. Instead, use Oracle CRS utilities to start and shut down Oracle RAC database instances.

The effects of changes to the states of Sun Cluster resources on Oracle database components are explained in the subsections that follow.

Effects of State Changes to Sun Cluster Resources for Oracle 10g R2 RAC Database Instances

In Oracle 10g, the Oracle CRS manage the startup and shutdown of Oracle database instances, listeners, and other components that are configured in the CRS. Oracle CRS are a mandatory component of Oracle 10g. CRS also monitor components that are started by CRS and, if failures are detected, perform actions to recover from failures.

Because Oracle CRS manage the startup and shutdown of Oracle database components, these components cannot be stopped and started exclusively under the control of the Sun Cluster RGM. Instead, Oracle CRS and the Sun Cluster RGM interoperate so that when Oracle RAC database instances are started and stopped by Oracle CRS, the state of the database instances is propagated to Sun Cluster resources.

TABLE 4-2 Propagation of State Changes Between Sun Cluster Resources and Oracle CRS Resources

Trigger	Initial State		Resulting State	
	Sun Cluster Resource	Oracle CRS Resource	Sun Cluster Resource	Oracle CRS Resource
Sun Cluster command to take offline a resource	Enabled and online	Enabled and online	Enabled and offline	Enabled and offline
Oracle CRS command to stop a resource	Enabled and online	Enabled and online	Enabled and offline	Enabled and offline
Sun Cluster command to bring online a resource	Enabled and offline	Enabled and offline	Enabled and online	Enabled and online
Oracle CRS command to start a resource	Enabled and offline	Enabled and offline	Enabled and online	Enabled and online
Sun Cluster command to disable a resource	Enabled and online	Enabled and online	Disabled and offline	Disabled and offline
Oracle CRS command to disable a resource	Enabled and online	Enabled and online	Enabled and online	Disabled and online
Oracle SQLPLUS command to shut down the database	Enabled and online	Enabled and online	Enabled and offline	Enabled and offline
Sun Cluster command to enable a resource	Disabled and offline	Disabled and offline	Enabled and online or offline	Enabled and online or offline
Oracle CRS command to enable a resource	Disabled and offline	Disabled and offline	Disabled and offline	Enabled and offline

The names of the states of Sun Cluster resources and Oracle CRS resources are identical. However, the meaning of each state name is different for Sun Cluster resources and Oracle CRS resources. For more information, see the following table.

TABLE 4-3 Comparisons of States for Sun Cluster Resources and Oracle CRS Resources

State	Meaning for Sun Cluster Resources	Meaning for Oracle CRS Resources
Enabled	The resource is available to the Sun 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 CRS 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 Sun 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 CRS 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 Sun Cluster resources, see “Resource and Resource Group States and Settings” in *Sun Cluster Concepts Guide for Solaris OS*.

For detailed information about the state of Oracle CRS resources, see your Oracle documentation.

Effects of State Changes to Sun Cluster Resources for Oracle 9i RAC Database Instances

In Oracle 9i, Oracle database components can be stopped and started exclusively under the control of the Sun Cluster RGM. The effects of state changes to Sun Cluster resources for Oracle 9i RAC database instances are as follows:

- Bringing online a resource for an Oracle 9i RAC database component starts the component on the nodes where the resource is brought online.
- Taking offline a resource for an Oracle 9i RAC database component stops the component on the nodes where the resource is taken offline.

Tuning Sun Cluster Support for Oracle RAC

To tune the Sun Cluster 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, “Sun Cluster Support for Oracle RAC Extension Properties”](#). Typically, you use the option `-p property=value` of the `clresource(1CL)` command to set extension properties of Sun Cluster Support for Oracle RAC resources. You can also use the procedures in Chapter 2, “Administering Data Service Resources,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* to configure the resources later.

Guidelines for Setting Timeouts

Many of the extension properties for Sun Cluster 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.

SPARC: VxVM Component Reconfiguration Step 4 Timeout

The time that is required for step 4 of a reconfiguration of the VxVM component of Sun Cluster 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 207.

EXAMPLE 4-1 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 Sun Cluster 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 201](#).

EXAMPLE 4-2 Setting the Reservation Step Timeout

```
# clresource set -p reservation_timeout=350 rac-framework-rs
```

This example sets the timeout for the reservation step of a reconfiguration of Sun Cluster 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-framework-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 202](#).

EXAMPLE 4-3 Setting the Communications Port Number for the Oracle UDLM

```
# clresource set -p port=7000 rac-udlm-rs
```

EXAMPLE 4-3 Setting the Communications Port Number for the Oracle UDLM (Continued)

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 112.

▼ 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:

- `SUNW.rac_udlm` – Only when the Oracle UDLM is *not* running on any cluster node
- `SUNW.rac_cvm` – Only when VxVM is *not* running in cluster mode on any cluster node

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 Sun Cluster Objects” on page 106.

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 Sun Cluster Support for Oracle RAC Fault Monitors

Fault monitoring for the Sun Cluster 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 9i RAC server resource
- Oracle 9i RAC listener resource

Each fault monitor is contained in a resource whose resource type is shown in the following table.

TABLE 4-4 Resource Types for Sun Cluster Support for Oracle RAC Fault Monitors

Fault Monitor	Resource Type
Scalable device group	SUNW.ScalDeviceGroup
Scalable file-system mount point	SUNW.ScalMountPoint
Oracle 9i RAC server	SUNW.scalable_rac_server
Oracle 9iRAC 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 Sun Cluster installations. Therefore, you should tune the Sun Cluster Support for Oracle RAC fault monitors *only* if you need to modify this preset behavior.

Tuning the Sun Cluster 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 Sun Cluster Data Services” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*. Information about the Sun Cluster 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

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 Network Appliance 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 StorEdge™ 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

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
2. If the partition is healthy, determining whether the database is operational

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 27.

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 118.

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 118.

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.

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

ERROR_TYPE	Data Type	Meaning
SCAN_LOG	Quoted regular expression	A string in an error message that Oracle has logged to the Oracle alert log file
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.

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 4-4 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 4-5 Ignoring a DBMS Error

```

{
ERROR_TYPE=DBMS_ERROR;
ERROR=4030;
ACTION=none;
CONNECTION_STATE=*;
NEW_STATE=*;
}

```

EXAMPLE 4-5 Ignoring a DBMS Error (Continued)

```
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:

- 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 4-6 Changing the Response to a Logged Alert

```
{  
ERROR_TYPE=SCAN_LOG;  
ERROR="ORA-00600: internal error";  
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 4-7 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.";
}
```

EXAMPLE 4-7 Changing the Maximum Number of Consecutive Timed-Out Probes (Continued)

```

{
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:

- 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.

Troubleshooting Sun Cluster Support for Oracle RAC

If you encounter a problem with Sun Cluster Support for Oracle RAC, troubleshoot the problem by using the techniques that are described in the following sections.

- “Verifying the Status of Sun Cluster Support for Oracle RAC” on page 129
- “Sources of Diagnostic Information” on page 139
- “Common Problems and Their Solutions” on page 140

Verifying the Status of Sun Cluster Support for Oracle RAC

The status of resource groups and resources for Sun Cluster Support for Oracle RAC indicates the status of Oracle RAC in your cluster. Use Sun 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 Sun Cluster Support for Oracle RAC

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- 1 **Become superuser or assume a role that provides `solaris.cluster.read` RBAC authorization.**

2 Display status information for the Sun 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 Sun Cluster Support for Oracle RAC

The following examples show the status of resource groups and resources for a configuration of Sun Cluster 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 10g R2 of Oracle RAC. The configuration in this example uses Sun StorEdge™ QFS shared file system on Solaris Volume Manager for Sun Cluster to store Oracle files.

The resource groups and resources for this configuration are shown in the following table.

Resource Group	Purpose	Resource Group Contents	
		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.rac_svm	rac-svm-rs
		SUNW.crs_framework	crs_framework-rs

Resource Group	Purpose	Resource Group Contents	
		Resource Type	Resource Instance Name
scalldg-rg	Resource group for scalable device-group resources	SUNW.ScalDeviceGroup	scaloradg-rs
qfsmnds-rg	Resource group for Sun StorEdge 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 5-1 Status of a Faulty RAC Framework Resource Group

```
# 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

scalldg-rg      pclus1        No             Pending online blocked
                  pclus2        No             Online
                  pclus3        No             Online
                  pclus4        No             Online

qfsmnds-rg      pclus1        No             Offline
                  pclus2        No             Online
                  pclus3        No             Offline
                  pclus4        No             Offline

scalmnt-rg      pclus1        No             Pending online blocked
                  pclus2        No             Online
                  pclus3        No             Online
                  pclus4        No             Online

rac_server_proxy-rg  pclus1        No             Pending online blocked
                  pclus2        No             Online
                  pclus3        No             Online
```

EXAMPLE 5-1 Status of a Faulty RAC Framework Resource Group (Continued)

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
rac-svm-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 scalddg-rg +

=== Cluster Resources ===

Resource Name	Node Name	State	Status Message
scaloradg-rs	pclus1	Offline	Offline
	pclus2	Online	Online - Diskgroup online
	pclus3	Online	Online - Diskgroup online
	pclus4	Online	Online - Diskgroup online

clresource status -g qfsmdds-rg +

=== Cluster Resources ===

Resource Name	Node Name	State	Status Message
qfs-db_qfs-OraHome-rs	pclus1	Offline	Offline

EXAMPLE 5-1 Status of a Faulty RAC Framework Resource Group *(Continued)*

```

                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	Offline	Offline
	pclus2	Online	Online
	pclus3	Online	Online
	pclus4	Online	Online
scal-db_qfs-OraData-rs	pclus1	Offline	Offline
	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

This example provides the following status information for a RAC framework resource group that is faulty.

- A reconfiguration error has prevented the `rac_f` 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_f` framework-rg resource group is offline and in the start failed state on cluster node `pclus1`.

EXAMPLE 5-1 Status of a Faulty RAC Framework Resource Group *(Continued)*

- 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.

EXAMPLE 5-2 Status of a Faulty RAC Database Resource Group

```
# 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
scalldg-rg	pclus1	No	Online
	pclus2	No	Online
	pclus3	No	Online
	pclus4	No	Online
qfsmgs-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 +
```

EXAMPLE 5-2 Status of a Faulty RAC Database Resource Group (Continued)

```
=== Cluster Resources ===
```

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
rac-svm-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 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 5-2 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

This example provides the following status information for a RAC database resource group that is faulty:

- The 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.

EXAMPLE 5-3 Status of an Operational Oracle RAC Configuration

```

# 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

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

```

EXAMPLE 5-3 Status of an Operational Oracle RAC Configuration (Continued)

```

rac-svm-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 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

EXAMPLE 5-3 Status of an Operational Oracle RAC Configuration (Continued)

```

                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	Online	Online - Oracle instance UP
	pclus2	Online	Online - Oracle instance UP
	pclus3	Online	Online - Oracle instance UP
	pclus4	Online	Online - Oracle instance UP

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.

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 directory `/var/cluster/ucmm` contains the sources of diagnostic information that are shown in the following table.

Source	Location
Oracle UDLM core files	<code>/var/cluster/ucmm/dlm_nodename/cores</code>
	If you cannot find the Oracle log files at this location, contact Oracle support.

Source	Location
Log file for the current userland cluster membership monitor (UCMM) reconfiguration	<code>/var/cluster/ucmm/ucmm_reconf.log</code>
Log files for previous UCMM reconfigurations	<code>/var/cluster/ucmm/ucmm_reconf.log.0 (0,1,...)</code> This location is dependent on the Oracle UDLN package.
Log files for UNIX Distributed Lock Manager (Oracle UDLN) events	<code>/var/cluster/ucmm/dlm_nodename/logs</code> If you cannot find the Oracle log files at this location, contact Oracle support.

The directory `/var/opt/SUNWscor/oracle_server/proxyresource` contains log files for the resource that represents the Oracle 10g R2 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 9i 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC. Each subsection provides information about the cause of the problem and a solution to the problem.

Node Panic During Initialization of Sun Cluster Support for Oracle RAC

If a fatal problem occurs during the initialization of Sun Cluster Support for Oracle RAC, the node panics with an error message 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:

- The license for VERITAS Volume Manager (VxVM) is missing or has expired.
- The ORCLud_{tm} package that contains the Oracle UDLM is not installed.
- The version of the Oracle UDLM is incompatible with the version of Sun Cluster Support for Oracle RAC.
- The amount of shared memory is insufficient to enable the Oracle UDLM to start.

A node might also panic during the initialization of Sun Cluster Support for Oracle RAC because a reconfiguration step has timed out. For more information, see [“Node Panic Caused by a Timeout” on page 143](#).

Solution: For instructions to correct the problem, see [“How to Recover From a Failure of the UCMM or a Related Component” on page 142](#).

Failure of the ucmmd Daemon to Start

The UCMM daemon, `ucmmd`, manages the reconfiguration of Sun Cluster Support for Oracle RAC. When a cluster is booted or rebooted, this daemon is started only after all components of Sun Cluster Support for Oracle RAC are validated. If the validation of a component on a node fails, the `ucmmd` fails to start on the node.

The most common causes of this problem are as follows:

- The ORCLud_{tm} package that contains the Oracle UDLM is not installed.
- An error occurred during a previous reconfiguration of a component Sun Cluster Support for Oracle RAC.
- A step in a previous reconfiguration of Sun Cluster 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 UCMM or a Related Component” on page 142](#).

▼ How to Recover From a Failure of the UCMM or a Related Component

Perform this task to correct the problems that are described in the following sections:

- [“Node Panic During Initialization of Sun Cluster Support for Oracle RAC” on page 141](#)
- [“Failure of the ucmm Daemon to Start” on page 141](#)

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 139](#).

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 *Sun Cluster Error Messages Guide for Solaris OS*.

2 Correct the problem that caused the component to return an error to the UCMM.

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.
- **If the ORCLud_{tm} package that contains the Oracle UDLM is not installed, ensure that the package is installed.**
 - 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 33](#).

- **If the version of the Oracle UDLM is incompatible with the version of Sun Cluster Support for Oracle RAC, install a compatible version of the package.**

For more information, see [“SPARC: Installing the Oracle UDLM” on page 33](#).

- **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” on page 30](#).

- **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 143](#).

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 Cluster Node” in *Sun Cluster System Administration Guide for Solaris OS*](#).

4 On the node where the problem occurred, bring online the RAC framework resource group.

- a. **Become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization.**

- b. **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
```

`-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 moved to the MANAGED state and brought online.

Node Panic Caused by a Timeout

The timing out of any step in the reconfiguration of Sun Cluster 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 110](#).

If a reconfiguration step times out, use the Sun Cluster maintenance commands to increase the value of the extension property that specifies the timeout for the step. For more information, see [Appendix C, “Sun Cluster 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 Node

Recovering from the failure of a node involves the following tasks:

1. Booting into maintenance mode the node that panicked
2. Performing the appropriate recovery action for the cause of the problem
3. Rebooting the node that panicked

For more information, see *Sun Cluster System Administration Guide for Solaris OS*.

Note – In an Oracle RAC environment, multiple Oracle instances cooperate to provide access to the same shared database. The Oracle clients can use any of the instances to access the database. Thus, if one or more instances have failed, clients can connect to a surviving instance and continue to access the database.

Failure of a SUNW.rac_f framework Resource to Start

If a SUNW.rac_f 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 Sun Cluster Support for Oracle RAC” on page 129](#).

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 as follows:

`Faulted - ucmmmd is not running`

Description: The ucmmmd daemon is not running on the node where the resource resides.

Solution: For information about how to correct this problem, see [“Failure of the ucmmmd Daemon to Start” on page 141](#).

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 Sun Cluster 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 UCMM or a Related Component” on page 142](#).
- 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 145](#).

▼ How to Recover From the Timing Out of the START Method

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

- 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 RAC framework resource group.**

To perform this operation, switch the primary nodes of the resource group to the other nodes where this group is online.

```
# clresourcegroup switch -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 RAC framework resource group. If this 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 Sun Cluster Support for Oracle RAC, bring the RAC framework resource group 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 *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC

This chapter explains how to modify an existing configuration of Sun Cluster Support for Oracle RAC.

- [“Overview of Tasks for Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC” on page 147](#)
- [“Modifying Online the Resource for a Scalable Device Group” on page 148](#)
- [“Upgrading Sun Cluster Support for Oracle RAC” on page 148](#)
- [“Extending an Existing Configuration of Sun Cluster Support for Oracle RAC” on page 157](#)
- [“Removing Sun Cluster Support for Oracle RAC” on page 167](#)

Overview of Tasks for Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC

[Table 6–1](#) summarizes the administration tasks for Sun Cluster Support for Oracle RAC.

Perform these tasks whenever they are required.

TABLE 6–1 Tasks for Modifying an Existing Configuration of Sun Cluster 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 148
Upgrade Sun Cluster Support for Oracle RAC to upgrade your cluster to use a new version of Sun Cluster	“Upgrading Sun Cluster Support for Oracle RAC” on page 148
Extend an existing configuration of Sun Cluster Support for Oracle RAC	“Extending an Existing Configuration of Sun Cluster Support for Oracle RAC” on page 157

TABLE 6-1 Tasks for Modifying an Existing Configuration of Sun Cluster Support for Oracle RAC
(Continued)

Task	Instructions
Remove Sun Cluster Support for Oracle RAC	“Removing Sun Cluster Support for Oracle RAC” on page 167

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.

Upgrading Sun Cluster Support for Oracle RAC

If you upgrade Sun Cluster foundation software to release 3.2, you must also upgrade the Sun Cluster Support for Oracle RAC software. For more information, see “Upgrading Sun Cluster Software,” in *Sun Cluster Software Installation Guide for Solaris OS*.

Your existing configuration of Sun Cluster Support for Oracle RAC might not include the RAC framework resource group. In this situation, you *must* register and configure the RAC framework resource group after upgrading to Sun Cluster 3.2. Otherwise, Oracle RAC cannot run with Sun Cluster. For more information, see [“Registering and Configuring the RAC Framework Resource Group” on page 53](#).

Sun Cluster 3.2 introduces several new features for Sun Cluster Support for Oracle RAC. Information about how to upgrade an existing configuration to use these new features 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 Sun Cluster 3.2. If you choose to continue to use the `SUNW.oracle_rac_server` resource type and the `SUNW.oracle_listener` resource type for Oracle 9i RAC database instances, no upgrade of these resource types is required.

Upgrading Resources in the RAC Framework Resource Group

Changes to resource types of resources in the RAC framework for Sun Cluster 3.2 are summarized in the following table.

TABLE 6-2 Changes to RAC Framework Resource Types

Resource Type	Changes
<code>SUNW.rac_cvm</code>	Extension property <code>reservation_timeout</code> removed.
<code>SUNW.rac_framework</code>	Extension property <code>reservation_timeout</code> added.
<code>SUNW.rac_hwraid</code>	Removed from Sun Cluster 3.2.
<code>SUNW.rac_svm</code>	Extension property <code>reservation_timeout</code> removed.
<code>SUNW.rac_udlm</code>	Tunability of the <code>oracle_config_file</code> extension property changed from any time to when disabled. Minimum allowed value of <code>port</code> extension property reduced from 1024 to 1.

Upgrade these resource types if you are upgrading to version 3.2 from an earlier version of Sun Cluster Support for Oracle RAC.

For general instructions that explain how to upgrade a resource type, see “Upgrading a Resource Type” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 RAC Framework Resource Types

The resource type versions of changed RAC framework resource types are listed in the following table for version 3.1 and version 3.2 of Sun Cluster. This table also lists the name of the Resource Type Registration (RTR) file of each changed resource type.

TABLE 6-3 Information for Registering the New Version of RAC Framework Resource Types

Resource Type	Resource Type Version in Sun Cluster 3.1	Resource Type Version in Sun Cluster 3.2	RTR File
SUNW.rac_cvm	1.0	2	/usr/cluster/lib/ucmm/rt/rac_cvm/etc/SUNW.rac_cvm
SUNW.rac_framework	1.0	2	/usr/cluster/lib/ucmm/rt/rac_framework/etc/SUNW.rac_framework
SUNW.rac_svm	1.0	2	/usr/cluster/lib/ucmm/rt/rac_svm/etc/SUNW.rac_svm
SPARC: SUNW.rac_udlm	1.0	2	/usr/cluster/lib/ucmm/rt/rac_udlm/SUNW.rac_udlm

To determine the version of the resource type that is registered, use the following command:

```
# clresourcetype show resource-type
```

resource-type Specifies the resource type whose version you are determining.

Information for Migrating Existing Instances of RAC Framework Resource Types

The information that you require to edit each instance of RAC framework resource types is listed in the following table.

TABLE 6-4 Information for Migrating Existing Instances of RAC Framework Resource Types

Resource Type	Value of type_version Property	Possible Migration Times	Changes to Extension Properties
SUNW.rac_cvm	2	Any time	Extension property reservation_timeout removed.
SUNW.rac_framework	2	Any time	Extension property reservation_timeout added.
SUNW.rac_svm	2	Any time	Extension property reservation_timeout removed.

TABLE 6-4 Information for Migrating Existing Instances of RAC Framework Resource Types
(Continued)

Resource Type	Value of type_version Property	Possible Migration Times	Changes to Extension Properties
SPARC: SUNW.rac_udlm	2	Any time	Tunability of the <code>oracle_config_file</code> extension property changed from any time to when disabled. Minimum allowed value of <code>port</code> extension property reduced from 1024 to 1.

The following example shows a command for editing an instance of the `SUNW.rac_udlm` resource type.

Note – The example shows the setting of an extension property that can be set only when the resource is disabled. The step for disabling the resource is not shown in the example.

EXAMPLE 6-1 Editing an Instance of the `SUNW.rac_udlm` Resource Type

```
# clresource set -p type_version=2 \  
-p port=100 rac_udlm-rs
```

This command edits a `SUNW.rac_framework` resource as follows:

- The `SUNW.rac_udlm` resource is named `rac_udlm`.
- The `type_version` property of this resource is set to 2.
- The communications port number that the Oracle UDLM uses is set to 100.

▼ 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 Sun Cluster Support for Oracle RAC.

In Sun Cluster release 3.1, `reservation_timeout` is an extension property of the following resource types:

- `SUNW.rac_cvm`
- `SUNW.rac_hwraid`
- `SUNW.rac_svm`

In Sun Cluster release 3.2, `reservation_timeout` is an extension property *only* of the `SUNW.rac_framework` resource type. 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 from 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 Sun Cluster release 3.2, 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. In Sun Cluster release 3.2, *version* is 2.

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 6-2 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 Sun Cluster framework for RAC
Per-node:	False
Type:	int

During the migration of the resource of type `SUNW.rac_f` framework, the following command is run to set properties as follows:

- The `type_version` property is set to 2.
- The `reservation_timeout` extension property is set to 350 seconds.

