Oracle® Database

Oracle Clusterware and Oracle Real Application Clusters Installation Guide

10*g* Release 2 (10.2) for Linux

B14203-01

June 2005

ORACLE®

Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide, 10*g* Release 2 (10.2) for Linux

B14203-01

Copyright © 2004, 2005, Oracle. All rights reserved.

Primary Authors: David Austin, Mark Bauer, Kevin Flood, Emily Murphy, Lyju Vadassery, Douglas Williams

Contributing Authors: Jonathan Creighton, Pat Huey, Raj Kumar

Contributors: Chris Allison, Karin Brandauer, Sudip Datta, Rajiv Jayaraman, Roland Knapp, Diana Lorentz, Barb Lundhild, Vijay Lunawat, John Patrick McHugh, Randy Neville, Philip Newlan, Michael Polaski, Dipak Saggi, Sudheendra Sampath, Janelle Simmons, Clive Simpkins, Khethavath P. Singh, Nitin Vengurlekar, Gary Young

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software—Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Send Us Your Comments ix

Preface xi

Intended Audience xi
Documentation Accessibility xi
Structure xii
Related Documents xiii
Conventions xv

What's New in Oracle Database 10*g* Oracle Real Application Clusters Installation and Configuration? xix

Oracle Database 10*g* Release 2 (10.2) New Features for RAC Installation and Configuration xix

Part I Oracle Clusterware and Oracle Real Application Clusters Installation Planning and Requirements

1 Introduction to Installing and Configuring Oracle Clusterware and Oracle Real Application Clusters

Oracle Clusterware and Oracle Real Application Clusters Documentation Overview 1-1
Oracle Real Application Clusters Administration and Deployment Guide 1-2
General System Installation Requirements 1-2
Overview of Cluster Verification Utility 1-2
Overview of Oracle Universal Installer 1-3
Overview of Upgrading and Multiple-release Systems 1-3
Hardware Requirements for Oracle Clusterware and Oracle Real Application Clusters 1-3
Software Requirements for Oracle Clusterware and Oracle Real Application Clusters 1-4
Configuration Tasks for Oracle Clusterware and Oracle Real Application Clusters 1-5
Pre-Installation, Installation, and Post-Installation Overview 1-6
Pre-Installation Overview for Oracle Database 10*g* Real Application Clusters 1-6
Installation Overview for Oracle Database 10*g* Real Application Clusters 1-6
Post-Installation Overview for Oracle Database 10*g* Real Application Clusters 1-7
Oracle Universal Installer and Real Application Clusters 1-7
Storage Considerations for Installing Oracle Database 10*g* Real Application Clusters 1-8
Overview of Automatic Storage Management 1-8
Shared Storage for Database Recovery Area 1-10

Additional Considerations for Using Oracle Database 10*g* Features in RAC 1-11
Oracle Database 10*g* and Real Application Clusters Components 1-11
Oracle Clusterware 1-11
The Installed Real Application Clusters Components 1-12
Oracle Database 10*g* Real Application Clusters Release Compatibility 1-12
Required Operating System Groups 1-13
Cloning Oracle Clusterware and RAC in Grid Environments 1-14
Cloning Oracle Clusterware Homes 1-14
Cloning RAC Homes 1-15

Part II Oracle Clusterware and Oracle Real Application Clusters Pre-Installation Procedures

2 Pre-Installation Tasks

Understanding and Using Cluster Verification Utility 2-1
Prerequisites for Using Cluster Verification Utility 2-2
Entering Cluster Verification Utility Commands 2-2
Using CVU to Determine if Installation Prerequisites are Complete 2-2
Using the Cluster Verification Utility Help 2-3
Using Cluster Verification Utility with Oracle 10g Release 1 2-3
Verbose Mode and "Unknown" Output 2-4
Logging In to the System as root 2-4
Creating Required Operating System Groups and User 2-5
Creating the Oracle Inventory Group 2-7
Creating the OSDBA Group 2-7
Creating an OSOPER Group (Optional) 2-8
Creating the Oracle Software Owner User 2-8
Verifying That the User nobody Exists 2-9
Creating Identical Users and Groups on Other Cluster Nodes 2-10
Configuring SSH on All Cluster Nodes 2-11
Configuring SSH on Cluster Member Nodes 2-11
Enabling SSH User Equivalency on Cluster Member Nodes 2-13
Preventing Oracle Clusterware Installation Errors Caused by stty Commands 2-14
Configuring the oracle User's Environment 2-15
Installing the cvuqdisk Package for Red Hat Linux 2-17
Checking the Hardware Requirements 2-18
Checking the Network Requirements 2-19
Checking the Network Setup with CVU 2-21
Checking the Software Requirements 2-22
Configuring Kernel Parameters 2-25
Checking the Hardware and Operating System Setup with CVU 2-27
Checking the Operating System Requirements Setup with CVU 2-28
Identifying Required Software Directories 2-28
Identifying or Creating an Oracle Base Directory 2-30
Creating the Oracle Clusterware Home Directory 2-32
Stopping Existing Oracle Processes for Upgrades or Co-existing Databases 2-34

3 Configuring Oracle Clusterware and Oracle Database Storage

Reviewing Storage Options for Oracle Clusterware, Database, and Recovery Files 3-1
Overview of Storage Options 3-1
Overview of Oracle Clusterware Storage Options 3-1
Overview of Oracle Database and Recovery File Options 3-2
General Storage Considerations 3-2
After You Have Selected Disk Storage Options 3-4
Checking for Available Shared Storage with CVU 3-4
Configuring Storage for Oracle Clusterware Files on a Supported Shared File System 3-5
Requirements for Using a file system for Oracle Clusterware Files 3-5
Checking NFS Buffer Size Parameters 3-6
Creating Required Directories for Oracle Clusterware Files on Shared File Systems 3-6
Configuring Storage for Oracle Clusterware Files on Raw Devices 3-8
Clusterware File Restrictions for Logical Volume Manager on Linux 3-8
Identifying Required Raw Partitions for Clusterware Files 3-8
Creating the Required Raw Partitions on IDE, SCSI, or RAID Devices 3-9
Binding the Partitions to Raw Devices 3-10
Choosing a Storage Option for Oracle Database Files 3-11
Configuring Disks for Automatic Storage Management 3-12
Identifying Storage Requirements for Automatic Storage Management 3-12
Using an Existing Automatic Storage Management Disk Group 3-15
Configuring Disks for Automatic Storage Management with ASMLIB 3-16
Configuring Database File Storage on ASM and Raw Devices 3-20
Configuring Database File Storage on Raw Devices 3-21
Database File Restrictions for Logical Volume Manager on Linux 3-21
Identifying Required Raw Partitions for Database Files 3-21
Creating Required Raw Partitions for Database Files on IDE, SCSI, or RAID Devices 3-22
Binding Partitions to Raw Devices for Database Files 3-23
Creating the Database Configuration Assistant Raw Device Mapping File 3-24

Part III Installing Oracle Clusterware and Oracle Database 10*g* with Oracle Real Application Clusters, Creating Oracle Real Application Clusters Databases, and Performing Post-Installation Tasks

4 Installing Oracle Clusterware

Verifying Oracle Clusterware Requirements with CVU 4-1
Troubleshooting Oracle Clusterware Setup 4-2
Preparing to Install Oracle Clusterware with OUI 4-3
Installing Oracle Clusterware with OUI 4-7
Running OUI to Install Oracle Clusterware 4-7
Troubleshooting Oracle Clusterware Installation Verification 4-7
Oracle Clusterware Background Processes 4-9

5 Installing Oracle Database 10*g* with Real Application Clusters

Verifying System Readiness for installing Oracle Database with CVU 5-1
Troubleshooting Installation Setup for Linux 5-2

Selecting a Configuration Option 5-3
Automatic Storage Management Configuration 5-4
Database Configuration Type Descriptions 5-4
General Purpose, Transaction Processing, and Data Warehouse Configuration Types 5-4
Using the Advanced Configuration Type 5-5
Behavior of OUI, DBCA, and Other Assistants During Installation 5-5
Installation of Oracle Database 10*g* with RAC Using the Oracle Universal Installer 5-5
De-Installing Real Application Clusters Software 5-7
De-Installing Oracle Database 10*g* RAC Software and ASM 5-7
De-Installing Oracle Clusterware 5-9