```
# clresource set -p type_version=2 \  
-p reservation_timeout=350 rac_framework
```

See Also The following documentation:

- [“Information for Registering the New Versions of RAC Framework Resource Types” on page 150](#)
- [“Information for Migrating Existing Instances of RAC Framework Resource Types” on page 150](#)
- The `clresource(1CL)` man page

▼ How to Remove the Resource of Type `SUNW.rac_hwraid`

In 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 Sun Cluster release 3.2. 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

Sun Cluster 3.2 introduces 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 the RAC Framework Resource Group” on page 149](#).

For detailed instructions for adding storage resources for Oracle files, see [“Registering and Configuring Storage Resources for Oracle Files” on page 65](#).

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 Sun Cluster 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 RAC Framework Resource Group” on page 165](#)

Adding Resources for Interoperation With Oracle 10g R2 CRS

Sun Cluster 3.2 introduces resource types that enable Sun Cluster and Oracle 10g R2 CRS to interoperate. These resource types also enable Oracle RAC database instances to be administered from Sun Cluster.

Note – Oracle 10g R1 CRS and Sun Cluster cannot interoperate. Instead, Oracle 10g R1 CRS start and shut down Oracle RAC database instances.

Before adding resources for interoperation with Oracle 10g R2 CRS, ensure that you have performed the tasks in the following sections:

- [“Upgrading Resources in the RAC Framework Resource Group” on page 149](#)
- [“Adding Storage Resources for Oracle Files” on page 153](#)

For detailed instructions for adding resources for interoperation with Oracle 10g R2 CRS, see [“Configuring Resources for Oracle RAC Database Instances” on page 82](#).

Upgrading to Sun Cluster 3.2 Resource Types for Oracle 9i RAC Database Instances

Sun Cluster 3.2 replaces the failover resource types that automate the startup and shutdown of Oracle 9i RAC database instances with multi-master resource types. The multi-master resources simplify the administration and enhance the scalability of Sun Cluster Support for Oracle RAC.

Upgrading to Sun Cluster 3.2 resource types for Oracle 9i 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 155](#)
2. [“How to Modify the Configuration of Sun Cluster Resources for Oracle 9i RAC Database Instances” on page 156](#)

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 the RAC Framework Resource Group” on page 149](#).

- 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 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 Sun Cluster Resources for Oracle 9i RAC Database Instances”](#) on page 156.

▼ How to Modify the Configuration of Sun Cluster Resources for Oracle 9i 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 155 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 Sun StorEdge QFS shared file system, remove all resource groups that contain resources for the Sun StorEdge QFS metadata server.

If you are not using the Sun StorEdge 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 StorEdge 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 153.

6 Configure new resources for Oracle 9i RAC database instances.

For instructions for performing this task, see [“Configuring Resources for Oracle RAC Database Instances”](#) on page 82.

Extending an Existing Configuration of Sun Cluster Support for Oracle RAC

Extend an existing configuration of Sun Cluster Support for Oracle RAC in any of the following situations:

- You are adding nodes to a cluster and you require Sun Cluster Support for Oracle RAC to run on the nodes. For more information, see [“How to Add Sun Cluster Support for Oracle RAC to Selected Nodes”](#) on page 158.
- You are adding a volume manager. For more information, see [“How to Add a Volume Manager Resource to the RAC Framework Resource Group”](#) on page 165.

▼ How to Add Sun Cluster Support for Oracle RAC to Selected Nodes

Perform this procedure if you are adding nodes to a cluster and you require Sun Cluster Support for Oracle RAC to run on the nodes. Perform this procedure from only one node.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 RAC framework resource group
- Resource groups for scalable device group resources
- Resource groups that contain resources for the Sun StorEdge™ QFS metadata server
- Resource groups for logical hostname resources
- The resource group for the RAC database

Before You Begin Ensure that the required Sun Cluster Support for Oracle RAC software packages are installed on each node to which you are adding Sun Cluster Support for Oracle RAC. For more information, see [“Installing the Sun Cluster Support for Oracle RAC Packages” on page 31](#).

- 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 Sun Cluster 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 RAC framework resource group.**

```
# clresourcegroup add-node -S -n nodelist rac-fwk-rg
```

-n *nodelist*

Specifies a comma-separated list of cluster nodes to which you are adding Sun Cluster Support for Oracle RAC.

rac-fwk-rg

Specifies the name of the resource group to which you are adding nodes.

4 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.

5 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 Sun Cluster Support for Oracle RAC.

scal-dg-rg

Specifies the name of the resource group to which you are adding nodes.

6 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.

7 Add the nodes to any resource groups that contain resources for the Sun StorEdge QFS metadata server.

If no resource groups that contain resources for the Sun StorEdge 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 Sun Cluster Support for Oracle RAC.

qfs-mds-rg

Specifies the name of the resource group to which you are adding nodes.

8 Bring online all resource groups to which you added nodes in Step 5.

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.

9 (Oracle 10g R2 only) Start the Oracle CRS.

If you are using Oracle 9i or Oracle 10g R1, omit this step.

```
# /etc/init.d/init.crs start
```

Startup will be queued to init within 30 seconds.

10 (Oracle 9i only) Add the nodes to all resource groups that contain logical hostname resources for each RAC database that is to run on the nodes.

If you are using Oracle 10g R1 or Oracle 10g R2, omit this step. For Oracle 10g R1 and Oracle 10g R2, 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 Sun Cluster Support for Oracle RAC.

lh-rg Specifies the name of the resource group to which you are adding nodes.

11 (Oracle 9i and Oracle 10g R2 only) Add the nodes to the resource group for each RAC database that is to run on the nodes.

If you are using Oracle 10g R1, omit this step. For Oracle 10g R1, no resource groups for 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 Sun Cluster Support for Oracle RAC.

rac-db-rg Specifies the name of the resource group to which you are adding nodes.

12 (Oracle 10g R2 only) For each node that you are adding, create the Oracle CRS resources that are required to represent Sun Cluster resources.

Create an Oracle CRS resource for each Sun 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 CRS Resource for Interoperation With Sun Cluster” on page 240.](#)

13 (Oracle 9i and Oracle 10g R2 only) Modify each resource for RAC databases to set a value of each per-node property for each node that you are adding.

If you are using Oracle 10g R1, omit this step. For Oracle 10g R1, no resource groups for 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 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 RAC databases, see the following sections:

- “SUNW.scalable_rac_server_proxy Extension Properties” on page 216
- “SUNW.scalable_rac_listener Extension Properties” on page 214
- “SUNW.scalable_rac_server Extension Properties” on page 210

```
# 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.

14 (Oracle 9i and Oracle 10g R2 only) Bring online each resource group for RAC databases.

If you are using Oracle 10g R1, omit this step. For Oracle 10g R1, no resource groups for 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 6-3 Adding Sun Cluster Support for Oracle RAC to Selected Nodes

This example shows the sequence of operations that is required to add Sun Cluster Support for Oracle RAC to nodes `pc1us3` and `pc1us4` of a four-node cluster.

The configuration of Sun Cluster Support for Oracle RAC in this example is as follows:

- The version of Oracle RAC is 10g R2.
- Sun StorEdge 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 name of the Oracle RAC database is swb.
- The Sun StorEdge 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 2-1](#).

The configuration of resource groups in this example is shown in the following table.

Resource Group	Purpose
rac-framework-rg	RAC framework resource group.
sca1dg-rg	Resource group for scalable device-group resources.
qfsmds-rg	Resource group for Sun StorEdge QFS metadata server resources.
sca1mnt-rg	Resource group for scalable file-system mount-point resources.
rac_server_proxy-rg	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 sca1mnt-rg
```

2. To add the nodes to the RAC framework resource group, the following command is run:

```
# clresourcegroup add-node -S -n pclus3,pclus4 rac-framework-rg
```

3. 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
```

4. 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 sca1dg-rg
```

5. 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
```

6. To add the nodes to the resource group that contains resources for the Sun StorEdge QFS metadata server, the following command is run:

```
# clresourcegroup add-node -n pclus3,pclus4 qfsmds-rg
```

7. To bring online the resource group that contains scalable device group resources, the following command is run:

```
# clresourcegroup online scaldg-rg
```

8. To start the Oracle CRS and to verify the correct startup of the Oracle CRS, 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
```

9. To add the nodes to the resource group for the 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 RAC database, the required Oracle CRS resources are created. The creation of these Oracle CRS resources is beyond the scope of this example.

10. 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`.

11. To bring online the resource group for the RAC database, the following command is run:

```
# clresourcegroup online rac_server_proxy-rg
```

▼ How to Add a Volume Manager Resource to the RAC Framework Resource Group

Perform this task if you are adding a volume manager to an existing configuration of Sun Cluster Support for Oracle RAC. The RAC 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 RAC framework resource is disabled and if the UCMM daemon is stopped on all cluster nodes.



Caution – This task requires downtime because you must disable the RAC framework resource and boot 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 resource of type `SUNW.rac_framework` in the RAC framework resource group and any other resources that depend on this resource.**

```
# clresource disable -r rac-fwk-rs
```

rac-fwk-rs Specifies the name of the resource of type `SUNW.rac_framework` that you are disabling.

- 3 **Reboot all the nodes that are in the node list of the RAC 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 `SUNW.rac_svm` resource type.**

```
# clresourcetype register SUNW.rac_svm
```

- b. **Add an instance of the `SUNW.rac_svm` resource type to the RAC framework resource group.**

Ensure that this instance depends on the resource of type `SUNW.rac_framework` that you disabled in [Step 2](#).

```
# clresource create -g rac-fwk-rg \  
-t SUNW.rac_svm \  
-p resource_dependencies=rac-fwk-rs rac-svm-rs
```

-g rac-fmwk-rg

Specifies the name RAC framework resource group. This resource group contains the resource of type SUNW.rac_f framework that you disabled in [Step 2](#).

-p resource_dependencies=rac-fmwk-rs

Specifies that this instance depends on the resource of type SUNW.rac_f framework that you disabled in [Step 2](#).

rac-svm-rs

Specifies the name that you are assigning to the resource of type SUNW.rac_svm.

- **If you are adding VxVM with the cluster feature, register and add the instance as follows.**

- a. **Register the SUNW.rac_cvm resource type.**

```
# clresourcetype register SUNW.rac_cvm
```

- b. **Add an instance of the SUNW.rac_cvm resource type to the resource group that you disabled in [Step 2](#).**

Ensure that this instance depends on the resource of type SUNW.rac_f framework that you disabled in [Step 2](#).

```
# clresource create -g rac-fmwk-rg \  
-t SUNW.rac_cvm \  
-p resource_dependencies=rac-fmwk-rs rac-cvm-rs
```

-g rac-fmwk-rg

Specifies the name RAC framework resource group. This resource group contains the resource of type SUNW.rac_f framework that you disabled in [Step 2](#).

-p resource_dependencies=rac-fmwk-rs

Specifies that this instance depends on the resource of type SUNW.rac_f framework that you disabled in [Step 2](#).

rac-cvm-rs

Specifies the name that you are assigning to the resource of type SUNW.rac_cvm.

- 5 **Bring online and in a managed state the RAC framework resource group and its resources.**

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

rac-fmwk-rg Specifies that the RAC framework resource group is to be moved to the MANAGED state and brought online. This resource group contains the resource of type SUNW.rac_f framework 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 58
VxVM with the cluster feature	“How to Create a VxVM Shared-Disk Group for the Oracle RAC Database” on page 64

Removing Sun Cluster Support for Oracle RAC

You can remove Sun Cluster Support for Oracle RAC from the following entities:

- **A cluster.** For more information, see [“How to Remove Sun Cluster Support for Oracle RAC From a Cluster” on page 167](#).
- **Selected nodes in a cluster.** For more information, see [“How to Remove Sun Cluster Support for Oracle RAC From Selected Nodes” on page 173](#).

▼ How to Remove Sun Cluster Support for Oracle RAC From a Cluster

Perform this task to remove Sun Cluster 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 procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

This task involves removing the following resource groups from the cluster in the following order:

- The resource group for the 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 StorEdge 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 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 9i and Oracle 10g R2 only) Remove the resource group for each RAC database that you are removing.

If you are using Oracle 10g R1, omit this step. For Oracle 10g R1, no resource groups for RAC databases are configured.

For each 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 9i only) Remove all resource groups for logical hostname resources that are used by each RAC database that you are removing.

If you are using Oracle 10g R1 or Oracle 10g R2, omit this step. For Oracle 10g R1 and Oracle 10g R2, 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 RAC database that you no longer require.

5 If you are removing Sun Cluster 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 CRS software

6 (Oracle 10g R2 only) Disable the CRS framework resource.

If you are using Oracle 9i or Oracle 10g R1, omit this step. For Oracle 9i and Oracle 10g R1, no CRS 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 StorEdge QFS metadata server.

If no resource groups that contain resources for the Sun StorEdge 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 StorEdge 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 *Sun StorEdge QFS Configuration and Administration Guide, Version 4, Update 5*.

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/dN`, 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-fwk-rg
```

rac-fwk-rg Specifies the resource group that you are removing.

13 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 Sun Cluster Support for Oracle RAC, see [“Automatically Generated Names for Sun Cluster Objects” on page 106](#).

14 (Optional) From each node in the cluster, uninstall the Sun Cluster Support for Oracle RAC software packages.

Use the Sun Java™ Enterprise System uninstaller for this purpose. For more information, see Chapter 8, “Uninstalling,” in *Sun Java Enterprise System 5 Installation Guide for UNIX*.

15 Reboot each node in the cluster.

Example 6–4 Removing Sun Cluster Support for Oracle RAC From a Cluster

This example shows the sequence of operations that is required to remove Sun Cluster Support for Oracle RAC from all nodes of a four-node cluster. The nodes in this cluster are named `pc1us1`, `pc1us2`, `pc1us3`, and `pc1us4`. Only one RAC database is configured on the cluster.

The configuration of Sun Cluster Support for Oracle RAC in this example is as follows:

- The version of Oracle RAC is 10g R2.
- Sun StorEdge 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 name of the Oracle RAC database is `swb`.
- The Sun StorEdge 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 2–1](#).

The configuration of resource groups in this example is shown in the following table.

Resource Group	Purpose
<code>rac-framework-rg</code>	RAC framework resource group.
<code>sca1dg-rg</code>	Resource group for scalable device-group resources.

Resource Group	Purpose
qfsmds-rg	Resource group for Sun StorEdge QFS metadata server resources.
scalmnt-rg	Resource group for scalable file-system mount-point resources.
rac_server_proxy-rg	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 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 CRS software

The removal of these items is beyond the scope of this example.

2. To disable the CRS 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 StorEdge QFS metadata server, the following command is run:

```
# clresourcegroup delete -F qfsmds-rg
```

After the removal of this resource group, Sun StorEdge QFS utilities are used to remove the Sun StorEdge 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 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC From Selected Nodes

Perform this task to remove Sun Cluster 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 procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

This task involves removing the selected nodes from the following resource groups in the following order:

- The resource group for the 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 StorEdge 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 9i and Oracle 10g R2 only) Remove the nodes from the resource group for each RAC database that you are removing.**

If you are using Oracle 10g R1, omit this step. For Oracle 10g R1, no resource groups for RAC databases are configured.

For each RAC database that you are removing, perform the following steps:

- a. **Take offline the resource group for the RAC database on the nodes from which you are removing Sun Cluster 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 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 9i only) Remove the nodes from all resource groups for logical hostname resources that are used by each RAC database that you are removing.

If you are using Oracle 10g R1 or Oracle 10g R2, omit this step. For Oracle 10g R1 and Oracle 10g R2, 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 Sun Cluster 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 Sun Cluster 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 9i 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 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 10g R2 only) Remove each node that you are removing from the list of nodes where the Oracle CRS resource for the Oracle database runs.

If you are using Oracle 9i or Oracle 10g R1, omit this step. For Oracle 9i and Oracle 10g R1, no Oracle CRS resources that represent Sun Cluster resources are configured.

Note – In this step, the syntax of Oracle commands for Oracle release 10g R2 is provided. If you are using a version of Oracle other than 10g R2, see your Oracle documentation for the correct command syntax.

```
# crs-home/crs_register ora.dbname.sid.inst \
-update -r "ora.node-name.vip"
```

crs-home Specifies the Oracle CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

dbname Specifies the database name of the database instance that the Oracle CRS resource represents.

sid Specifies the Oracle SID of the database instance that the Oracle CRS resource represents.

node-name Specifies the host name of the node where the Oracle CRS resource runs.

6 (Oracle 10g R2 only) From each node that you are removing, remove each Oracle CRS resource that represents a Sun Cluster resource from whose resource group you are removing nodes.

An Oracle CRS resource is configured for each Sun Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend.

If you are using Oracle 9i or Oracle 10g R1, omit this step. For Oracle 9i and Oracle 10g R1, no Oracle CRS resources that represent Sun Cluster resources are configured.

Note – In this step, the syntax of Oracle commands for Oracle release 10g R2 is provided. If you are using a version of Oracle other than 10g R2, see your Oracle documentation for the correct command syntax.

For each Oracle CRS resource that you are removing, perform the following steps on each node from which you are removing the resource:

a. Stop the Oracle CRS resource that you are removing.

```
# crs-home/crs_stop sun.node-name.sc-rs
```

crs-home Specifies the Oracle CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

node-name Specifies the host name of the node where the Oracle CRS resource runs.

sc-rs Specifies the name of the Sun Cluster resource that the Oracle CRS resource represents.

b. Unregister the Oracle CRS resource that you are removing.

```
# crs-home/crs_unregister sun.node-name.sc-rs
```

crs-home Specifies the Oracle CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

node-name Specifies the host name of the node where the Oracle CRS resource runs.

sc-rs Specifies the name of the Sun Cluster resource that the Oracle CRS resource represents.

c. Delete the profile for the Oracle CRS resource that you are removing.

```
# crs-home/crs_profile -delete sun.node-name.sc-rs \
-dir /var/cluster/ucmm/profile
```

crs-home Specifies the Oracle CRS home directory. This directory contains the Oracle CRS binary files and Oracle CRS configuration files.

node-name Specifies the host name of the node where the Oracle CRS resource runs.

sc-rs Specifies the name of the Sun Cluster resource that the Oracle CRS resource represents.

7 Use Oracle utilities to remove the following items from each node from which you are removing Sun Cluster Support for Oracle RAC:

- The RAC database
- Oracle CRS

8 Switch any resource groups that contain resources for the Sun StorEdge QFS metadata server to a node from which you are *not* removing Sun Cluster Support for Oracle RAC.

If no resource groups that contain resources for the Sun StorEdge QFS metadata server are configured, 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 Sun Cluster 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 Sun Cluster 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 StorEdge QFS metadata server.

If no resource groups that contain resources for the Sun StorEdge 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 StorEdge QFS shared file systems from the nodes.

For instructions for performing this task, see *Sun StorEdge QFS Configuration and Administration Guide, Version 4, Update 5*.

- 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC.

```
# clresourcegroup offline -n nodelist rac-fwk-rg
```

-n nodelist

Specifies a comma-separated list of cluster nodes from which you are taking offline the resource group.

rac-fwk-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-fwk-rg
```

-n nodelist

Specifies a comma-separated list of cluster nodes that you are removing from the resource group.

rac-fwk-rg

Specifies the name of the resource group from which you are removing nodes.

15 (Optional) From each node that you removed, uninstall the Sun Cluster Support for Oracle RAC software packages.

Use the Sun Java Enterprise System uninstaller for this purpose. For more information, see Chapter 8, “Uninstalling,” in *Sun Java Enterprise System 5 Installation Guide for UNIX*.

16 Reboot each node from which you removed Sun Cluster Support for Oracle RAC.

Example 6–5 Removing Sun Cluster Support for Oracle RAC From Selected Nodes

This example shows the sequence of operations that is required to remove Sun Cluster Support for Oracle RAC from nodes `pc1us3` and `pc1us4` of a four-node cluster.

The configuration of Sun Cluster Support for Oracle RAC in this example is as follows:

- The version of Oracle RAC is 10g R2.
- Sun StorEdge 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 name of the Oracle RAC database is swb.
- The Sun StorEdge 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 2-1](#).

The configuration of resource groups in this example is shown in the following table.

Resource Group	Purpose
rac-framework-rg	RAC framework resource group.
scaIdg-rg	Resource group for scalable device-group resources.
qfsmds-rg	Resource group for Sun StorEdge QFS metadata server resources.
scaImnt-rg	Resource group for scalable file-system mount-point resources.
rac_server_proxy-rg	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 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 CRS 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 CRS resources that represent Sun Cluster resources, the following commands are run:

```
Removal of resource for Oracle database files from node pclus3
# /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 pclus3
```

```

# /db_qfs/OraHome/crs/bin/crs_stop sun.pclu3.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclu3.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclu3.scaloramnt-OraHome-rs \
-dir /var/cluster/ucmm/profile
    Removal of resource for Oracle database files from node plcu3
# /db_qfs/OraHome/crs/bin/crs_stop sun.pclu4.scaloramnt-OraData-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclu4.scaloramnt-OraData-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclu4.scaloramnt-OraData-rs \
-dir /var/cluster/ucmm/profile
    Removal of resource for Oracle binary files from node plcu4
# /db_qfs/OraHome/crs/bin/crs_stop sun.pclu4.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_unregister sun.pclu4.scaloramnt-OraHome-rs
# /db_qfs/OraHome/crs/bin/crs_profile -delete sun.pclu4.scaloramnt-OraHome-rs \
-dir /var/cluster/ucmm/profile

```

The commands remove Oracle CRS resources that represent the following Sun 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 `plcu3` and `plcu4`, Oracle utilities are used to remove the following items from these nodes:

- The Oracle RAC database
- The Oracle RAC software
- The Oracle CRS 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 StorEdge QFS metadata server to node `plcu1`, the following command is run:

```
# clresourcegroup switch -n plcu1 qfsmds-rg
```

5. To remove nodes `plcu3` and `plcu4` from the resource group that contains scalable file-system mount-point resources, the following commands are run:

```
# clresourcegroup offline -n plcu3,plcu4 scalmnt-rg
# clresourcegroup remove-node -n plcu3,plcu4 scalmnt-rg
```

6. To remove nodes `plcu3` and `plcu4` from the node list of the resource group that contains resources for the Sun StorEdge QFS metadata server, the following command is run:

```
# clresourcegroup remove-node -n plcu3,plcu4 qfsmds-rg
```

After nodes `pclus3` and `pclus4` are removed from the node list, the configuration of the Sun StorEdge QFS shared file systems is removed from these nodes. This operation is beyond the scope of this example.

- 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
```

- 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
```

- 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
```

After the removal of nodes `pclus3` and `pclus4` from the RAC framework resource group, the following operations are performed:

- The removal of the Sun Cluster Support for Oracle RAC software packages from nodes `pclus3` and `pclus4`
- The reboot of nodes `pclus3` and `pclus4`

These operations are beyond the scope of this example.

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
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
rac-svm-rs	pclus1	Online	Online
	pclus2	Online	Online
crs_framework-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

Sample Configurations of This Data Service

The software architectures of Oracle 9i and Oracle 10g are different. As a result of these differences, the configuration of resources and resource groups for the Sun Cluster 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 subsections that follow illustrate the configuration of resources and resource groups for typical combinations of storage management schemes for Oracle 9i and Oracle 10g on the SPARC® platform. On the x86 platform, no Oracle UDLM resource is required.

- [“Sample Oracle 10g Configurations” on page 186](#)
- [“Sample Oracle 9i Configurations” on page 188](#)

Sample Oracle 10g Configurations

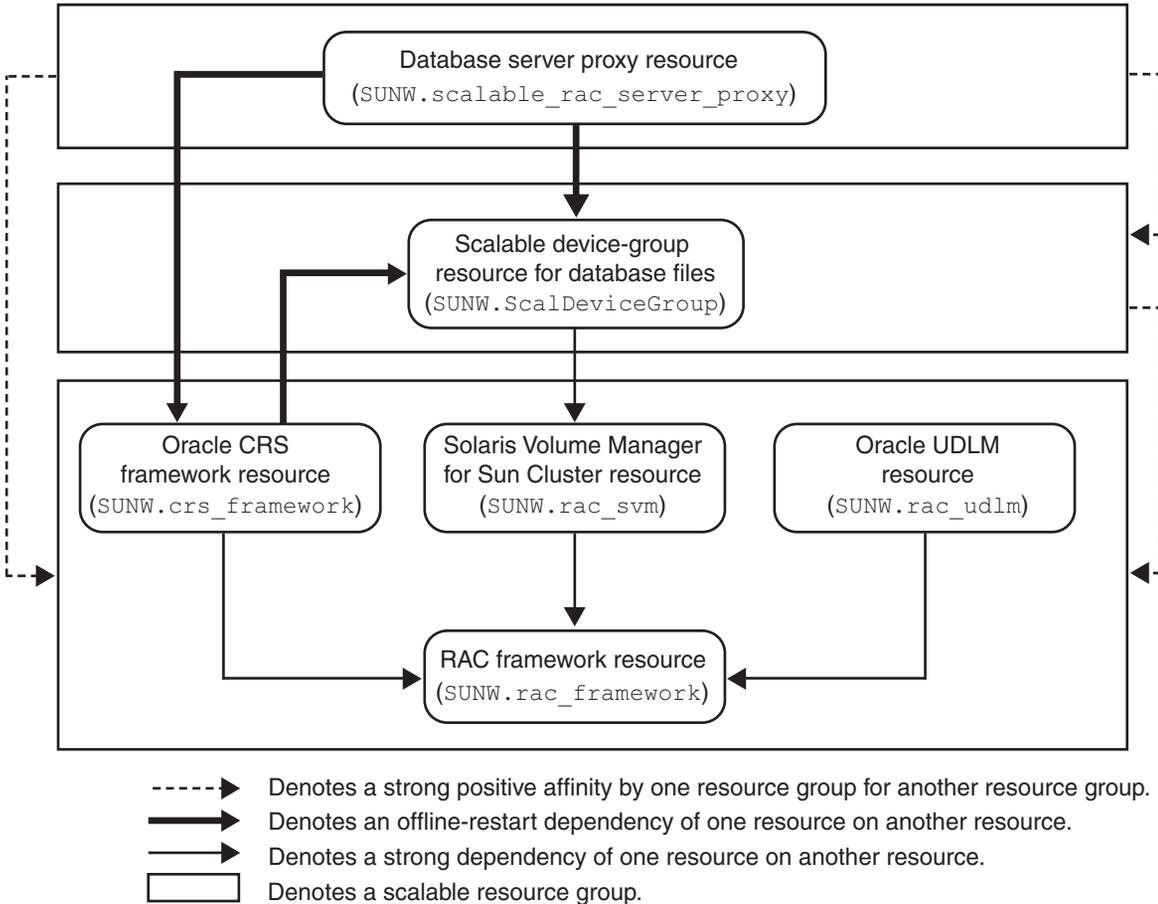


FIGURE A-1 Configuration of Oracle 10g With Solaris Volume Manager for Sun Cluster

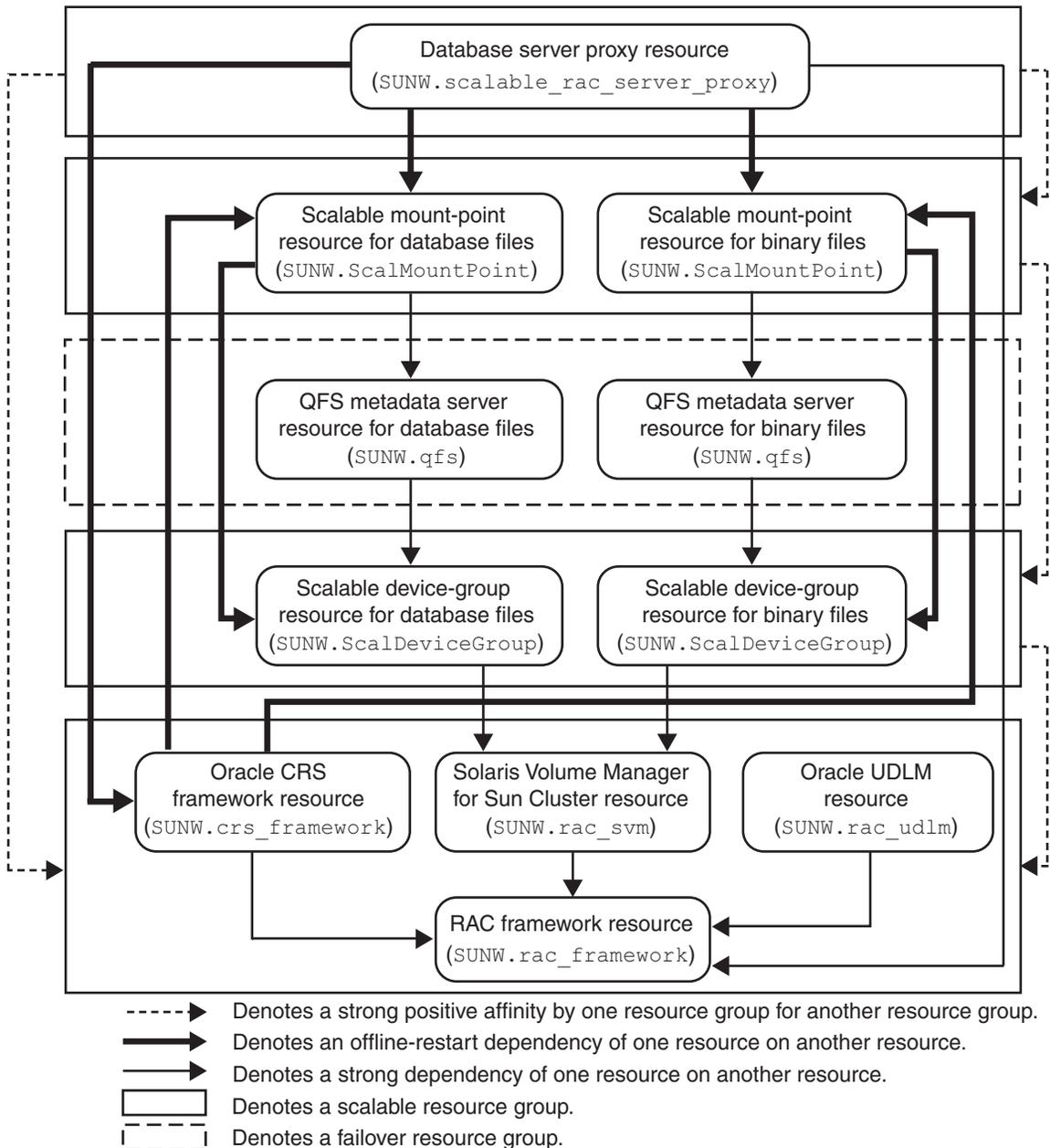


FIGURE A-2 Configuration of Oracle 10g With Solaris Volume Manager for Sun Cluster and Sun StorEdge™ QFS Shared File System

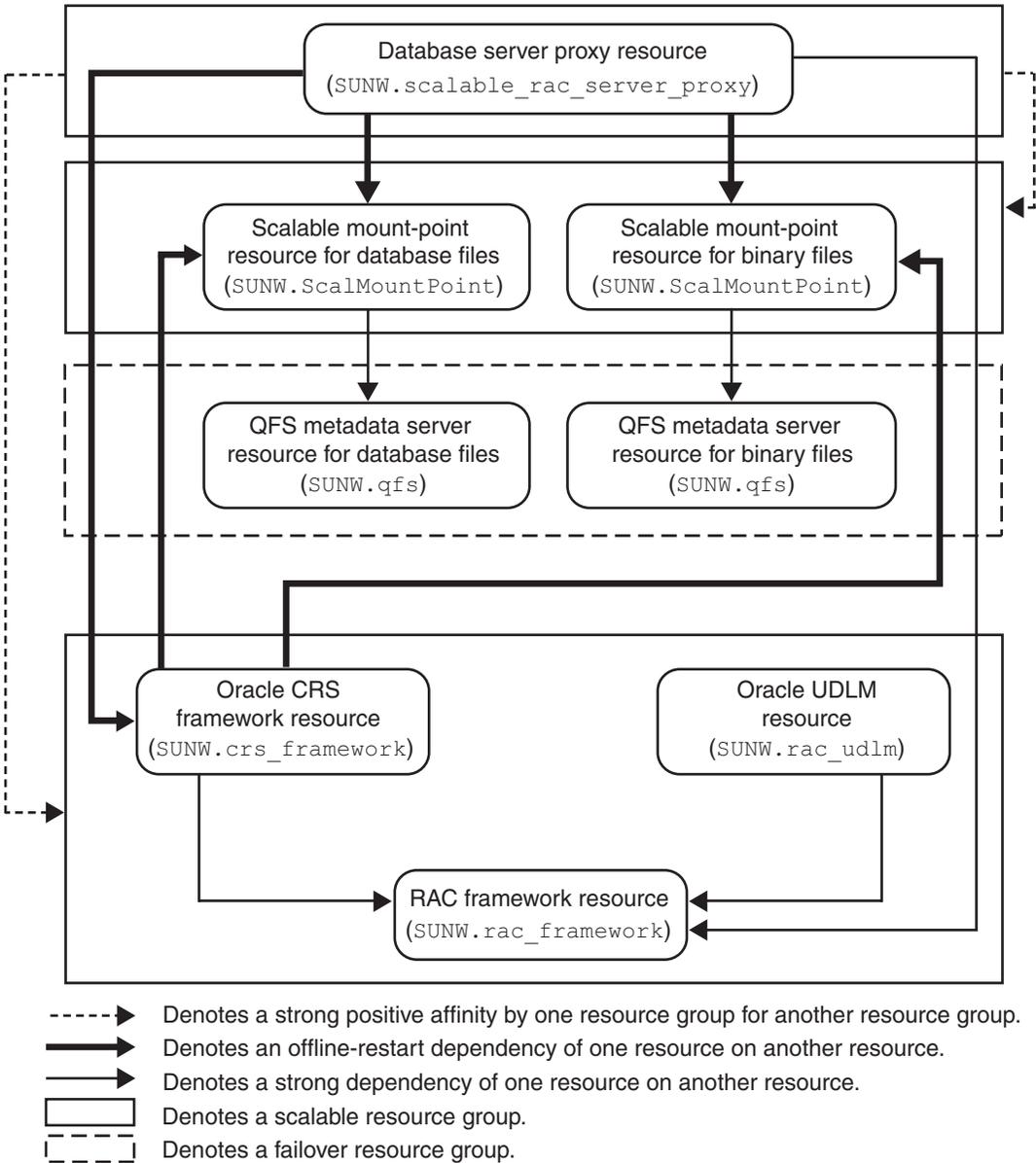


FIGURE A-3 Configuration of Oracle 10g With Sun StorEdge QFS Shared File System and Hardware RAID

Sample Oracle 9i Configurations

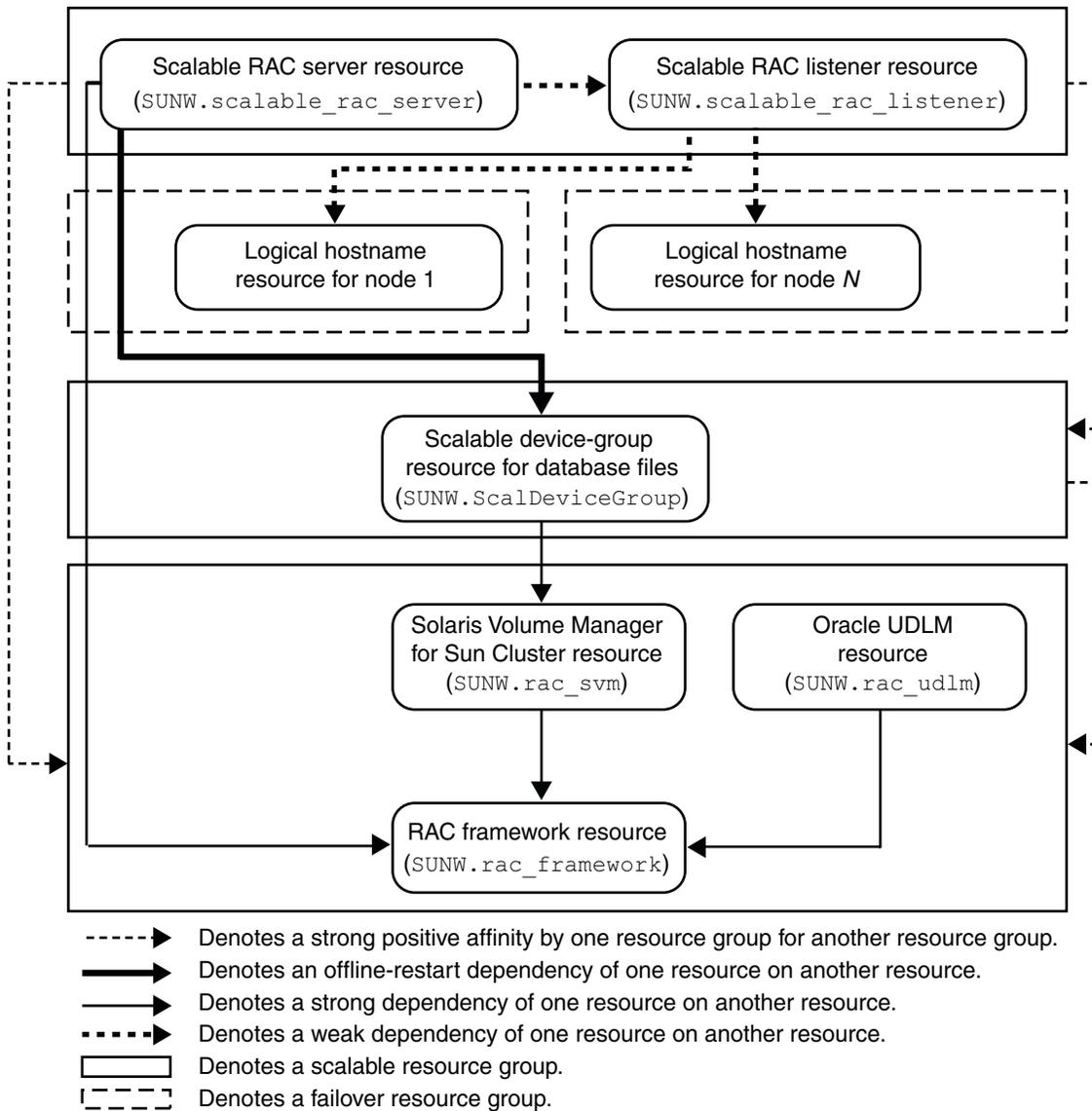


FIGURE A-4 Configuration of Oracle 9i With Solaris Volume Manager for Sun Cluster