6 Creating Real Application Clusters Databases with Database Configuration Assistant

Using Database Configuration Assistant in Real Application Clusters 6-1
Benefits of Using Database Configuration Assistant 6-2
Oracle Real Application Clusters High Availability Services 6-2
Service Configuration and Instance Preferences 6-2
Transparent Application Failover Policies 6-2
Automatic Listener Migration from Release 10.1 to 10.2 6-3
Verifying Requirements for DBCA 6-3
Creating an Oracle Real Application Clusters Database with DBCA 6-4
Deleting a Real Application Clusters Database with DBCA 6-6

7 Configuring the Server Parameter File in Oracle Real Application Clusters Environments

Parameter Files and Real Application Clusters 7-1
Using Server Parameter Files in Real Application Clusters 7-1
Location of The Server Parameter File 7-2
Parameter File Search Order in Real Application Clusters 7-2
Migrating to the Server Parameter File in Real Application Clusters Environments 7-3
Server Parameter File Placement in Real Application Clusters 7-3
Procedures for Migrating to the Server Parameter File 7-3
Server Parameter File Errors in Real Application Clusters 7-4

8 Oracle Real Application Clusters Post-Installation Procedures

Required Post-Installation Tasks 8-1
Back Up the Voting Disk After Installation 8-1
Download and Install Patches 8-1
Configure Oracle Products 8-2
Oracle Real Application Clusters 10*g* Installed on an OCFS 8-2
Verify Required Post-Installation is Completed 8-3
Recommended Post-Installation Tasks 8-3
Verifying Enterprise Manager Operations 8-3
Recommended Post-Installation Tasks 8-3
Back Up the root.sh Script 8-3
Set Up Users Accounts 8-3

Set the ORACLE_HOME and ORACLE_SID Environments 8-4

Part IV Oracle Real Application Clusters Environment Configuration

9 Understanding the Oracle Real Application Clusters Installed Configuration

Understanding the Configured Environment in Real Application Clusters 9-1
Oracle Cluster Registry in Oracle Real Application Clusters 9-1
Oratab Configurations for Oracle Real Application Clusters 9-2
Database Components Created Using Database Configuration Assistant 9-2
Tablespaces and Data files 9-2
Control Files 9-3
Redo Log Files 9-3
Managing Undo Tablespaces in Real Application Clusters 9-3
Initialization Parameter Files 9-4
Configuring Service Registration-Related Parameters in Real Application Clusters 9-4
Configuring the Listener File (listener.ora) 9-5
Local Listeners 9-5
Multiple Listeners 9-5
How Oracle Database Uses the Listener (listener.ora File) 9-5
Listener Registration and PMON Discovery 9-6
Directory Server Access (ldap.ora File) 9-6
Net Service Names (tnsnames.ora File) 9-7
Net Services Profile (sqlnet.ora File) 9-11

Part V Oracle Clusterware and Oracle Real Application Clusters Installation and Configuration Reference Information

A Troubleshooting the Oracle Clusterware and Oracle Real Application Clusters Installation Process

Troubleshooting the Oracle Real Application Clusters Installation A-1
General Installation Issues A-1
Real Application Clusters Installation Error Messages A-2
Performing Cluster Diagnostics During Real Application Clusters Installations A-2

B Using Scripts to Create Oracle Real Application Clusters Databases

Creating a Database Using Scripts B-1

C Configuring Raw Devices for Oracle Real Application Clusters

Raw Devices Required by DBCA for Non-OCFS Environments C-1
Planning Your Raw Device Creation Strategy C-1

D Converting to Oracle Real Application Clusters from Single-Instance Oracle Databases

Single-Instance to Cluster-Enabled Conversion Administrative Issues D-1
Converting with DBCA D-2

Single Instance on a Non-Cluster Computer to Oracle Database 10*g* with RAC D-2
Back Up the Original Single-Instance Database D-2
Perform the Pre-Installation Steps D-3
Set Up the Cluster D-3
Validate the Cluster D-3
Copy the Preconfigured Database Image D-3
Install Oracle Database 10*g* Software with Real Application Clusters D-3
Single Instance on a Cluster to Oracle Database 10*g* RAC D-4
Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home D-4
Single Instance on a Cluster Running from a RAC-Disabled Oracle Home D-7
Single Instance on a Cluster Running from a Single-Instance Installed Oracle Home D-7
Converting with rconfig and Oracle Enterprise Manager D-7
Prerequisites for Converting to RAC Databases D-8
Single-Instance to RAC Conversion Scenarios D-8
Converting Single-Instance Databases to RAC Using rconfig D-9
Example of an rconfig XML Input File D-10
Converting Single-Instance Databases to RAC Using Oracle Enterprise Manager D-11
Converting Single-Instance ASM to Cluster ASM D-12
Post-Conversion Steps D-13

E Directory Structure for Oracle Database 10*g* Oracle Real Application Clusters Environments

Understanding the Real Application Clusters Directory Structure E-1
Directory Structures for Real Application Clusters E-1

F How to Stop Processes in an Existing Oracle Real Application Clusters Database

Back Up the Oracle Real Application Clusters Database F-1
Shut Down Oracle Real Application Clusters Databases F-1
Stop All Oracle Processes F-1
Stopping Oracle Clusterware or Cluster Ready Services Processes F-2
Stopping Oracle Database 10g Processes Before Adding Products or Upgrading F-2

Index

Send Us Your Comments

Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide 10*g* Release 2 (10.2) for Linux

B14203-01

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

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

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

- Electronic mail: infodev_us@oracle.com
- FAX: (650) 506-7227. Attn: Server Technologies Documentation Manager
- Postal service:

  Oracle Corporation
  Server Technologies Documentation Manager
  500 Oracle Parkway, Mailstop 4op11
  Redwood Shores, CA 94065
  USA

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

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

Preface

Oracle Real Application Clusters Installation and Configuration Guide explains how to install and configure Oracle Clusterware and Oracle Real Application Clusters (RAC). This preface contains the following topics:

- Intended Audience
- Documentation Accessibility
- Structure
- Related Documents
- Conventions

Intended Audience

Oracle Real Application Clusters Installation and Configuration Guide provides configuration information for network and system administrators, and database installation information for database administrators (DBAs) who install and configure RAC.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

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

Structure

This document is divided into five parts. It contains the following chapters:

Part I: "Oracle Clusterware and Oracle Real Application Clusters Installation Planning and Requirements"

Part I introduces the Oracle Clusterware and RAC installation process.

Chapter 1, "Introduction to Installing and Configuring Oracle Clusterware and Oracle Real Application Clusters"

This chapter describes the RAC installation process and provides Oracle Clusterware and RAC installation planning information.

Part II: Oracle Clusterware and Oracle Real Application Clusters Pre-Installation Procedures

Part II describes the platform-specific pre-installation procedures for installing Oracle Clusterware and RAC.

Chapter 2, "Pre-Installation Tasks"

This chapter describes the server pre-installation procedures for installing Oracle Clusterware and RAC.

Chapter 3, "Configuring Oracle Clusterware and Oracle Database Storage"

This chapter describes pre-installation procedures for configuring storage for Oracle Clusterware and Oracle Real Application Clusters.

Part III: "Installing Oracle Clusterware and Oracle Database 10g with Oracle Real Application Clusters, Creating Oracle Real Application Clusters Databases, and Performing Post-Installation Tasks"

Part III describes how to install Oracle Clusterware and Oracle Database 10*g* with Real Application Clusters for Linux.

Chapter 4, "Installing Oracle Clusterware"

This chapter describes how to install Oracle Clusterware for Linux.

Chapter 5, "Installing Oracle Database 10g with Real Application Clusters"

This chapter describes how to install Oracle Database 10*g* with Oracle Real Application Clusters on all operating systems.

Chapter 6, "Creating Real Application Clusters Databases with Database Configuration Assistant"

This chapter explains how to use Database Configuration Assistant to create RAC databases.

Chapter 8, "Oracle Real Application Clusters Post-Installation Procedures"

This chapter describes the post-installation tasks for RAC.

Part IV: Oracle Real Application Clusters Environment Configuration

Part IV provides Oracle Database 10*g* Real Application Clusters environment configuration information.