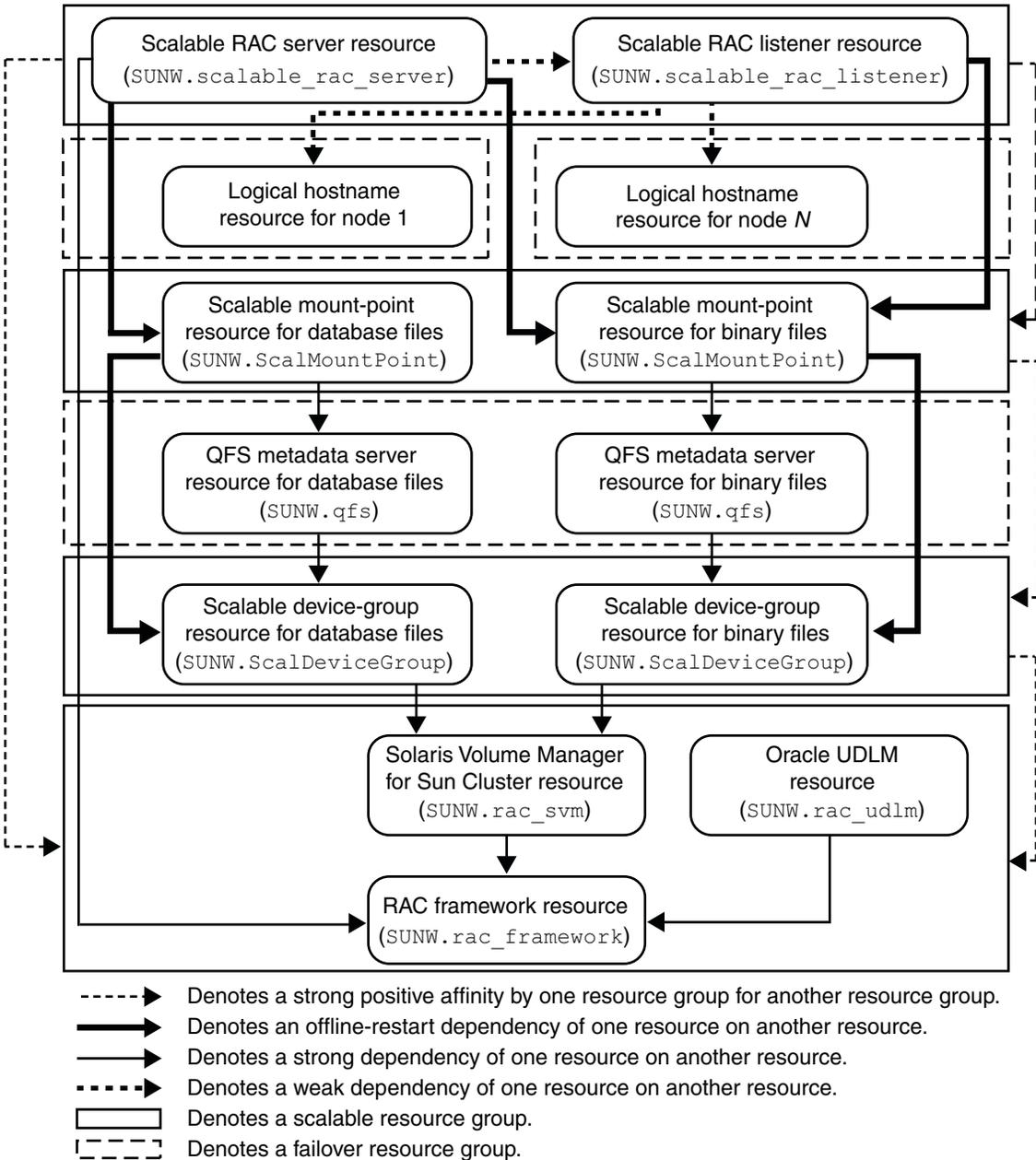


FIGURE A-5 Configuration of Oracle 9i With Solaris Volume Manager for Sun Cluster and Sun StorEdge QFS Shared File System

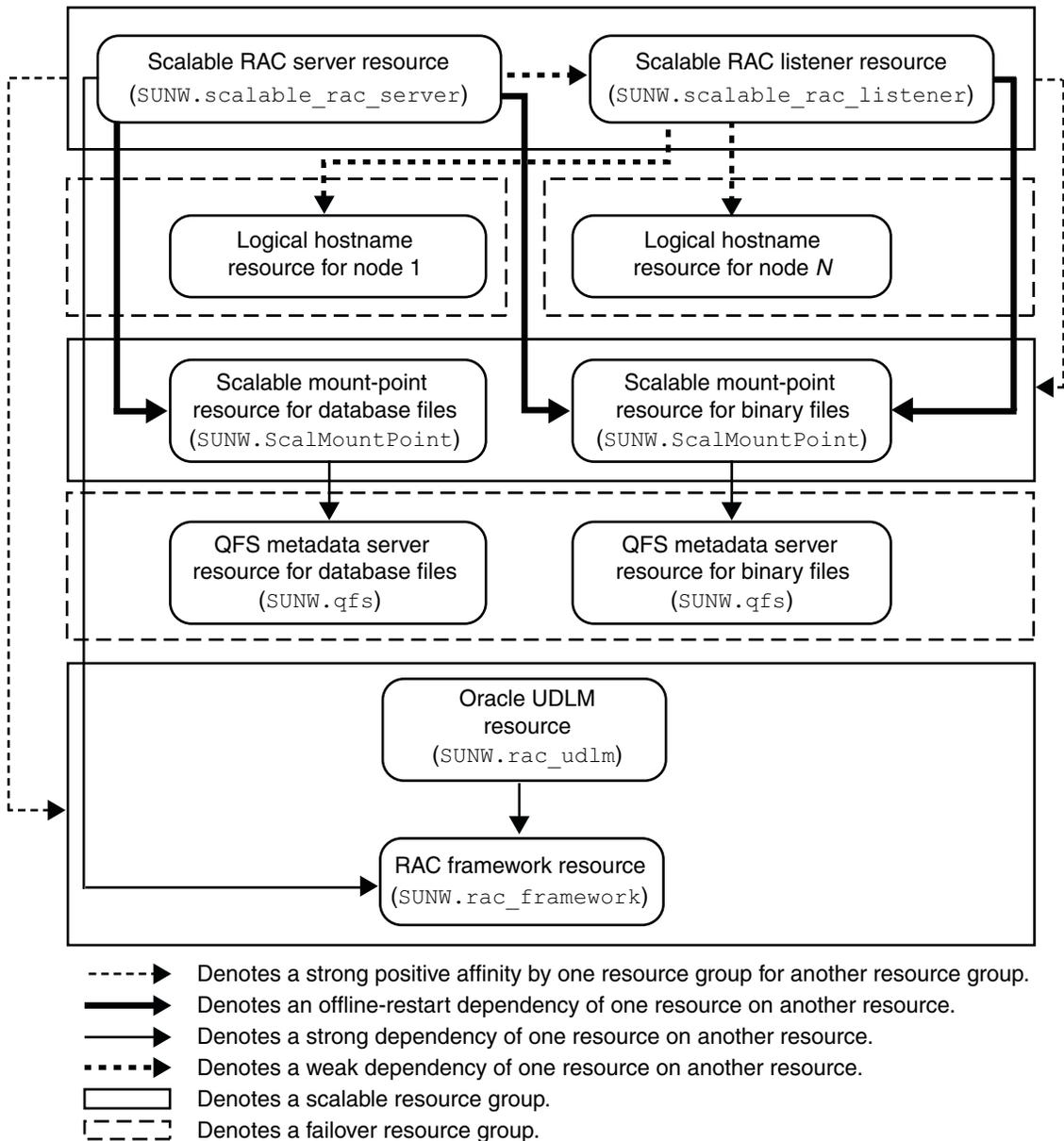


FIGURE A-6 Configuration of Oracle 9i With Sun StorEdge QFS Shared File System and Hardware RAID

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	*	*	osnfpmp: 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	STOP	*	*	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)

Sun Cluster Support for Oracle RAC Extension Properties

The extension properties that you can set for each Sun Cluster Support for Oracle RAC resource type are listed in the following sections:

- “[SUNW.rac_framework Extension Properties](#)” on page 201
- “[SPARC: SUNW.rac_udlm Extension Properties](#)” on page 202
- “[SUNW.rac_svm Extension Properties](#)” on page 205
- “[SPARC: SUNW.rac_cvm Extension Properties](#)” on page 207
- “[SUNW.scalable_rac_server Extension Properties](#)” on page 210
- “[SUNW.scalable_rac_listener Extension Properties](#)” on page 214
- “[SUNW.scalable_rac_server_proxy Extension Properties](#)” on page 216
- “[SUNW.crs_framework Extension Properties](#)” on page 219
- “[SUNW.ScalDeviceGroup Extension Properties](#)” on page 219
- “[SUNW.ScalMountPoint Extension Properties](#)” on page 221

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 112. 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.

SUNW.rac_framework **Extension Properties**

`reservation_timeout`

This property specifies the timeout (in seconds) for the reservation step of a reconfiguration of Sun Cluster Support for Oracle RAC.

Data Type: Integer

Default: 325

Range: 100 – 99999 seconds

Tunable: Any time

SPARC: SUNW.rac_udlm Extension Properties

failfastmode

This property specifies the failfast mode of the node on which the UNIX Distributed Lock Manager (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

`schedclass``Schedclass`

This property specifies the scheduling class of the Oracle UDLM that is passed to the `prioctl(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 `prioctl` 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.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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC.

Data Type: Integer

Default: 40

Range: 30 – 99999 seconds

Tunable: Any time

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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun Cluster 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.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 118](#).

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/proxyrs`. 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/initoracle-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_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_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: 1–25

Default: 5

Tunable: When disabled

crs_home

This property specifies the directory in which the Oracle CRS 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 CRS resource for which this resource is acting as a proxy.

Data Type: Integer

Range: 10–600

Default: 120

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 CRS resource for which this resource is acting as a proxy.

Data Type: Integer

Range: 5–100

Default: 25

Tunable: Any time

`startup_wait_count`

This property specifies the maximum number of attempts by this resource to confirm that the Oracle CRS 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 CRS 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.crs_framework Extension Properties

The `SUNW.crs_framework` resource type has no extension properties.

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 Network Appliance NAS device.

`s-qfs` Specifies that the file system is a Sun StorEdge™ 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 StorEdge QFS shared file system, these options are obtained from the `/etc/opt/SUNWsamfs/samfs.cmd` file.
- For a file system on a Network Appliance 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 StorEdge 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 StorEdge QFS shared file system documentation.
- For a file system on a Network Appliance 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 Network Appliance 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 Network Appliance NAS device is exporting.

The Network Appliance NAS device and the file system must already be configured for use with Sun Cluster. For more information, see *Sun Cluster 3.1 - 3.2 With Network-Attached Storage Devices Manual for Solaris OS*.

Data type: String

Default: No default defined

Range: Not applicable

Tunable: When disabled

Command-Line Alternatives

Sun Cluster maintenance commands enable you to automate the creation, modification, and removal of the RAC framework resource group 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 Sun Cluster Support for Oracle RAC Extension Properties” on page 225
- “Registering and Configuring the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands” on page 226
- “Creating Storage Management Resources by Using Sun Cluster Maintenance Commands” on page 229
- “Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands” on page 235
- “Registering and Configuring Sun Cluster Resources for Interoperation With Oracle 9i by Using Sun Cluster Maintenance Commands” on page 242

Setting Sun Cluster 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC” on page 110
- Appendix C, “Sun Cluster Support for Oracle RAC Extension Properties”

Registering and Configuring the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands

The task in this section is an alternative for the resource-configuration steps in “How to Register and Configure the RAC Framework Resource Group by Using `clsetup`” on page 53.

Overview of the RAC Framework Resource Group

The RAC framework resource group enables Oracle RAC to run with Sun 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 Sun Cluster commands
- `SPARC: SUNW.rac_udlm`, which represents the UNIX Distributed Lock Manager (Oracle UDLM) component of Sun Cluster Support for Oracle RAC

In addition, the RAC framework resource group contains an instance of a single-instance resource type that represents the volume manager that you are using for Oracle files, if any.

- Solaris Volume Manager for Sun Cluster is represented by the `SUNW.rac_svm` resource type.
- 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.

▼ How to Register and Configure the RAC Framework Resource Group by Using Sun Cluster Maintenance Commands

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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.**

Tip – If you require Sun Cluster 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 Sun Cluster Support for Oracle RAC is to be enabled. The Sun Cluster 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 Sun Cluster 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 Sun Cluster 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 Sun 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 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 Register the SUNW.rac_udlm resource type.

```
# clresourcetype register SUNW.rac_udlm
```

6 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_f 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_f 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 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 SUNW.rac_svm resource type.

```
# clresourcetype register SUNW.rac_svm
```

b. Add an instance of the SUNW.rac_svm resource type to the resource group that you created in Step 2.

Ensure that this instance depends on the rac_f framework resource that you created in Step 4.

```
# clresource create -g rac-fmwk-rg \  
-t SUNW.rac_svm \  
-p resource_dependencies=rac-fmwk-rs rac-svm-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-fwk-rs`
 Specifies that this instance depends on the SUNW.rac_framework resource that you created in [Step 4](#).

`rac-svm-rs`

Specifies the name that you are assigning to the SUNW.rac_svm resource.

- **If you are using VxVM with the cluster feature, register and add the instance as follows.**

- a. **Register the SUNW.rac_cvm resource type.**

```
# clresourcetype register SUNW.rac_cvm
```

- b. **Add an instance of the SUNW.rac_cvm resource type to the resource group that you created in [Step 2](#).**

Ensure that this instance depends on the rac_framework resource that you created in [Step 4](#).

```
# clresource create -g rac-fwk-rg \  
-t SUNW.rac_cvm \  
-p resource_dependencies=rac-fwk-rs rac-cvm-rs
```

`-g rac-fwk-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-fwk-rs`

Specifies that this instance depends on the SUNW.rac_framework resource that you created in [Step 4](#).

`rac-cvm-rs`

Specifies the name that you are assigning to the SUNW.rac_cvm resource.

- 8 **Bring online and in a managed state the RAC framework resource group and its resources.**

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

`rac-fwk-rg` Specifies that the resource group that you created in [Step 2](#) is to be moved to the MANAGED state and brought online.

Creating Storage Management Resources by Using Sun 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 66.

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 StorEdge 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 StorEdge™ QFS or Network Appliance 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 StorEdge QFS shared file system or NFS file system on a Network Appliance NAS device that you are using for Oracle files.

The resource that represents a Sun StorEdge QFS shared file system can start only if the file system's Sun StorEdge QFS metadata server is running. Similarly, the resource that represents a Sun StorEdge QFS shared file system can stop only if the file system's Sun StorEdge QFS metadata server is stopped. To meet this requirement, configure a resource for each Sun StorEdge QFS metadata server. For more information, see [“Resources for the Sun StorEdge QFS Metadata Server”](#) on page 230.

Resources for the Sun StorEdge QFS Metadata Server

If you are using the Sun StorEdge QFS shared file system, create one resource for each Sun StorEdge QFS metadata server. The configuration of resource groups for these resources depends on the version of Oracle that you are using.

Configuration of Sun StorEdge QFS Resource Groups With Oracle 9i and Oracle 10g R2

If you are using Oracle 9i or Oracle 10g R2, 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 StorEdge QFS metadata server.
- If your configuration contains a large number of file systems, configure resources for the Sun StorEdge QFS metadata 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 StorEdge QFS Resource Groups With Oracle 10g R1

If you are using Oracle 10g, Oracle CRS manage 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 StorEdge QFS File Systems for Database Files and Related Files](#)” on page 46. In this situation, ensure that the file system that contains the Oracle CRS 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 CRS are started only after all Sun StorEdge QFS file systems are mounted.

If you are using Oracle 10g R1, the configuration of resource groups must ensure that Sun 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 CRS voting disk to depend on the other resource groups.

▼ How to Create a Resource for a Scalable Device Group

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 Sun Cluster 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-fwk-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 Register and Configure Resources for the Sun StorEdge QFS Metadata Server

Perform this task *only* if you are using the Sun StorEdge QFS shared file system.

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 StorEdge 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](#)” on page 231.

```
# clresourcegroup create -n nodelist \  
[-p rg_affinities==+scal-dg-rg \  
[-p rg_description="description" \  
qfs-mds-rg
```

- 3 **Register the `SUNW.qfs` resource type.**
- 4 **For each Sun StorEdge 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](#)” on page 231.

```
# 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 Create a Resource for a File-System Mount Point

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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” on page 231](#).

Tip – If you require Sun Cluster 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 StorEdge QFS shared file system, type the following command:**

Set a strong dependency by the instance of `SUNW.ScalMountPoint` on the resource for the Sun StorEdge QFS metadata server for the file system. The resource for the Sun StorEdge QFS metadata server set is created in [“How to Register and Configure Resources for the Sun StorEdge QFS Metadata Server” on page 232](#).

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” on page 231](#).

```
# 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 mountpointdir=mp-path \  
-p mountpointdir=mp-path \  
-p mountpointdir=mp-path \  
-p mountpointdir=mp-path
```

```
-p filesystemtype=s-qfs \  
-p targetfilesystem=fs-name qfs-mp-rs
```

- For each file system on a Network Appliance 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](#)” on page 231.

```
# 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
```

Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands

The task in this section is an alternative for the resource-configuration steps in “[How to Enable Sun Cluster and Oracle 10g R2 CRS to Interoperate](#)” on page 84

Resources for interoperation with Oracle 10g R2 enable you to administer RAC database instances by using Sun Cluster interfaces. These resources also ensure that dependencies by Oracle CRS resources on Sun Cluster resources are met. These resources enable the high-availability frameworks that are provided by Sun Cluster and Oracle CRS to interoperate.

The following resources for interoperation with Oracle 10g are required:

- A Sun Cluster resource to act as a proxy for the Oracle RAC database
- A Sun Cluster resource to represent the Oracle CRS framework
- Oracle CRS resources to represent scalable device groups
- Oracle CRS resources to represent scalable file-system mount points

You must assign to an Oracle CRS resource that represents a Sun Cluster resource a name in the following form:

```
sun.node.sc-rs
```

node Specifies the name of the node where the Oracle CRS resource is to run.

sc-rs Specifies the name of the Sun Cluster resource that the Oracle CRS resource represents.