Chapter 7, "Configuring the Server Parameter File in Oracle Real Application Clusters Environments"

This chapter describes the use of the server parameter file (SPFILE) in Oracle Real Application Clusters.

Chapter 9, "Understanding the Oracle Real Application Clusters Installed Configuration"

This chapter describes the Oracle Database 10*g* Real Application Clusters installed configuration.

Part V: Oracle Clusterware and Oracle Real Application Clusters Installation and Configuration Reference Information

Part V provides reference information for the installation and configuration of RAC.

Appendix A, "Troubleshooting the Oracle Clusterware and Oracle Real Application Clusters Installation Process"

This appendix provides RAC installation and configuration troubleshooting information.

Appendix B, "Using Scripts to Create Oracle Real Application Clusters Databases"

This appendix explains how to use scripts in RAC.

Appendix C, "Configuring Raw Devices for Oracle Real Application Clusters"

This appendix explains how to configure shared disk subsystems using raw devices in RAC environments.

Appendix D, "Converting to Oracle Real Application Clusters from Single-Instance Oracle Databases"

This appendix describes how to convert to Oracle Database 10*g* RAC from single-instance Oracle databases.

Appendix E, "Directory Structure for Oracle Database 10g Oracle Real Application Clusters Environments"

This appendix describes the directory structure for the installed RAC software.

Related Documents

For more information, refer to the following Oracle resources:

Oracle Real Application Clusters Documentation

- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*

Error messages are available only online, or by using Tahiti, the Oracle documentation search tool.

Installation Guides

- *Oracle Diagnostics Pack Installation*

Operating System-Specific Administrative Guides

- *Oracle Database Administrator's Reference, 10g* Release 2 (10.2) *for UNIX Systems*
- *Oracle Database Platform Guide for Microsoft Windows (32-Bit)*
- *Oracle Database System Administration Guide 10g* Release 2 (10.2) *for IBM z/OS (OS/390)*

Oracle Database 10*g* Real Application Clusters Management

- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Database 2 Day DBA*
- *Getting Started with the Oracle Diagnostics Pack*

Generic Documentation

- *Oracle Database New Features*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Database Concepts*
- *Oracle Database Reference*

Printed documentation is available for sale in the Oracle Store at the following Web site:

`http://oraclestore.oracle.com/`

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at the following Web site:

`http://otn.oracle.com/membership/`

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web site at the following Web site:

`http://otn.oracle.com/documentation/`

Oracle error message documentation is available only in HTML. You can browse the error messages by range in the Documentation directory of the installation media. When you find a range, use your browser's "find in page" feature to locate a specific message. When connected to the Internet, you can search for a specific error message using the error message search feature of the Oracle online documentation.

Printed documentation is available for sale in the Oracle Store at the following Web site:

`http://oraclestore.oracle.com/`

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free. Register for OTN at the following Web site:

`http://otn.oracle.com/membership/`

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web Site:

`http://otn.oracle.com/documentation/`

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italics	Italic typeface indicates book titles or emphasis.	*Oracle Database Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.
`UPPERCASE monospace (fixed-width) font`	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a `NUMBER` column. You can back up the database by using the `BACKUP` command. Query the `TABLE_NAME` column in the `USER_TABLES` data dictionary view. Use the `DBMS_STATS.GENERATE_STATS` procedure.
`lowercase monospace (fixed-width) font`	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter `sqlplus` to start SQL*Plus. The password is specified in the `orapwd` file. Back up the data files and control files in the `/disk1/oracle/dbs` directory. The `department_id`, `department_name`, and `location_id` columns are in the `hr.departments` table. Set the `QUERY_REWRITE_ENABLED` initialization parameter to `true`. Connect as `oe` user. The `JRepUtil` class implements these methods.

Convention	Meaning	Example
lowercase italic monospace (fixed-width) font	Lowercase italic monospace font represents placeholders or variables.	You can specify the *parallel_clause*. Run *old_release*.SQL where *old_release* refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[ ]	Brackets enclose one or more optional items. Do not enter the brackets.	DECIMAL (*digits* [, *precision*])
{ }	Braces enclose two or more items, one of which is required. Do not enter the braces, and do not enter more than one option.	{ENABLE | DISABLE}
|	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	{ENABLE | DISABLE} [COMPRESS | NOCOMPRESS]
...	Horizontal ellipsis points indicate either: ■ That we have omitted parts of the code that are not directly related to the example ■ That you can repeat a portion of the code	CREATE TABLE ... AS *subquery*; SELECT *col1*, *col2*, ... , *coln* FROM employees;
. . .	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	SQL> SELECT NAME FROM V$DATAFILE; NAME ---------- /fsl/dbs/tbs_01.dbf /fs1/dbs/tbs_02.dbf . . . /fsl/dbs/tbs_09.dbf 9 rows selected.
Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;
Italics	Italicized text indicates placeholders or variables for which you must supply particular values.	CONNECT SYSTEM/*system_password* DB_NAME = *database_name*

Convention	Meaning	Example

UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	`SELECT last_name, employee_id FROM employees;`
		`SELECT * FROM USER_TABLES;`
		`DROP TABLE hr.employees;`

lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files.
	Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.
		`SELECT last_name, employee_id FROM employees;`
		`sqlplus hr/hr`
		`CREATE USER mjones IDENTIFIED BY ty3MU9;`

What's New in Oracle Database 10*g* Oracle Real Application Clusters Installation and Configuration?

This section describes Oracle Database 10*g* Release 2 (10.2) features as they pertain to the installation and configuration of Oracle Real Application Clusters (RAC). The topic in this section is:

- Oracle Database 10g Release 2 (10.2) New Features for RAC Installation and Configuration

> **Note:** Updates from the preceding 10.*x* releases are preceded by the release number, in parentheses. For example: (10.1). Where there are lists of updates to a feature that contain multiple releases, these releases are preceded by the release number in parentheses. For example:
>
> - (10.1) *feature*
> - (10.2) *feature*

Oracle Database 10*g* Release 2 (10.2) New Features for RAC Installation and Configuration

- Oracle Database 10*g* Release 2 (10.2) is provided with Cluster Verification Utility. Cluster Verification Utility (CVU) is a validation tool that you can use to check whether or not your cluster is properly configured, to avoid installation and database creation failures. It provides the following cluster setup checks:
  - The cluster configuration meets the requirements for installing Oracle Clusterware
  - The cluster configuration meets the requirements for Real Application Clusters installation
  - The cluster configuration meets the requirements for creating a database with Oracle Real Application Clusters, or meets the requirements for a change in database configuration

  You can use the CVU command-line interface:

  - To validate cluster components individually, including node connectivity, and proper configuration of administrative privileges, Oracle Cluster Registry

(OCR), Oracle Clusterware, and other required components for Real Application Clusters

– To validate a pre-defined set of requirements, including the proper setup for node connectivity, shared storage accessibility, integrity of the Oracle Clusterware stack, and other requirements that a system must meet for a specific stage of Oracle Clusterware or RAC database installation or configuration.

CVU is available on the Oracle 10*g* Release 2 (10.2) installation media. You can either run CVU from the installation media, or run it from you local system after installation to perform configuration checks before installing Oracle software. CVU commands to perform installation checks are provided in this installation guide.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for detailed information about CVU

- With Oracle Database 10*g* Release 2 (10.2), you can now create duplicates of cluster nodes (or *cloning* a node) using an image file. In addition, cloning is now the preferred method for adding nodes. The process for cloning, as well as details about using non-interactive (silent) installation scripts, is described in *Oracle Universal Installer and OPatch User's Guide*.

  Both Oracle Clusterware and Oracle Real Application Clusters can be cloned.

- With Oracle Database 10*g* Release 2 (10.2), Cluster Ready Services, or CRS, is now called Oracle Clusterware.

- Oracle Database 10*g* Release 2 (10.2) enables rolling upgrades from Oracle Database 10*g* Release 1 (10.1) Cluster Ready Services to Oracle Database 10*g* Release 2 (10.2) Oracle Clusterware.

  During Oracle Clusterware installation, if Oracle Universal Installer (OUI) detects a 10*g* Release 1 (10.1) Cluster Ready Services installation, then OUI provides the option to install Oracle Clusterware across all nodes in the cluster, or across a subset of nodes in the cluster. During Oracle Clusterware installation, Oracle Database 10*g* Release 1 (10.1) Cluster Ready Services remains available on nodes that are not being upgraded.