For example, the name of the Oracle CRS resource for node `pc1us1` that represents the Sun Cluster resource `scal-dg-rs` must be as follows:

```
sun.pc1us1.scal-dg-rs
```

▼ How to Create Sun Cluster Resources for

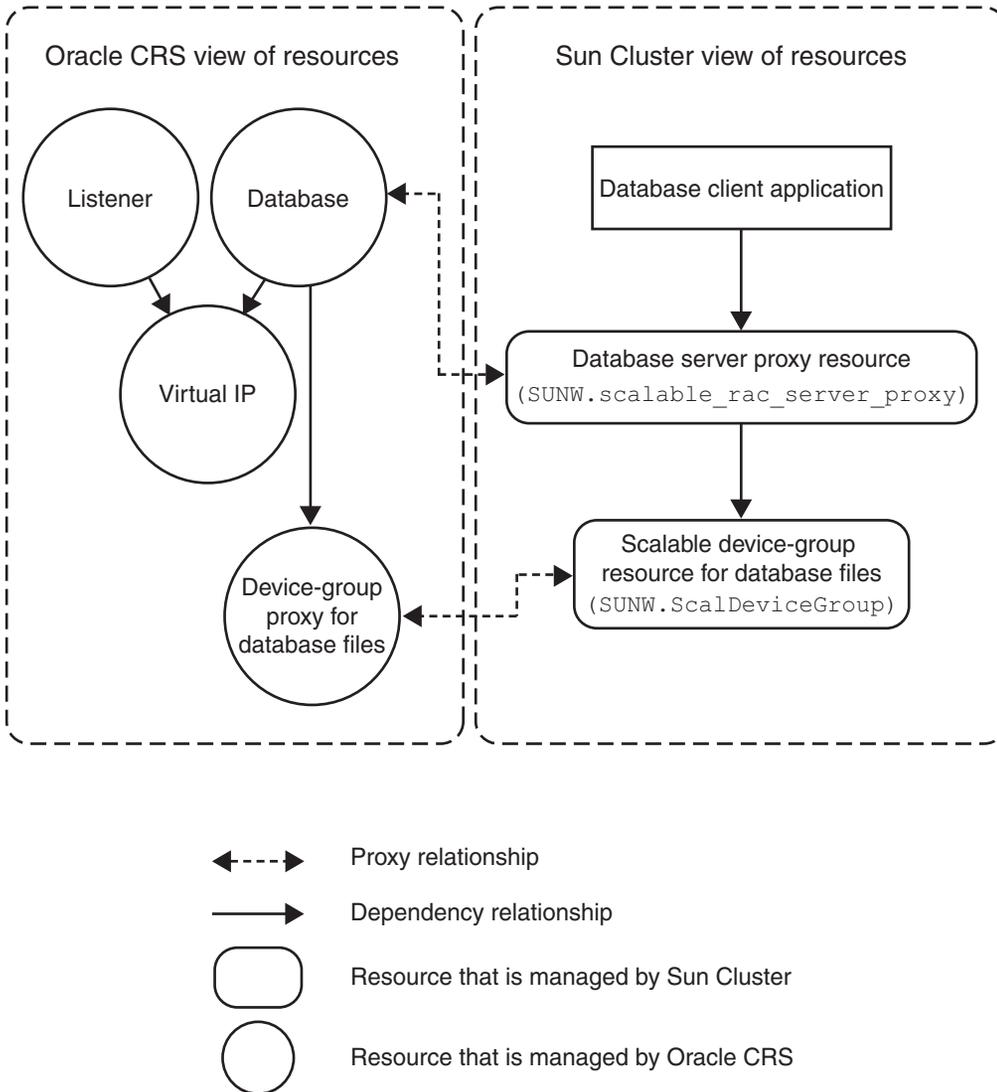


FIGURE D-1 Proxy Resources for Configurations With a Volume Manager

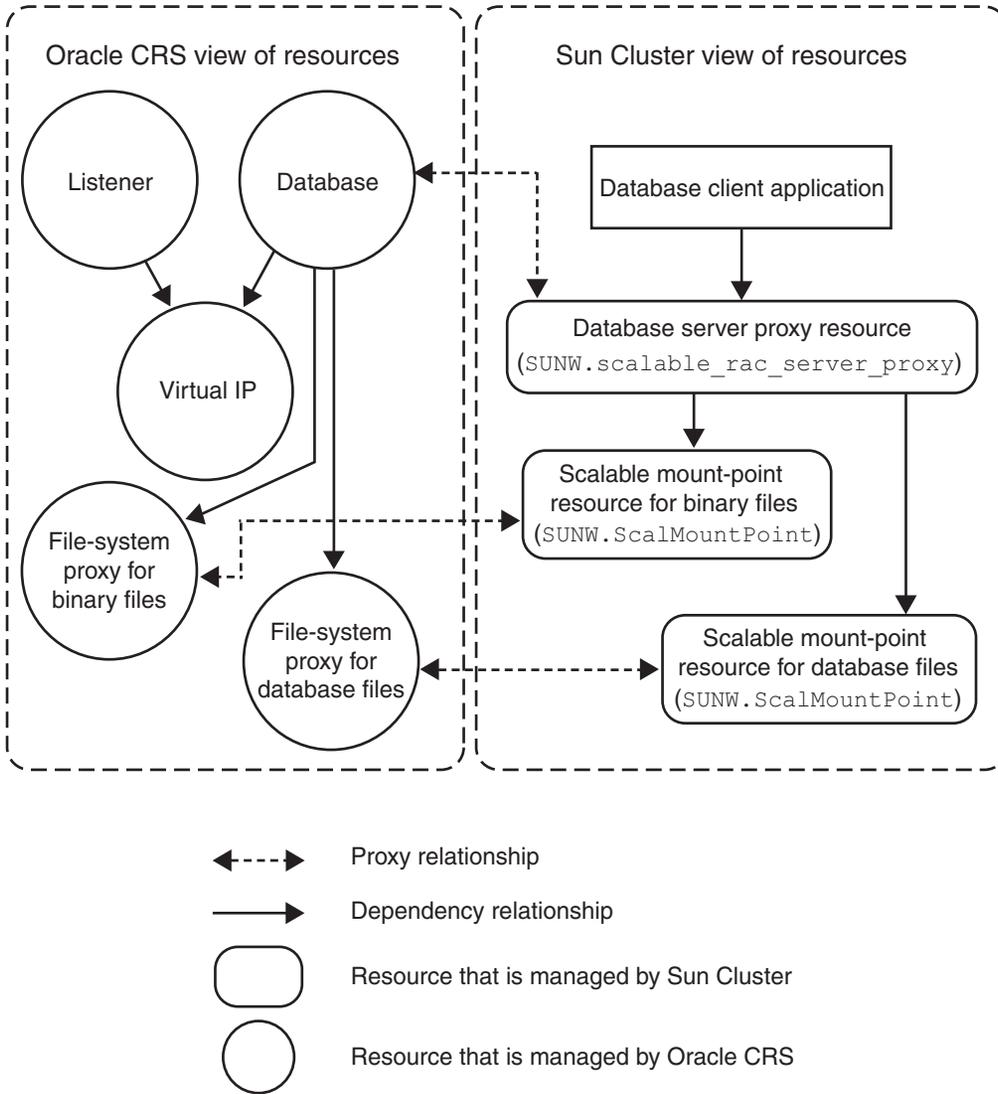


FIGURE D-2 Proxy Resources for Configurations With a Shared File System

Interoperation With Oracle 10g

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 53](#).

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” on page 231](#).
- 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” on page 233](#).

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” on page 233](#).

```
# 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” on page 231](#).
- 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” on page 233](#).

Tip – If you require Sun Cluster 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-fwk-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” on page 231](#).
- 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” on page 233](#).

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 an Oracle CRS Resource for Interoperation With Sun Cluster

Oracle CRS resources are similar to Sun Cluster resources. Oracle CRS resources represent items that the CRS manage in a similar way to how Sun Cluster resources represent items that the Sun Cluster RGM manages.

Depending on your configuration, some Oracle components that are represented as CRS resources might depend on file systems and global devices that Sun 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 CRS resource for each Sun Cluster resource for scalable device groups and scalable file-system mount points on which Oracle components depend. The Oracle CRS resources that you create track the status of their associated Sun Cluster resources. The Oracle CRS resources also ensure the orderly startup of Oracle CRS resources.

Perform this task on each cluster node where Sun Cluster Support for Oracle RAC is to run.

Note – Some steps in this procedure require you to use Oracle CRS commands. In these steps, the syntax of the command for Oracle release 10g R2 is provided. If you are using a version of Oracle other than 10g R2, see your Oracle documentation for the correct command syntax.

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

2 If the `/var/cluster/ucmm/profile` directory does not exist, create it.

Profiles for CRS resources are created in this directory.

```
# mkdir -p /var/cluster/ucmm/profile
```

3 Create a profile for the Oracle CRS 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 CRS 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 Sun Cluster resource for which the Oracle CRS resource is a proxy is online.**a. Obtain the state of the Sun Cluster resource.**

```
# clresource status sc-rs
```

b. If the state of the Sun Cluster resource is *not* online, bring online the resource group that contains the Sun Cluster resource.

If the state of the Sun Cluster resource is online, omit this step.

```
# clresourcegroup -emM sc-rg
```

6 Start the Oracle CRS resource that you registered in [Step 4](#).

```
# crs-home/bin/crs_start sun.node.sc-rs
```

7 Add the Oracle CRS resource that you registered in [Step 4](#) to the list of resources that the dependent Oracle CRS resource requires.**a. If the dependent Oracle CRS 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 CRS resource requires.

```
# crs-home/bin/crs_stat -p depend-crs-rs | grep REQUIRED_RESOURCES
```

c. Append the name of the Oracle CRS 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"
```

Registering and Configuring Sun Cluster Resources for Interoperation With Oracle 9i by Using Sun 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 9i RAC Database Instances”](#) on page 89.

Resources for interoperation with Oracle 9i enable you to administer RAC database instances by using Sun 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 9i are required:

- Oracle RAC server resources
- Oracle listener resources
- Logical hostname resources

Oracle 9i RAC Server Resources

Note – If you are using Oracle 10g, no Oracle RAC server resources are required. For more information, see [“Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands”](#) on page 235.

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 9i Listener Resources

Note – If you are using Oracle 10g, no Oracle listener resources are required. For more information, see [“Creating Resources for Interoperation With Oracle 10g by Using Sun Cluster Maintenance Commands”](#) on page 235.

If your configuration of Oracle RAC requires Oracle listeners, configure each listener to serve only one 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 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 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 9i Listener Resources

Note – If you are using Oracle 10g, 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 Sun 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 Sun Cluster Resources for Interoperation With Oracle 9i

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

This procedure provides the long forms of the Sun Cluster maintenance commands. Most commands also have short forms. Except for the forms of the command names, the commands are identical. For a list of the commands and their short forms, see Appendix A, “Sun Cluster Object-Oriented Commands,” in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

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 Sun Cluster 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 Sun Cluster 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
```

<code>-n nodelist</code>	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.
<code>-p rg_description="description"</code>	Specifies an optional brief description of the resource group. This description is displayed when you use Sun Cluster maintenance commands to obtain information about the resource group.
<code>lh-name-rg</code>	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
```

<code>-h lh-name</code>	Specifies the logical hostname that this resource is to make available. An entry for this logical hostname must exist in the name service database.
<code>-g lh-name-rg</code>	Specifies that you are adding the resource to the resource group that you created in Step 2 .
<code>lh-name-rs</code>	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-fwk-rg \
[-p rg_description="description" \]
-p rg_mode=Scalable rac-db-rg
```

- n *nodelist*
Specifies a comma-separated list of cluster nodes on which Sun Cluster Support for Oracle RAC is to be enabled. The Sun Cluster 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 Sun Cluster 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 Sun Cluster Support for Oracle RAC is to be enabled. This number must equal the number of nodes in *nodelist*.
 - p *rg_affinities=++rac-fwk-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 Sun 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 StorEdge QFS shared file system or a Network Appliance 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 65.

`-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.

```
# clresource_type 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-fwk-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-fwk-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 Sun Cluster Manager, this resource is named `rac-framework-rs`.

If you are using a volume manager or the Sun StorEdge QFS shared file system for database files, you must also specify the storage resource for database files.

If you are using the Sun StorEdge 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 65](#).

-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 Sun Cluster Resources for Interoperation With Oracle 9i

This example shows the sequence of operations that is required to register and configure Sun Cluster resources for interoperation with Oracle 9i on a two-node cluster.

The example makes the following assumptions:

- Sun Cluster 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
```

- 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
```

- To register the SUNW.scalable_rac_listener resource type, the following command is run:

```
# clresourcetype register SUNW.scalable_rac_listener
```

- 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.

- To register the SUNW.scalable_rac_server resource type, the following command is run:

```
# clresourcetype register SUNW.scalable_rac_server
```

- 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 Sun Cluster Support for Oracle RAC”](#) on page 95.

Index

Numbers and Symbols

32-bit mode, 24

64-bit mode, 24

A

abort step timeout

Oracle UDLM, 203

Solaris Volume Manager for Sun Cluster, 205

VxVM, 207

action file, *See* custom action file

ACTION keyword, 120

actions

file-system mount-point fault monitor, 114-115

listener fault monitor, 117

preset for fault monitor, 193-200

scalable device-group fault monitor, 114

server fault monitor

definition, 116

modifying, 120

adding

messages to log files, 121

resources to RAC framework resource group, 165-167

Sun Cluster Support for Oracle RAC to nodes, 158-164

volume manager resources, 165-167

volumes to be monitored, 148

administering, Sun Cluster Support for Oracle RAC, 105-127

affinities

examples

Oracle 10g, 186

Oracle 9i, 188

file-system mount-point resource groups, 234

RAC server resource groups

Oracle 10g, 238

Oracle 9i, 244

scalable device-group resource groups, 232

Sun StorEdge QFS resource groups, 233

alert files

in cluster file system, 51

file-system options

Sun StorEdge QFS shared file system, 47

UNIX file system, 52

storage management schemes, 22

in Sun StorEdge QFS shared file system, 45-46

alert_log_file extension property, 210

alert logs

modifying response to errors, 123-124

use of by fault monitor, 117

apache directory, 76

architecture requirements, processors, 24

archived redo log files

in cluster file system, 51

file-system options

Sun StorEdge QFS shared file system, 47

UNIX file system, 52

optimum I/O performance, 52

storage management schemes, 22

in Sun StorEdge QFS shared file system, 46

arrays

- disk, 43-45
 - See also* redundant array of independent disks (RAID)

ASM (Automatic Storage Management)

- configuring, 49-51
- extension properties, 201-202
- installing, 49-51
- Oracle file types supported, 22
- reservation step timeout, 111
- task summary, 39-40

ASM_DISKSTRING parameter, 50

Automatic Storage Management (ASM)

- configuring, 49-51
- extension properties, 201-202
- installing, 49-51
- Oracle file types supported, 22
- reservation step timeout, 111
- task summary, 39-40

automating

- database startup and shutdown for Oracle 10g
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 237-240
- database startup and shutdown for Oracle 9i
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 242-251

avm_stop_step_timeout extension property, 207

B

binary files

- in cluster file system, 51
- determining location, 23-24
- file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
- storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 45-46

boot, verifying, 102-103

C

caution notice

- adding resources to RAC framework resource group, 165
- Oracle UDLM, 33
- rebooting nodes, 31
- server fault monitor customizations, 118
- Sun Cluster Support for Oracle RAC verification, 102
- VxVM, 42

changing, *See* modifyingchecking, *See* verifying

client_retries extension property, 216

client_retry_interval extension property, 216

clnode command, 18

clsetup utility

- comparison with Sun Cluster maintenance commands, 53, 66, 83
- CRS resources, 84-89
- database resources
 - Oracle 10g, 84-89
 - Oracle 9i, 89-94
- names of resources created by, 106-107
- proxy resources, 84-89
- RAC framework resources, 53-57
- storage resources, 66-71

cluster file system

- creating, 51-53
- installation of Oracle files, 24
- mounting, 51-53
- node-specific directories, 76-78
- node-specific files, 79-81
- Oracle file types supported, 22
- requirements, 51
- task summary, 41

Cluster Ready Services (CRS)

- extension properties, 219
- file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
- installing on subset of nodes, 75
- overriding networking defaults, 74
- resources
 - creating with `clsetup` utility, 84-89

- Cluster Ready Services (CRS), resources (*Continued*)
 - creating with Sun Cluster maintenance
 - commands, 240-241
 - state changes, 107-109
 - storage management schemes, 22
- commands
 - license verification, 21
 - node information, 18
- communications ports
 - Oracle UDLM
 - extension properties, 202
 - guidelines for setting, 111-112
 - VxVM, 209
- components, validation failures, 141
- concatenating, slices, 59
- configuration daemon, VxVM, 209
- configuration files
 - in cluster file system, 51
 - determining location, 23-24
 - file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
 - Oracle DLM, 202
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 45-46
- configuration information, UCMM, 139
- configuring
 - ASM, 49-51
 - DID, 43-45, 49-51
 - file-system mount-point resources
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance
 - commands, 233-235
 - hardware RAID, 43-45
 - listener
 - planning, 242-243
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance
 - commands, 244-251
 - Network Appliance NAS devices, 41
 - Oracle 10g RAC server
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance
 - commands, 237-240
 - configuring (*Continued*)
 - Oracle 9i RAC listener
 - with `clsetup` utility, 89-94
 - Oracle 9i RAC server
 - planning, 242
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance
 - commands, 242-251
 - RAC framework resource group
 - overview, 53
 - for clusters, 53-57, 226-229
 - planning, 226
 - scalable device-group resources
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance
 - commands, 231-232
 - Solaris Volume Manager for Sun Cluster, 41-42
 - Sun Cluster Support for Oracle RAC
 - examples, 185-188
 - for selected nodes, 158-164
 - Sun StorEdge QFS resources
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance
 - commands, 232-233
 - UFS, 52
 - VxVM, 42
 - connect_ cycle extension property, 210
 - connect_ string extension property, 210
 - CONNECTION_STATE keyword, 120
 - control files
 - in cluster file system, 51
 - file-system options, 47
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
 - core files
 - fault monitors, 117-118
 - Oracle UDLM, 139
 - creating
 - cluster file system, 51-53
 - CRS resources
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance
 - commands, 240-241
 - device groups, 57-65

creating (*Continued*)

- file-system mount-point resources
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 233-235
- global device groups, 57-65
- listener resource
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 246
- listener resource group
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 245
- logical hostname resource groups
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 244
- LUN, 43-45
- multi-owner disk sets, 58-64
- Oracle 10g RAC server resource group
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 238
- Oracle 10g RAC server resources
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 239
- Oracle 9i RAC server resource group
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 245
- Oracle 9i RAC server resources
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 247
- proxy resources
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 240-241
- RAC framework resource group
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 226-229
- scalable device-group resources
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 231-232
- shared-disk groups, 64-65
- slices, 43

creating (*Continued*)

- Sun StorEdge QFS resources
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 232-233
- Sun StorEdge QFS shared file system, 47-48
- CRS (Cluster Ready Services)
 - extension properties, 219
 - file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
 - installing on subset of nodes, 75
 - overriding networking defaults, 74
 - resources
 - creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 240-241
 - state changes, 107-109
 - storage management schemes, 22
- `crs_framework` resource type
 - dependencies, 88, 238
 - extension properties, 219
 - instantiating
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 238
 - names of instances created by `clsetup`, 106
 - registering
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 238
- `crs_home` extension property, 216
- custom action file
 - format, 118-121
 - maximum number of entries in, 118
 - order of entries in, 123
 - propagating to cluster nodes, 126-127
 - specifying, 127
 - validating, 127
- `custom_action_file` extension property, 211
- customizing, server fault monitor, 118-127
- `cvm_abort_step_timeout` extension property, 207
- `cvm_return_step_timeout` extension property, 207
- `cvm_start_step_timeout` extension property, 207
- `cvm_step1_timeout` extension property, 208
- `cvm_step2_timeout` extension property, 208

- cvm_step3_timeout extension property, 208
- cvm_step4_timeout extension property,
 - definition, 208
- Cvm_step4_timeout extension property, guidelines for setting, 110
- cvm_stop_step_timeout extension property, 208
- cylinders, restrictions, 44

- D**
- daemons
 - ucmmd
 - failure to start, 141
 - unexpected termination, 141
 - vxconfigd, 209
 - vxkmsgd, 209
- data files
 - file-system options, 47
 - location on shared file system, 81-82
 - storage management schemes, 22
- Data Guard, 25
- database administrator (DBA)
 - creating, 27-30
 - granting access to file systems, 48
 - granting access to volumes, 60
- database management system (DBMS)
 - See also* relational database management system (RDBMS)
 - errors
 - modifying response to, 121-123
 - preset actions, 193-200
 - timeouts, 117-118
- databases
 - automating startup and shutdown for Oracle 10g
 - with clsetup utility, 84-89
 - with Sun Cluster maintenance commands, 237-240
 - automating startup and shutdown for Oracle 9i
 - with clsetup utility, 89-94
 - with Sun Cluster maintenance commands, 242-251
 - creating, 81-82
 - file systems for, 46
 - instance name, 212, 217
 - databases (*Continued*)
 - resources
 - creating with clsetup utility, 82-94
 - creating with Sun Cluster maintenance commands, 235-241
 - verifying
 - Oracle 10g resources, 98-101
 - Oracle 9i resources, 101-102
 - DAU (disk allocation unit), 48
 - db_name extension property, 216
 - DBA (database administrator)
 - creating, 27-30
 - granting access to file systems, 48
 - granting access to volumes, 60
 - dbca command, 81
 - DBMS (database management system)
 - See also* RDBMS (relational database management system)
 - errors
 - modifying response to, 121-123
 - preset actions, 193-200
 - timeouts, 117-118
 - debug_level extension property
 - scalable_rac_listener resource type, 214
 - scalable_rac_server_proxy resource type, 217
 - scalable_rac_server resource type, 211
 - ScalDeviceGroup resource type, 219
 - ScalMountPoint resource type, 221
 - SUNW.rac_svm resource type, 205
 - SUNW.scalable_rac_listener resource type, 214
 - SUNW.scalable_rac_server_proxy resource type, 217
 - SUNW.scalable_rac_server resource type, 211
 - SUNW.ScalDeviceGroup resource type, 219
 - SUNW.ScalMountPoint resource type, 221
 - Degraded - reconfiguration in progress message, 144
 - deleting
 - See* modifying
 - See* removing
 - See* uninstalling
 - dependencies
 - crs_framework resource type, 88, 238
 - LogicalHostname resource type, 94, 243

dependencies (*Continued*)

- offline-restart, 95
 - qfs resource type, 71, 233
 - rac_cvm resource type, 57, 229
 - rac_framework resource type, 56
 - rac_svm resource type, 57, 228
 - rac_udlm resource type, 57, 228
 - scalable_rac_listener resource type, 94, 243
 - scalable_rac_server_proxy resource type, 88, 239
 - scalable_rac_server resource type, 94, 244
 - ScalDeviceGroup resource type, 71, 232
 - ScalMountPoint resource type, 71, 234
 - SUNW.crs_framework resource type, 88, 238
 - SUNW.LogicalHostname resource type, 94, 243
 - SUNW.qfs resource type, 71, 233
 - SUNW.rac_cvm resource type, 57, 229
 - SUNW.rac_framework resource type, 56
 - SUNW.rac_svm resource type, 57, 228
 - SUNW.rac_udlm resource type, 57, 228
 - SUNW.scalable_rac_listener resource type, 94, 243
 - SUNW.scalable_rac_server_proxy resource type, 88, 239
 - SUNW.scalable_rac_server resource type, 94, 244
 - SUNW.ScalDeviceGroup resource type, 71, 232
 - SUNW.ScalMountPoint resource type, 71, 234
- device groups
- See also* multi-owner disk sets
 - See also* shared-disk groups
 - creating, 57-65
 - extension properties, 219-221
 - monitoring additional volumes, 148
 - optimum I/O performance, 52
 - resource group affinities, 232
 - restrictions, 64
 - storage resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
- device identity (DID)
- configuring, 43-45, 49-51

devices

- adding to multi-owner disk sets, 59
- raw, 43-45
 - See also* redundant array of independent disks (RAID)
 - removing from multi-owner disk sets, 170
- diagnostic information, 139-140
- DID (device identity)
 - configuring, 43-45, 49-51
- directories
 - node-specific, 76-78
 - Oracle home, 212, 215, 217
 - `/var/opt`, 29
- disabling, RAC server resources, 107-109
- disk allocation unit (DAU), 48
- disk sets, multi-owner, 58-64
- diskgroupname extension property, 219
- disks
 - arrays, 43-45
 - See also* redundant array of independent disks (RAID)
 - effect of number on reservation timeout, 111
 - installation of Oracle files, 23
 - Oracle file types supported, 22
 - slices, 43
 - soft partitions, 59
 - stripe width, 48
- distributed lock manager (DLM), 202
- DLM (distributed lock manager), 202

E

- editing, *See* modifying
- enabling
 - See also* starting
 - listener, 249
 - RAC server resource group
 - Oracle 10g, 240
 - Oracle 9i, 249
- environment variables, 213, 215, 218
- ERROR keyword, 119
- error messages, cluster, 18
- ERROR_TYPE keyword, 119

errors

- in custom action file, 127
- DBMS
 - modifying response to, 121-123
 - preset actions for, 193-200
- ignoring, 122-123
- responding to, 121-122
- SGA, 121
- timeouts
 - core-file creation, 117-118
 - modifying maximum allowed, 124-126
- types detected by fault monitor, 118
- /etc/group file, 27
- /etc/opt/SUNWsamfs/samfs.cmd file, 47
- /etc/passwd file, 28
- /etc/shadow file, 28
- /etc/system file, 31
- /etc/vfstab file
 - Sun StorEdge QFS, 47
 - UNIX file system, 52
- /etc/nsswitch.conf file, 26
- event logs, 139
- examples, configuration, 185-188
- executing, *See* starting
- extension properties
 - ASM, 201-202
 - crs_framework resource type, 219
 - hardware RAID, 201-202
 - Network Appliance network-attached storage (NAS)
 - devices, 201-202
 - Oracle UDLM, 202-205
 - rac_cvm resource type, 207-210
 - rac_framework resource type, 201-202
 - rac_svm resource type, 205-207
 - rac_udlm resource type, 202-205
 - RAID, 201-202
 - restrictions, 112
 - scalable_rac_listener resource type, 214-216
 - scalable_rac_server_proxy resource
 - type, 216-219
 - ScalDeviceGroup resource type, 219-221
 - ScalMountPoint resource type, 221-224
 - setting, 225
 - Solaris Volume Manager for Sun Cluster, 205-207

extension properties (*Continued*)

- Sun StorEdge QFS shared file system, 201-202
- SUNW.crs_framework resource type, 219
- SUNW.rac_cvm resource type, 207-210
- SUNW.rac_framework resource type, 201-202
- SUNW.rac_svm resource type, 205-207
- SUNW.rac_udlm resource type, 202-205
- SUNW.scalable_rac_listener resource
 - type, 214-216
- SUNW.scalable_rac_server_proxy resource
 - type, 216-219
- SUNW.scalable_rac_server resource
 - type, 210-214
- SUNW.ScalDeviceGroup resource type, 219-221
- SUNW.ScalMountPoint resource type, 221-224
- VxVM, 207-210

F

- Failfast: Aborting because "ucmmd" died
 - message, 141
- failfastmode extension property, 202
- failover
 - IP, 144
 - LogicalHostname resource, 243
- failures
 - component validations, 141
 - nodes
 - network timeouts, 243
 - reconnection after, 144
 - recovery actions, 144
 - panics
 - during initialization, 141
 - during installation, 31
 - timeouts, 143-144
 - public networks, 26-27
 - rac_framework resource
 - startup, 144-146
 - reconfiguration of resources, 144
 - START method timeout, 145
 - startup of resources, 144-146
 - stopping of resources, 146
 - SUNW.rac_framework resource, 144-146

failures (*Continued*)

- ucmmd daemon
 - startup, 141
 - unexpected termination, 141
- fault monitoring, 20
- fault monitors
 - actions
 - file-system mount-point fault monitor, 114-115
 - listener fault monitor, 117
 - modifying, 120
 - scalable device-group fault monitor, 114
 - server fault monitor, 116
 - alert logs, 117
 - caution notice, 118
 - core-file creation, 117-118
 - customizing, 118-127
 - device groups
 - resource type for, 113
 - error types detected by, 118
 - file systems
 - resource type for, 113
 - mount points
 - resource type for, 113
 - Oracle RAC listener
 - resource type for, 113
 - Oracle RAC server
 - resource type for, 113
 - preset actions, 193-200
 - tuning, 113-118
- Faulted - ucmmd is not running message, 144
- file-system mount-point fault monitor, 114-115
- file-system mount points
 - resource groups for
 - affinities, 234
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 233-235
 - planning, 230
 - resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 233-235
 - planning, 230
 - state information, 139

file-system mount points, resources for (*Continued*)

- `syslog()` function, 139
- file systems
 - See also* cluster file system
 - See also* Sun StorEdge QFS shared file system
 - extension properties, 221-224
 - installation of Oracle files, 24
 - restrictions, 58, 64
 - storage resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
 - UFS options, 52
- files
 - alert log
 - modifying response to errors in, 123-124
 - use of by fault monitor, 117
 - core
 - fault monitors, 117-118
 - Oracle UDLM, 139
 - custom action
 - format, 118-121
 - order of entries in, 123
 - propagating to cluster nodes, 126-127
 - specifying, 127
 - validating, 127
 - determining location, 21-24
 - diagnostic information, 139
 - `/etc/group`, 27
 - `/etc/opt/SUNWsamfs/samfs.cmd`, 47
 - `/etc/passwd`, 28
 - `/etc/shadow`, 28
 - `/etc/system`, 31
 - `/etc/vfstab`
 - Sun StorEdge QFS, 47
 - UNIX file system, 52
 - `/etc/nsswitch.conf`, 26
 - group, 27
 - node-specific, 79-81
 - `nsswitch.conf`, 26
 - Oracle configuration, 23-24
 - Oracle RAC, 45, 51
 - `passwd`, 28

files (*Continued*)

- samfs.cmd, 47
- shadow, 28
- storage management schemes for, 21-24
- Sun Cluster Support for Oracle RAC logs
 - additional messages in, 121
- system, 31
- vfstab
 - Sun StorEdge QFS, 47
 - UNIX file system, 52
- filesystemtype extension property, 221
- flashback log files
 - in cluster file system, 51
 - file-system options, 52
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
- fragmentation, of memory, 121
- framework, *See* Real Application Clusters (RAC)
 - framework resource group

G

- gen use type, volumes, 65
- global device groups
 - See also* multi-owner disk sets
 - See also* shared-disk groups
 - creating, 57-65
 - monitoring additional volumes, 148
 - storage resources for
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 231-232
 - planning, 230
- global devices
 - adding to multi-owner disk sets, 59
 - removing from multi-owner disk sets, 170
- global zone, 20
- group database, nsswitch.conf file, 26
- group file, 27
- groups, creating, 27-30
- Guard option, Oracle RAC, 25

H

- hardware redundant array of independent disks (RAID)
 - configuring, 43-45
 - extension properties, 201-202
 - installing, 43-45
 - Oracle file types supported, 22
 - reservation step timeout, 111
 - task summary, 39
- hardware requirements, 21
- heap memory, 122
- help, 18
- home directory
 - Oracle, 212, 215, 217
- hostnames, restrictions, 25

I

- I/O (input/output) performance, 52
- identifiers
 - system, 212, 217
 - user, 26
- ignoring, minor errors, 122-123
- input/output (I/O) performance, 52
- installing
 - ASM, 49-51
 - hardware RAID, 43-45
 - Network Appliance NAS devices, 41
 - Oracle RAC
 - overview, 74-75
 - verifying installation, 81
 - Oracle UDLM, 33-34
 - Solaris Volume Manager for Sun Cluster, 41-42
 - storage management software, 41-53
 - Sun Cluster Support for Oracle RAC, 31-33
 - verifying installation, 95-103
 - VxVM, 42
- insufficient memory errors, 121, 122
- interfaces, network, 74
- Internet Protocol (IP), 144
- io timeout extension property, 222
- IP (Internet Protocol), 144

K

kernel parameters, 30
keywords, custom action file, 119

L

license requirements, 21
listener
 extension properties, 214-216
 resource groups
 creating with `clsetup` utility, 89-94
 creating with Sun Cluster maintenance
 commands, 244-251
 planning, 242-243
 resource groups for
 enabling, 249
 resources
 creating with `clsetup` utility, 89-94
 creating with Sun Cluster maintenance
 commands, 246
listener fault monitor, 117
listener_name extension property
 scalable_rac_listener resource type, 214
 SUNW.scalable_rac_listener resource type, 214
load, effect on reservation timeout, 111
local disks
 installation of Oracle files, 23
 Oracle file types supported, 22
location
 diagnostic information files, 139
 log files, 139-140
 Oracle files, 21-24
 redo log files, 52
log files
 locations, 139-140
 RAC listener, 214
 RAC server, 211
 RAC server proxy, 217
 Sun Cluster Support for Oracle RAC
 additional messages in, 121
 use in troubleshooting, 139
logged alerts
 modifying response to, 123-124
 use of by fault monitor, 117

logical unit number (LUN), creating, 43-45
logicaldeviceList extension property, 220
LogicalHostname resource type
 dependencies, 94, 243
 instantiating
 with `clsetup` utility, 89-94
 with Sun Cluster maintenance commands, 245
 names of instances created by `clsetup`, 107
 planning, 243-244
LUN (logical unit number), creating, 43-45

M

maximum values
 entries in custom action file, 118
 number of allowed timeouts, 124-126
memory
 shared, 30-31
 shortage of, 121, 122
MESSAGE keyword, 121
messages
 debug, 211, 214, 217
 panics, 141
 startup failure, 144
messages file, 18
messaging daemon, VxVM, 209
metadata server
 resource groups for
 creating with `clsetup` utility, 66-71
 creating with Sun Cluster maintenance
 commands, 232-233
 planning, 230
 resources for
 creating with `clsetup` utility, 66-71
 creating with Sun Cluster maintenance
 commands, 232-233
 planning, 230
mirrored devices, adding to multi-owner disk sets, 59
modifying
 extension properties
 command for, 110
 tunable when disabled, 112-113
 number of allowed timeouts, 124-126
 response to DBMS errors, 121-123

- modifying (*Continued*)
 - response to logged alerts, 123-124
 - server fault monitor actions, 120
 - Sun Cluster Support for Oracle RAC
 - by removing nodes from, 173-184
 - monitor_probe_interval extension property, 217
 - monitor_retry_count extension property
 - ScalDeviceGroup resource type, 220
 - ScalMountPoint resource type, 222
 - SUNW.ScalDeviceGroup resource type, 220
 - SUNW.ScalMountPoint resource type, 222
 - monitor_retry_interval extension property
 - ScalDeviceGroup resource type, 221
 - ScalMountPoint resource type, 222
 - SUNW.ScalDeviceGroup resource type, 221
 - SUNW.ScalMountPoint resource type, 222
 - monitoring, Sun Cluster Support for Oracle RAC, 129
 - mount options, UFS, 52
 - mount points, extension properties, 221-224
 - mounting
 - cluster file system, 51-53
 - Sun StorEdge QFS shared file system, 47-48
 - mountoptions extension property, 222
 - mountpointdir extension property, 223
 - multi-owner disk sets
 - creating, 58-64
 - storage resources for
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
- N**
- name service
 - bypassing, 26-27
 - database user entries in, 27
 - names
 - Oracle database instance, 212, 217
 - NAS devices, *See* Network Appliance network-attached storage (NAS) devices
 - network/agent directory, 76
 - Network Appliance network-attached storage (NAS) devices
 - configuring, 41
 - extension properties, 201-202
 - installing, 41
 - Oracle file types supported, 22
 - storage resources for
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
 - task summary, 40-41
 - Network Information Service (NIS)
 - bypassing, 26-27
 - database user entries in, 27
 - network interfaces, 74
 - network/log directory, 76
 - network timeouts, effect on node failure detection, 243
 - network/trace directory, 76
 - networks
 - private, 74
 - public
 - failures, 26-27
 - installation options for Oracle 10g, 74
 - NEW_STATE keyword, 120
 - NIS (Network Information Service)
 - bypassing, 26-27
 - database user entries in, 27
 - nodes
 - adding Sun Cluster Support for Oracle RAC to, 158-164
 - caution notice, 31
 - directories specific to, 76-78
 - failure
 - public networks, 26-27
 - failures
 - network timeouts, 243
 - reconnection after, 144
 - recovery actions, 144
 - files specific to, 79-81
 - installing CRS on subset, 75
 - network timeouts, 243
 - panics
 - during initialization, 141

- nodes, panics (*Continued*)
 - during installation, 31
 - timeouts, 143-144
- preparing for Oracle RAC, 26-31
- rebooting, 31
- reconnection after failure, 144
- removing Sun Cluster Support for Oracle RAC
 - from, 173-184
- uninstalling
 - Sun Cluster Support for Oracle RAC software packages from, 180
- non-global zones, 20
- nsswitch.conf file, 26
- num_ports extension property, definition, 202
- Num_ports extension property, guidelines for setting, 111

O

- OCR (Oracle cluster registry) files
 - in cluster file system, 51
 - file-system options, 52
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
- offline-restart dependencies, 95
- oinstall group, 27
- Online message, 145
- online redo log files
 - in cluster file system, 51
 - file-system options, 47
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
- oper group, 27
- OPS/RAC, *See* Oracle RAC
- Oracle, error numbers, 193-200
- Oracle 10g
 - configuring after Sun Cluster upgrade, 154
 - database resources
 - creating with clsetup utility, 84-89
 - extension properties, 216-219
 - databases
 - administering with Sun Cluster, 107-109
 - creating, 81-82
 - location of data files, 82

- Oracle 10g (*Continued*)
 - log files, 140
 - overriding networking defaults, 74
 - resource group affinities, 238
 - sample configurations, 186
 - verifying
 - database resources, 98-101
 - installation, 81
- Oracle 9i
 - database resources
 - creating with clsetup utility, 89-94
 - extension properties, 210-214
 - databases
 - administering with Sun Cluster, 109
 - creating, 81-82
 - location of data files, 82
 - log files, 140
 - resource group affinities, 244
 - sample configurations, 188
 - upgrading database resources, 155-157
 - use of older resource types, 149
 - verifying
 - database resources, 101-102
 - installation, 81
- Oracle binary files, location, 23-24
- Oracle cluster registry (OCR) files
 - in cluster file system, 51
 - file-system options, 52
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
- oracle_config_file extension property, 202
- Oracle configuration files, location, 23-24
- Oracle Data Guard, 25
- Oracle distributed lock manager (DLM), 202
- Oracle files
 - binary file location, 23-24
 - configuration file location, 23-24
 - disks, 23
 - local disks, 23
 - storage management schemes for, 21-24
 - storage resources for
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232

- Oracle files, storage resources for (*Continued*)
 - planning, 230
 - Sun StorEdge QFS shared file system, 24
 - oracle_home extension property
 - scalable_rac_listener resource type, 215
 - scalable_rac_server_proxy resource type, 217
 - scalable_rac_server resource type, 212
 - SUNW.scalable_rac_listener resource type, 215
 - SUNW.scalable_rac_server_proxy resource type, 217
 - SUNW.scalable_rac_server resource type, 212
 - Oracle Parallel Server
 - See also* Oracle RAC
 - equivalence to Oracle RAC, 20
 - incompatibility with Solaris Volume Manager for Sun Cluster, 23
 - Oracle RAC
 - overview, 20
 - 32-bit mode, 24
 - 64-bit mode, 24
 - Data Guard, 25
 - files related to, 45
 - Guard option, 25
 - installing
 - overview, 74-75
 - verifying installation, 81
 - log file locations, 139-140
 - multi-owner disk sets for, 58-64
 - preparing nodes for, 26-31
 - shared-disk groups for, 64-65
 - verifying installation and configuration, 81
 - oracle_rac_listener resource type, 149
 - oracle_rac_server resource type, 149
 - Oracle RDBMS (relational database management system)
 - file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
 - processor architecture requirements, 24
 - storage management schemes, 22
 - Oracle relational database management system (RDBMS)
 - file-system options
 - Sun StorEdge QFS shared file system, 47
 - Oracle relational database management system (RDBMS), file-system options (*Continued*)
 - UNIX file system, 52
 - processor architecture requirements, 24
 - storage management schemes, 22
 - oracle_sid extension property
 - scalable_rac_server_proxy resource type, 217
 - scalable_rac_server resource type, 212
 - SUNW.scalable_rac_server_proxy resource type, 217
 - SUNW.scalable_rac_server resource type, 212
 - Oracle UDLM (UNIX Distributed Lock Manager)
 - caution notice, 33
 - communications ports
 - extension properties, 202
 - guidelines for setting, 111-112
 - core files, 139
 - effect on tunable when disabled extension properties, 112
 - event logs, 139
 - extension properties, 202-205
 - installing, 33-34
 - log files, 140
 - processor architecture requirements, 24
 - resource type for, 226
 - restrictions, 112
 - oracle user, 27
 - granting access to file systems, 48
 - granting access to volumes, 60
 - order, entries in custom action file, 123
 - overriding, server fault monitor presets, 118-127
 - overview
 - Oracle RAC, 20
 - RAC framework resource group, 226
- P**
- packages, 31-33
 - panics
 - during initialization, 141
 - during installation, 31
 - timeouts, 143-144
 - parameter_file extension property
 - scalable_rac_server resource type, 212

parameter_file extension property (*Continued*)
 SUNW.scalable_rac_server resource type, 212

partitions
 restrictions, 44
 soft, 59

passwd file, 28

patch requirements, 21

port extension property, definition, 202

Port extension property, guidelines for setting, 111

ports, *See* communications ports

prerequisites, *See* requirements

preset actions, fault monitor, 193-200

preventing
 unnecessary restarts
 for DBMS errors, 122-123
 for timeouts, 124-126

primary groups, 27

private network interfaces, 74

probe_timeout extension property
 scalable_rac_listener resource type, 215
 scalable_rac_server resource type, 213
 SUNW.scalable_rac_listener resource type, 215
 SUNW.scalable_rac_server resource type, 213

processor architecture requirements, 24

programs, vxclust, 209

proxy_probe_timeout extension property, 218

proxy resources
 creating with clsetup utility, 84-89
 creating with Sun Cluster maintenance
 commands, 240-241
 examples, 236

prtconf -v command, 18

prtdiag -v command, 18

psrinfo -v command, 18

public networks
 failure, 26-27
 installation options for Oracle 10g, 74

Q

QFS file system, *See* Sun StorEdge QFS shared file
system

qfs resource type
 dependencies, 71, 233

qfs resource type (*Continued*)
 instantiating
 with clsetup utility, 66-71
 with Sun Cluster maintenance commands, 233
 names of instances created by clsetup, 106
 registering
 with clsetup utility, 66-71
 with Sun Cluster maintenance commands, 233

R

RAC (Real Application Clusters) framework resource
group
 overview, 226
 adding resources to, 165-167
 configuring
 overview, 53
 for clusters, 53-57, 226
 creating
 with clsetup utility, 53-57
 with Sun Cluster maintenance
 commands, 226-229
 planning, 226
 upgrading resources in, 149-153
 verifying, 95-96

RAC (Real Application Clusters) server
configuring for Oracle 10g
 with clsetup utility, 84-89
 with Sun Cluster maintenance
 commands, 237-240
configuring for Oracle 9i
 planning, 242
 with clsetup utility, 89-94
 with Sun Cluster maintenance
 commands, 242-251

log files, 140

resource groups for Oracle 10g
 creating with clsetup utility, 84-89
 creating with Sun Cluster maintenance
 commands, 238
 enabling, 240

resource groups for Oracle 9i
 creating with clsetup utility, 89-94

- RAC (Real Application Clusters) server, resource groups for Oracle 9i (*Continued*)
- creating with Sun Cluster maintenance commands, 245
 - enabling, 249
 - planning, 242
- resources for Oracle 10g
- creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 239
 - disabling, 107-109
 - enabling, 240
- resources for Oracle 9i
- creating with `clsetup` utility, 89-94
 - creating with Sun Cluster maintenance commands, 247
 - disabling, 109
 - enabling, 249
 - planning, 242
- `rac_cvm` resource type
- dependencies, 57, 229
 - extension properties, 207-210
 - instantiating
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 229
 - names of instances created by `clsetup`, 106
 - purpose, 226
 - registering
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 229
 - restrictions, 112
 - upgrading, 149-153
- `rac_framework` resource type
- dependencies, 56
 - extension properties, 201-202
 - failure of instance to start, 144-146
 - instantiating
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 227
 - monitoring instances of, 129
 - names of instances created by `clsetup`, 106
 - purpose, 226
 - registering
 - with `clsetup` utility, 53-57
- `rac_framework` resource type, registering (*Continued*)
- with Sun Cluster maintenance commands, 227
 - START method timeout, 145
 - upgrading, 149-153
- `rac_hwraid` resource type
- obsolescence, 149
 - removing instance of, 153
- `rac_svm` resource type
- dependencies, 57, 228
 - extension properties, 205-207
 - instantiating
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 228
 - names of instances created by `clsetup`, 106
 - purpose, 226
 - registering
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 228
 - upgrading, 149-153
- `rac_udlm` resource type
- dependencies, 57, 228
 - extension properties, 202-205
 - instantiating
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 228
 - names of instances created by `clsetup`, 106
 - purpose, 226
 - registering
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 228
 - restrictions, 112
 - upgrading, 149-153
- RAID (redundant array of independent disks)
- configuring, 43-45
 - extension properties, 201-202
 - installing, 43-45
 - Oracle file types supported, 22
 - reservation step timeout, 111
 - task summary, 39
- raw devices, 43-45
- See also* redundant array of independent disks (RAID)
- RDBMS (relational database management system)
- See also* DBMS (database management system)

RDBMS (relational database management system)

(Continued)

- file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
- processor architecture requirements, 24
- storage management schemes, 22
- rdbms/audit directory, 76
- rdbms/log directory, 76
- Real Application Clusters (RAC) framework resource group
 - overview, 226
 - adding resources to, 165-167
 - configuring
 - overview, 53
 - for clusters, 53-57, 226
 - creating
 - with `clsetup` utility, 53-57
 - with Sun Cluster maintenance commands, 226-229
 - planning, 226
 - upgrading resources in, 149-153
 - verifying, 95-96
- Real Application Clusters (RAC) server
 - configuring for Oracle 10g
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 237-240
 - configuring for Oracle 9i
 - planning, 242
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 242-251
 - log files, 140
 - resource groups for Oracle 10g
 - creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 238
 - enabling, 240
 - resource groups for Oracle 9i
 - creating with `clsetup` utility, 89-94
 - creating with Sun Cluster maintenance commands, 245
 - enabling, 249

Real Application Clusters (RAC) server, resource groups for Oracle 9i *(Continued)*

- planning, 242
- resources for Oracle 10g
 - creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 239
 - disabling, 107-109
 - enabling, 240
- resources for Oracle 9i
 - creating with `clsetup` utility, 89-94
 - creating with Sun Cluster maintenance commands, 247
 - disabling, 109
 - enabling, 249
 - planning, 242
- reconfiguration, failures, 144
- reconfiguration timeouts
 - ASM, 201
 - Oracle UDLM, 203
 - reservation step, 201
 - Solaris Volume Manager for Sun Cluster definition, 205
 - VxVM
 - definition, 207
 - reconfiguration step 4, 110
- recovery files
 - in cluster file system, 51
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
- redo log files
 - See* archived redo log files
 - See* online redo log files
- redundant array of independent disks (RAID)
 - configuring, 43-45
 - extension properties, 201-202
 - installing, 43-45
 - Oracle file types supported, 22
 - reservation step timeout, 111
 - task summary, 39
- registering
 - `crs_framework` resource type
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 238

registering (*Continued*)

- qfs resource type
 - with clsetup utility, 66-71
 - with Sun Cluster maintenance commands, 233
- rac_cvm resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 229
- rac_framework resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 227
- rac_svm resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 228
- rac_udlm resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 228
- scalable_rac_listener resource type
 - with clsetup utility, 89-94
 - with Sun Cluster maintenance commands, 246
- scalable_rac_server_proxy resource type
 - with clsetup utility, 84-89
 - with Sun Cluster maintenance commands, 239
- scalable_rac_server resource type
 - with clsetup utility, 89-94
 - with Sun Cluster maintenance commands, 247
- ScalDeviceGroup resource type
 - with clsetup utility, 66-71
 - with Sun Cluster maintenance commands, 232
- ScalMountPoint resource type
 - with clsetup utility, 66-71
 - with Sun Cluster maintenance commands, 234
- SUNW.crs_framework resource type
 - with clsetup utility, 84-89
 - with Sun Cluster maintenance commands, 238
- SUNW.qfs resource type
 - with clsetup utility, 66-71
 - with Sun Cluster maintenance commands, 233
- SUNW.rac_cvm resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 229
- SUNW.rac_framework resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 227

registering (*Continued*)

- SUNW.rac_svm resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 228
- SUNW.rac_udlm resource type
 - with clsetup utility, 53-57
 - with Sun Cluster maintenance commands, 228
- SUNW.scalable_rac_listener resource type
 - with clsetup utility, 89-94
 - with Sun Cluster maintenance commands, 246
- SUNW.scalable_rac_server_proxy resource type
 - with clsetup utility, 84-89
 - with Sun Cluster maintenance commands, 239
- SUNW.scalable_rac_server resource type
 - with clsetup utility, 89-94
 - with Sun Cluster maintenance commands, 247
- SUNW.ScalDeviceGroup resource type
 - with clsetup utility, 66-71
 - with Sun Cluster maintenance commands, 232
- SUNW.ScalMountPoint resource type
 - with clsetup utility, 66-71
 - with Sun Cluster maintenance commands, 234
- relational database management system (RDBMS)
 - See also* database management system (DBMS)
 - file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
 - processor architecture requirements, 24
 - storage management schemes, 22
- removing
 - See also* uninstalling
 - Sun Cluster Support for Oracle RAC
 - from clusters, 167-173
 - from selected nodes, 173-184
 - SUNW.rac_hwraid resource, 153
- requirements
 - hardware, 21
 - multi-owner disk sets, 58
 - Oracle files, 21-24
 - processor architecture, 24
 - shared-disk groups, 64
 - software, 21
- reservation step timeout
 - description, 201

- reservation step timeout (*Continued*)
 - guidelines for setting, 111
- reservation_timeout extension property
 - description, 201
 - guidelines for setting, 111
 - preservation of nondefault values, 151-153
- Resource Group Manager (RGM), restrictions, 226
- resource groups
 - file-system mount point
 - affinities, 234
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance commands, 233-235
 - planning, 230
 - listener
 - creating with clsetup utility, 89-94
 - creating with Sun Cluster maintenance commands, 244-251
 - enabling, 249
 - planning, 242-243
 - Oracle 10g RAC server
 - affinities, 238
 - creating with clsetup utility, 84-89
 - creating with Sun Cluster maintenance commands, 238
 - enabling, 240
 - Oracle 9i RAC server
 - affinities, 244
 - creating with clsetup utility, 89-94
 - creating with Sun Cluster maintenance commands, 245
 - enabling, 249
 - planning, 242
 - RAC framework
 - creating with clsetup utility, 53-57
 - creating with Sun Cluster maintenance commands, 226-229
 - planning, 226
 - sample configurations, 185-188
 - scalable device group
 - affinities, 232
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - resource groups, scalable device group (*Continued*)
 - planning, 230
 - Sun StorEdge QFS metadata server
 - affinities, 233
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance commands, 232-233
 - planning, 230
 - resource types
 - crs_framework
 - dependencies, 88, 238
 - extension properties, 219
 - instantiating with clsetup utility, 84-89
 - instantiating with Sun Cluster maintenance commands, 238
 - names of instances created by clsetup, 106
 - registering with clsetup utility, 84-89
 - registering with Sun Cluster maintenance commands, 238
 - fault monitors, 113
 - LogicalHostname
 - dependencies, 94, 243
 - instantiating with clsetup utility, 89-94
 - instantiating with Sun Cluster maintenance commands, 245
 - names of instances created by clsetup, 107
 - planning, 243-244
 - names of instances created by clsetup, 106-107
 - oracle_rac_listener, 149
 - oracle_rac_server, 149
 - qfs
 - dependencies, 71, 233
 - instantiating with clsetup utility, 66-71
 - instantiating with Sun Cluster maintenance commands, 233
 - names of instances created by clsetup, 106
 - registering with clsetup utility, 66-71
 - registering with Sun Cluster maintenance commands, 233
 - rac_cvm
 - dependencies, 57, 229
 - extension properties, 207-210
 - instantiating with clsetup utility, 53-57

resource types, rac_cvm (*Continued*)

- instantiating with Sun Cluster maintenance commands, 229
- names of instances created by clsetup, 106
- purpose, 226
- registering with clsetup utility, 53-57
- registering with Sun Cluster maintenance commands, 229
- restrictions, 112
- upgrading, 149-153

rac_framework

- dependencies, 56
- extension properties, 201-202
- failure of instance to start, 144-146
- instantiating with clsetup utility, 53-57
- instantiating with Sun Cluster maintenance commands, 227
- monitoring instances of, 129
- names of instances created by clsetup, 106
- purpose, 226
- registering with clsetup utility, 53-57
- registering with Sun Cluster maintenance commands, 227
- START method timeout, 145
- upgrading, 149-153

rac_hwraid

- obsolescence, 149
- removing instance of, 153

rac_svm

- dependencies, 57, 228
- extension properties, 205-207
- instantiating with clsetup utility, 53-57
- instantiating with Sun Cluster maintenance commands, 228
- names of instances created by clsetup, 106
- purpose, 226
- registering with clsetup utility, 53-57
- registering with Sun Cluster maintenance commands, 228
- upgrading, 149-153

rac_udlm

- dependencies, 57, 228
- extension properties, 202-205
- instantiating with clsetup utility, 53-57

resource types, rac_udlm (*Continued*)

- instantiating with Sun Cluster maintenance commands, 228
- names of instances created by clsetup, 106
- purpose, 226
- registering with clsetup utility, 53-57
- registering with Sun Cluster maintenance commands, 228
- restrictions, 112
- upgrading, 149-153

restrictions, 226

sample configurations, 185-188

scalable_rac_listener

- dependencies, 94, 243
- extension properties, 214-216
- instantiating with clsetup utility, 89-94
- instantiating with Sun Cluster maintenance commands, 246
- names of instances created by clsetup, 106
- registering with clsetup utility, 89-94
- registering with Sun Cluster maintenance commands, 246

scalable_rac_server

- overview, 244
- dependencies, 94, 244
- instantiating with clsetup utility, 89-94
- instantiating with Sun Cluster maintenance commands, 247
- names of instances created by clsetup, 106
- registering with clsetup utility, 89-94
- registering with Sun Cluster maintenance commands, 247

scalable_rac_server_proxy

- dependencies, 88, 239
- extension properties, 216-219
- instantiating with clsetup utility, 84-89
- names of instances created by clsetup, 106
- registering with clsetup utility, 84-89
- registering with Sun Cluster maintenance commands, 239

ScalDeviceGroup

- dependencies, 71, 232
- extension properties, 219-221
- instantiating with clsetup utility, 66-71

- resource types, ScalDeviceGroup (*Continued*)
 - instantiating with Sun Cluster maintenance commands, 232
 - modifying instances of, 148
 - names of instances created by clsetup, 106
 - registering with clsetup utility, 66-71
 - registering with Sun Cluster maintenance commands, 232
- ScalMountPoint
 - dependencies, 71, 234
 - extension properties, 221-224
 - instantiating with clsetup utility, 66-71
 - instantiating with Sun Cluster maintenance commands, 234
 - names of instances created by clsetup, 106
 - registering with clsetup utility, 66-71
 - registering with Sun Cluster maintenance commands, 234
- SUNW.crs_framework
 - dependencies, 88, 238
 - extension properties, 219
 - instantiating with clsetup utility, 84-89
 - instantiating with Sun Cluster maintenance commands, 238
 - names of instances created by clsetup, 106
 - registering with clsetup utility, 84-89
 - registering with Sun Cluster maintenance commands, 238
- SUNW.LogicalHostname
 - dependencies, 94, 243
 - instantiating with clsetup utility, 89-94
 - instantiating with Sun Cluster maintenance commands, 245
 - names of instances created by clsetup, 107
 - planning, 243-244
- SUNW.oracle_rac_listener, 149
- SUNW.oracle_rac_server, 149
- SUNW.qfs
 - dependencies, 71, 233
 - instantiating with clsetup utility, 66-71
 - instantiating with Sun Cluster maintenance commands, 233
 - names of instances created by clsetup, 106
 - registering with clsetup utility, 66-71
- resource types, SUNW.qfs (*Continued*)
 - registering with Sun Cluster maintenance commands, 233
- SUNW.rac_cvm
 - dependencies, 57, 229
 - extension properties, 207-210
 - instantiating with clsetup utility, 53-57
 - instantiating with Sun Cluster maintenance commands, 229
 - names of instances created by clsetup, 106
 - purpose, 226
 - registering with clsetup utility, 53-57
 - registering with Sun Cluster maintenance commands, 229
 - restrictions, 112
 - upgrading, 149-153
- SUNW.rac_framework
 - dependencies, 56
 - extension properties, 201-202
 - failure of instance to start, 144-146
 - instantiating with clsetup utility, 53-57
 - instantiating with Sun Cluster maintenance commands, 227
 - monitoring instances of, 129
 - names of instances created by clsetup, 106
 - purpose, 226
 - registering with clsetup utility, 53-57
 - registering with Sun Cluster maintenance commands, 227
 - START method timeout, 145
 - upgrading, 149-153
- SUNW.rac_hwraid
 - obsolescence, 149
 - removing instance of, 153
- SUNW.rac_svm
 - dependencies, 57, 228
 - extension properties, 205-207
 - instantiating with clsetup utility, 53-57
 - instantiating with Sun Cluster maintenance commands, 228
 - names of instances created by clsetup, 106
 - purpose, 226
 - registering with clsetup utility, 53-57

- resource types, `SUNW.rac_svm` (*Continued*)
 - registering with Sun Cluster maintenance commands, 228
 - upgrading, 149-153
- `SUNW.rac_udlm`
 - dependencies, 57, 228
 - extension properties, 202-205
 - instantiating with `clsetup` utility, 53-57
 - instantiating with Sun Cluster maintenance commands, 228
 - names of instances created by `clsetup`, 106
 - purpose, 226
 - registering with `clsetup` utility, 53-57
 - registering with Sun Cluster maintenance commands, 228
 - restrictions, 112
 - upgrading, 149-153
- `SUNW.scalable_rac_listener`
 - dependencies, 94, 243
 - extension properties, 214-216
 - instantiating with `clsetup` utility, 89-94
 - instantiating with Sun Cluster maintenance commands, 246
 - names of instances created by `clsetup`, 106
 - registering with `clsetup` utility, 89-94
 - registering with Sun Cluster maintenance commands, 246
- `SUNW.scalable_rac_server`
 - overview, 244
 - dependencies, 94, 244
 - extension properties, 210-214
 - instantiating with `clsetup` utility, 89-94
 - instantiating with Sun Cluster maintenance commands, 247
 - names of instances created by `clsetup`, 106
 - registering with `clsetup` utility, 89-94
 - registering with Sun Cluster maintenance commands, 247
- `SUNW.scalable_rac_server_proxy`
 - dependencies, 88, 239
 - extension properties, 216-219
 - instantiating with `clsetup` utility, 84-89
 - instantiating with Sun Cluster maintenance commands, 239
- resource types, `SUNW.scalable_rac_server_proxy` (*Continued*)
 - names of instances created by `clsetup`, 106
 - registering with `clsetup` utility, 84-89
 - registering with Sun Cluster maintenance commands, 239
- `SUNW.ScalDeviceGroup`
 - dependencies, 71, 232
 - extension properties, 219-221
 - instantiating with `clsetup` utility, 66-71
 - instantiating with Sun Cluster maintenance commands, 232
 - modifying instances of, 148
 - names of instances created by `clsetup`, 106
 - registering with `clsetup` utility, 66-71
 - registering with Sun Cluster maintenance commands, 232
- `SUNW.ScalMountPoint`
 - dependencies, 71, 234
 - extension properties, 221-224
 - instantiating with `clsetup` utility, 66-71
 - instantiating with Sun Cluster maintenance commands, 234
 - names of instances created by `clsetup`, 106
 - registering with `clsetup` utility, 66-71
 - registering with Sun Cluster maintenance commands, 234
- resources
 - adding to RAC framework resource group, 165-167
 - CRS
 - creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 240-241
 - state changes, 107-109
 - database
 - creating with `clsetup` utility, 82-94
 - creating with Sun Cluster maintenance commands, 235-241
 - file-system mount point
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 233-235
 - planning, 230

resources (*Continued*)

listener

- creating with `clsetup` utility, 89-94
- creating with Sun Cluster maintenance commands, 244-251
- planning, 242-243

Oracle 10g RAC server

- creating with `clsetup` utility, 84-89
- creating with Sun Cluster maintenance commands, 239
- disabling, 107-109
- enabling, 240

Oracle 9i RAC server

- creating with `clsetup` utility, 89-94
- creating with Sun Cluster maintenance commands, 247
- disabling, 109
- enabling, 249
- planning, 242

proxy

- creating with `clsetup` utility, 84-89
- creating with Sun Cluster maintenance commands, 240-241
- examples, 236

RAC framework resource group

- creating with `clsetup` utility, 53-57
- creating with Sun Cluster maintenance commands, 226-229
- planning, 226

sample configurations, 185-188

scalable device group

- creating with `clsetup` utility, 66-71
- creating with Sun Cluster maintenance commands, 231-232
- planning, 230

storage

- creating with `clsetup` utility, 66-71
- creating with Sun Cluster maintenance commands, 229-235
- planning, 230

Sun Cluster, 107-109

Sun StorEdge QFS metadata server

- creating with `clsetup` utility, 66-71

resources, Sun StorEdge QFS metadata server (*Continued*)

- creating with Sun Cluster maintenance commands, 232-233
- planning, 230

responding, to major errors, 121-122

restarts

- preventing
 - for DBMS errors, 122-123
 - for timeouts, 124-126

restrictions

- cylinders, 44
- device groups, 64
- file systems, 58, 64
- Guard option, 25
- hostnames, 25
- multi-owner disk sets, 58
- partitions, 44
- rebooting nodes, 31
- resource types, 226
- RGM, 226
- shared-disk groups, 64
- Solaris Volume Manager for Sun Cluster, 23
- Sun Cluster command usage, 25
- tuning extension properties, 112
- zones, 20

return step timeout

- Solaris Volume Manager for Sun Cluster, 205
- VxVM, 207

RGM (Resource Group Manager), restrictions, 226

running, *See* starting**S**

samfs.cmd file, 47

sample configurations, 185-188

scalable device-group fault monitor, 114

scalable device-group resources

- modifying, 148
- state information, 139
- `syslog()` function, 139

scalable device groups

- resource group affinities, 232

- scalable device groups (*Continued*)
 - resource groups for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
 - resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
- scalable_rac_listener resource type
 - dependencies, 94, 243
 - extension properties, 214-216
 - instantiating
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 246
 - names of instances created by `clsetup`, 106
 - registering
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 246
- scalable_rac_server_proxy resource type
 - dependencies, 88, 239
 - extension properties, 216-219
 - instantiating
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 239
 - names of instances created by `clsetup`, 106
 - registering
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 239
- scalable_rac_server resource type
 - overview, 244
 - dependencies, 94, 244
 - extension properties, 210-214
 - instantiating
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 247
 - names of instances created by `clsetup`, 106
 - registering
 - with `clsetup` utility, 89-94
 - with Sun Cluster maintenance commands, 247
- ScalDeviceGroup resource type
 - dependencies, 71, 232
- ScalDeviceGroup resource type (*Continued*)
 - extension properties, 219-221
 - instantiating
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 232
 - modifying instances of, 148
 - names of instances created by `clsetup`, 106
 - registering
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 232
- ScalMountPoint resource type
 - dependencies, 71, 234
 - extension properties, 221-224
 - instantiating
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 234
 - names of instances created by `clsetup`, 106
 - registering
 - with `clsetup` utility, 66-71
 - with Sun Cluster maintenance commands, 234
- schedclass extension property, 203
- schedpriority extension property, 203
- secondary groups, 27
- server
 - extension properties
 - Oracle 10g, 216-219
 - Oracle 9i, 210-214
 - resource groups for Oracle 10g
 - creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 238
 - enabling, 240
 - resource groups for Oracle 9i
 - creating with `clsetup` utility, 89-94
 - creating with Sun Cluster maintenance commands, 245
 - enabling, 249
 - planning, 242
 - resources for Oracle 10g
 - creating with `clsetup` utility, 84-89
 - creating with Sun Cluster maintenance commands, 239
 - disabling, 107-109
 - enabling, 240

- server (*Continued*)
 - resources for Oracle 9i
 - creating with `clsetup` utility, 89-94
 - creating with Sun Cluster maintenance commands, 247
 - disabling, 109
 - enabling, 249
- server fault monitor
 - actions
 - definition, 116
 - modifying, 120
 - alert logs, 117
 - caution notice, 118
 - customizing, 118-127
 - error types detected by, 118
 - overview, 115-117
 - preset actions, 193-200
- sessions
 - effect of errors on, 121, 122
- SGA (shared global area), errors, 121
- shadow file, 28
- shared-disk groups
 - creating, 64-65
 - storage resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
- shared global area (SGA), errors, 121
- shared memory, 30-31
- show-rev subcommand, 18
- showrev `-p` command, 18
- shutdown, verifying, 102-103
- shutting down
 - databases
 - Oracle 10g, 107-109
 - Oracle 9i, 109
- slices
 - concatenating, 59
 - disk, 43
- snmp_ro.ora file, 79
- snmp_rw.ora file, 79
- soft partitions, 59
- software packages, 31-33
- software requirements, 21
- Solaris Volume Manager, 24
- Solaris Volume Manager for Sun Cluster
 - configuring, 41-42
 - extension properties, 205-207
 - installing, 41-42
 - multi-owner disk sets, 58-64
 - Oracle file types supported, 22
 - Oracle Parallel Server, 23
 - resource type for, 226
 - restrictions, 23
 - storage resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance commands, 231-232
 - planning, 230
 - task summary, 37-38
- SPFILE file
 - in cluster file system, 51
 - file-system options, 52
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 45-46
- sqlplus command, 81
- srvm/log directory, 76
- Start failed state, 144
- START method, 145
- start step timeout
 - Oracle DLM, 203
 - Solaris Volume Manager for Sun Cluster, 205
 - VxVM, 207
- starting
 - databases
 - Oracle 10g, 107-109
 - Oracle 9i, 109
 - failure by resources, 144-146
- startup_wait_count extension property, 218
- state information
 - database resources, 107-109
 - file-system mount-point resources, 139
 - scalable device-group resources, 139
- status information, Sun Cluster Support for Oracle RAC, 129
- stopping, failure by resources, 146

- storage management schemes
 - choosing, 21-24
 - installing software for, 41-53
 - resource types for, 226
- storage resources
 - configuring after Sun Cluster upgrade, 153-154
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 229-235
 - planning, 230
 - verifying, 96-98
- stripes, 48
 - adding to volume, 59
- Sun Cluster
 - command usage restrictions, 25
 - framework, 21
- Sun Cluster maintenance commands
 - comparison with `clsetup` utility, 53, 66, 83
 - CRS resources, 240-241
 - proxy resources, 240-241
 - RAC framework resource group
 - creating, 226-229
 - storage resources, 230, 231-232
 - tuning extension properties with, 110-113
- Sun Cluster Support for Oracle RAC
 - overview, 20
 - administering, 105-127
 - configuring
 - for selected nodes, 158-164
 - examples, 185-188
 - fault monitors, 113-118
 - installing, 31-33
 - verifying installation, 95-103
 - log files
 - additional messages in, 121
 - modifying
 - by removing nodes from, 173-184
 - monitoring, 129
 - removing
 - from clusters, 167-173
 - from selected nodes, 173-184
 - sample configurations, 185-188
 - software packages
 - installing, 31-33
 - Sun Cluster Support for Oracle RAC, software packages
 - (*Continued*)
 - uninstalling from clusters, 171
 - uninstalling from selected nodes, 180
 - status information, 129
 - troubleshooting, 129-146
 - tuning, 110-113
 - upgrading, 148-157
 - Sun StorEdge disk arrays, 43-45
 - Sun StorEdge QFS metadata server
 - resource groups for
 - affinities, 233
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 232-233
 - planning, 230
 - resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 232-233
 - planning, 230
 - Sun StorEdge QFS shared file system
 - creating, 47-48
 - extension properties, 201-202
 - installation of Oracle files, 24
 - mounting, 47-48
 - node-specific directories, 76-78
 - node-specific files, 79-81
 - Oracle file types supported, 22
 - requirements, 45-46
 - storage resources for
 - creating with `clsetup` utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 231-232
 - planning, 230
 - task summary, 36-37
 - SUNW.crs_framework resource type
 - dependencies, 88, 238
 - extension properties, 219
 - instantiating
 - with `clsetup` utility, 84-89
 - with Sun Cluster maintenance commands, 238
 - names of instances created by `clsetup`, 106

`SUNW.crs_framework` resource type (*Continued*)
 registering
 with `clsetup` utility, 84-89
 with Sun Cluster maintenance commands, 238

`SUNW.LogicalHostname` resource type
 dependencies, 94, 243
 instantiating
 with `clsetup` utility, 89-94
 with Sun Cluster maintenance commands, 245
 names of instances created by `clsetup`, 107
 planning, 243-244

`SUNW.oracle_rac_listener` resource type, 149

`SUNW.oracle_rac_server` resource type, 149

`SUNW.qfs` resource type
 dependencies, 71, 233
 instantiating
 with `clsetup` utility, 66-71
 with Sun Cluster maintenance commands, 233
 names of instances created by `clsetup`, 106
 registering
 with `clsetup` utility, 66-71
 with Sun Cluster maintenance commands, 233

`SUNW.rac_cvm` resource type
 dependencies, 57, 229
 extension properties, 207-210
 instantiating
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 229
 names of instances created by `clsetup`, 106
 purpose, 226
 registering
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 229
 restrictions, 112
 upgrading, 149-153

`SUNW.rac_framework` resource type
 dependencies, 56
 extension properties, 201-202
 failure of instance to start, 144-146
 instantiating
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 227
 monitoring instances of, 129
 names of instances created by `clsetup`, 106

`SUNW.rac_framework` resource type (*Continued*)
 purpose, 226
 registering
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 227
 START method timeout, 145
 upgrading, 149-153

`SUNW.rac_hwraid` resource type
 obsolescence, 149
 removing instance of, 153

`SUNW.rac_svm` resource type
 dependencies, 57, 228
 extension properties, 205-207
 instantiating
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 228
 names of instances created by `clsetup`, 106
 purpose, 226
 registering
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 228
 upgrading, 149-153

`SUNW.rac_udlm` resource type
 dependencies, 57, 228
 extension properties, 202-205
 instantiating
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 228
 names of instances created by `clsetup`, 106
 purpose, 226
 registering
 with `clsetup` utility, 53-57
 with Sun Cluster maintenance commands, 228
 restrictions, 112
 upgrading, 149-153

`SUNW.scalable_rac_listener` resource type
 dependencies, 94, 243
 extension properties, 214-216
 instantiating
 with `clsetup` utility, 89-94
 with Sun Cluster maintenance commands, 246
 names of instances created by `clsetup`, 106
 registering
 with `clsetup` utility, 89-94

- SUNW.scalable_rac_listener resource type, registering (*Continued*)
 with Sun Cluster maintenance commands, 246
- SUNW.scalable_rac_server_proxy resource type
 dependencies, 88, 239
 extension properties, 216-219
 instantiating
 with clsetup utility, 84-89
 with Sun Cluster maintenance commands, 239
 names of instances created by clsetup, 106
 registering
 with clsetup utility, 84-89
 with Sun Cluster maintenance commands, 239
- SUNW.scalable_rac_server resource type
 overview, 244
 dependencies, 94, 244
 extension properties, 210-214
 instantiating
 with clsetup utility, 89-94
 with Sun Cluster maintenance commands, 247
 names of instances created by clsetup, 106
 registering
 with clsetup utility, 89-94
 with Sun Cluster maintenance commands, 247
- SUNW.ScalDeviceGroup resource type
 dependencies, 71, 232
 extension properties, 219-221
 instantiating
 with clsetup utility, 66-71
 with Sun Cluster maintenance commands, 232
 names of instances created by clsetup, 106
 registering
 modifying instances of, 148
 with clsetup utility, 66-71
 with Sun Cluster maintenance commands, 232
- SUNW.ScalMountPoint resource type
 dependencies, 71, 234
 extension properties, 221-224
 instantiating
 with clsetup utility, 66-71
 with Sun Cluster maintenance commands, 234
 names of instances created by clsetup, 106
 registering
 with clsetup utility, 66-71
- SUNW.ScalMountPoint resource type, registering (*Continued*)
 with Sun Cluster maintenance commands, 234
- svm_abort_step_timeout extension property, 205
- svm_return_step_timeout extension property, 205
- svm_start_step_timeout extension property, 205
- svm_step1_timeout extension property, 206
- svm_step2_timeout extension property, 206
- svm_step3_timeout extension property, 206
- svm_step4_timeout extension property,
 definition, 206
- Svm_step4_timeout extension property, guidelines for setting, 110
- syntax errors, custom action file, 127
- syslog() function, 139
- syslog messages, 211, 214, 217
- system configuration files, *See* Oracle configuration files
- system file, 31
- system identifiers
 Oracle, 212, 217
- system messages file, 139
- system panics, *See* panics
- system parameter file
 in cluster file system, 51
 file-system options, 52
 storage management schemes, 22
 in Sun StorEdge QFS shared file system, 45-46
- system properties, effect on fault monitors, 113
- ## T
- target filesystem extension property, 223
- TCP/IP (Transmission Control Protocol/Internet Protocol), timeout, 243
- technical support, 18
- timeouts
 ASM, 201
 core-file creation, 117-118
 log files for, 139
 modifying maximum allowed, 124-126
 Oracle UDLM, 203
 panics, and, 143-144
 reservation step, 111, 201

timeouts (*Continued*)

- Solaris Volume Manager for Sun Cluster
 - definition, 205
- TCP/IP, 243
- VxVM
 - definition, 207
 - reconfiguration step 4, 110
- topology requirements, 21
- trace files
 - in cluster file system, 51
 - file-system options
 - Sun StorEdge QFS shared file system, 47
 - UNIX file system, 52
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 45-46
- Transmission Control Protocol/Internet Protocol (TCP/IP), timeout, 243
- troubleshooting, Sun Cluster Support for Oracle RAC, 129-146
- tuning
 - fault monitors, 113-118
 - Sun Cluster Support for Oracle RAC, 110-113
- turning off, *See* disabling
- turning on, *See* starting

U

- UCMM (userland cluster membership monitor)
 - configuration information, 139
 - failure to start, 141
 - unexpected termination, 141
- ucmm_reconf.log file, 139
- ucmmd daemon
 - failure to start, 141
 - unexpected termination, 141
- udlm_abort_step_timeout extension property, 203
- udlm.conf configuration file, 202
- udlm_start_step_timeout extension property, 203
- udlm_step1_timeout extension property, 204
- udlm_step2_timeout extension property, 204
- udlm_step3_timeout extension property, 204
- udlm_step4_timeout extension property, 204
- udlm_step5_timeout extension property, 204
- UFS (UNIX file system), configuring, 52

uninstalling

- Sun Cluster Support for Oracle RAC software
 - packages
 - from clusters, 171
 - from selected nodes, 180
- UNIX Distributed Lock Manager (Oracle UDLM)
 - caution notice, 33
 - communications ports
 - extension properties, 202
 - guidelines for setting, 111-112
 - core files, 139
 - effect on tunable when disabled extension properties, 112
 - event logs, 139
 - extension properties, 202-205
 - installing, 33-34
 - log files, 140
 - processor architecture requirements, 24
 - resource type for, 226
 - restrictions, 112
- UNIX file system (UFS), configuring, 52
- upgrading, Sun Cluster Support for Oracle RAC, 148-157
- user_env extension property
 - scalable_rac_listener resource type, 215
 - scalable_rac_server_proxy resource type, 218
 - scalable_rac_server resource type, 213
 - SUNW.scalable_rac_listener resource type, 215
 - SUNW.scalable_rac_server_proxy resource type, 218
 - SUNW.scalable_rac_server resource type, 213
- user identifiers, 26
- userland cluster membership monitor (UCMM)
 - configuration information, 139
 - failure to start, 141
 - unexpected termination, 141
- users
 - creating, 27-30
 - granting access to file systems, 48
 - granting access to volumes, 60

V

- validating, custom action file, 127

- validation failures, components, 141
- /var/adm/messages file, 18
- /var/cluster/ucmm directory, 139
- /var/opt/SUNWscor/oracle_server directory, 140
- /var/opt directory, 29
- verifying
 - cluster boot, 102-103
 - cluster shutdown, 102-103
 - database resources
 - Oracle 10g, 98-101
 - Oracle 9i, 101-102
 - Oracle RAC, 81
 - RAC framework resource group, 95-96
 - storage resources, 96-98
- VERITAS Volume Manager (VxVM)
 - caution notice, 42
 - configuring, 42
 - effect on tunable when disabled extension
 - properties, 112
 - extension properties, 207-210
 - installing, 42
 - license requirements, 21
 - Oracle file types supported, 22
 - reconfiguration step 4 timeout, 110
 - reservation step timeout, 111
 - resource type for, 226
 - restrictions, 112
 - shared-disk groups, 64-65
 - storage resources for
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 231-232
 - planning, 230
 - task summary, 38-39
 - vxlicrep command, 21
- vfstab file
 - Sun StorEdge QFS, 47
 - UNIX file system, 52
- volume managers, 24
 - See also* Solaris Volume Manager for Sun Cluster
 - adding to existing configuration, 165-167
 - storage resources for
 - creating with clsetup utility, 66-71
- volume managers, storage resources for (*Continued*)
 - creating with Sun Cluster maintenance
 - commands, 231-232
 - planning, 230
- volumes, 220
 - adding to multi-owner disk sets, 59
 - gen use type, 65
 - monitoring, 148
 - removing from multi-owner disk sets, 170
- voting disk
 - in cluster file system, 51
 - file-system options, 52
 - storage management schemes, 22
 - in Sun StorEdge QFS shared file system, 46
- vxclust_num_ports extension property, 209
- vxclust_port extension property, 209
- vxclust program, 209
- vxconfigd daemon, 209
- vxconfigd_port extension property, 209
- vxkmsgd daemon, 209
- vxkmsgd_port extension property, 209
- vxlicrep command, 21
- VxVM (VERITAS Volume Manager)
 - caution notice, 42
 - configuring, 42
 - effect on tunable when disabled extension
 - properties, 112
 - extension properties, 207-210
 - installing, 42
 - license requirements, 21
 - Oracle file types supported, 22
 - reconfiguration step 4 timeout, 110
 - reservation step timeout, 111
 - resource type for, 226
 - restrictions, 112
 - shared-disk groups, 64-65
 - storage resources for
 - creating with clsetup utility, 66-71
 - creating with Sun Cluster maintenance
 - commands, 231-232
 - planning, 230
 - task summary, 38-39
 - vxlicrep command, 21

W

- wait_for_online extension property
 - scalable_rac_server resource type, 214
 - SUNW.scalable_rac_server resource type, 214
- width, disk stripes, 48
- wizards
 - Oracle RAC database instances, 83
 - RAC framework resource group, 53
 - storage resources, 66

Z

- zones, 20