- Starting with Oracle Database 10*g* Release 2 (10.2), Oracle Clusterware should be installed in a separate Oracle Clusterware home directory. This is a change to the Optimal Flexible Architecture (OFA) rules. You should not install Oracle Clusterware in a release-specific Oracle home mount point, typically `/u01/app/oracle/product/10.2.0`, as succeeding versions of Oracle Clusterware will overwrite the Oracle Clusterware installation in the same path. Also, If Oracle Clusterware 10*g* Release 2 (10.2) detects an existing Oracle Cluster Ready Services installation, then it overwrites the existing installation in the same path.

- Oracle Database 10*g* Release 2 (10.2) Oracle Clusterware installation provides the option to use a Cluster Configuration File. The Cluster Configuration File simplifies Oracle Clusterware installation in situations such as installing in test environments, or installing Oracle Clusterware on a large number of nodes.

- With Oracle Database 10*g* Release 2 (10.2), Database Configuration Assistant (DBCA) provides a simplified procedure for creating an Automatic Storage Management (ASM) instance and for configuring disk groups.

- With Oracle Database 10*g* Release 2 (10.2), Automatic Storage Management should be installed in a separate ASM home directory. This is a change to the Optimal Flexible Architecture (OFA) rules.
- With Oracle Database 10*g* Release 2 (10.2), during a rolling upgrade of ASM instances on a cluster database, ASM clients continue to be available with no performance affects.

  A rolling upgrade of ASM is when the new ASM software is installed on each node using a "software only" installation, and then the ASM instance is migrated on individual nodes in succession to the new version. As individual nodes are taken down for migration, the other nodes continue to be available.

  Typically, the only noticeable effect on clients using cluster database nodes with ASM during a rolling upgrade is that features of the newer ASM release are unavailable until the rolling upgrade is completed on all nodes.

  To use this new feature, the difference between the older ASM and newer ASM software must be no greater than one patchset version in a major version release number, and the last patchset before a new release. For example, if the last patchset release of Oracle Database 10g release 1 (10.1) is patchset 10.1.0.5, then Oracle Database 10*g* Release 2 (10.2) is compatible with that patchset. Similarly, Oracle Database 10*g* Release 2 (10.2) patchset 10.2.0.1 is compatible with 10.2.0.2, and with the 10.2.0.0 release, but is not compatible with 10.1.0.5.
- With Oracle Database 10*g* Release 2 (10.2), a single ASM instance for each node is now able to serve disk groups to all the database instances in the cluster, whether or not database instances are configured with the RAC parameter turned on in the database parameter file. This change simplifies managing the cluster, because you do not need to distribute disks across multiple ASM instances statically. Instead, you can manage all disks with a single cluster of ASM instances.
- (10.1) Oracle Database 10*g* with RAC is available in both Standard Edition and Enterprise Edition.
- (10.1) Oracle Database 10*g* installation requires you to perform a two-phase process in which you run Oracle Universal Installer (OUI) twice. The first phase installs Oracle Clusterware 10*g* Release 2 (10.2) and the second phase installs the Oracle Database 10*g* software with RAC. The installation also enables you to create and configure services for your RAC environment.

  (10.1) If you have a previous Oracle Database version with RAC or Oracle Parallel Server, then OUI activates Database Upgrade Assistant (DBUA) to automatically upgrade your previous release Oracle Database. DBUA can upgrade the following to Oracle Database 10*g* Release 2 (10.2): Oracle8*i* (8.1.7.4) Oracle Parallel Server databases; Oracle9*i* (9.0.1.4 and higher) and Oracle 9*i* (9.2.0.4 and higher) RAC databases; and can upgrade 10.1.0.2 and higher RAC databases, including those using ASM. It can also upgrade ASM 10.1 to ASM 10.2.

  See Also: *Oracle Database Upgrade Guide* for information about how to plan for upgrading an existing database
- Oracle Database 10*g* Release 2 (10.2) Oracle Clusterware contains the cluster management software required to support Oracle Database 10*g* RAC databases. Oracle Clusterware also provides high availability components that provide many system management features, including determining node availability, cluster group membership, and locking services for Oracle Database processes. The components of Oracle Clusterware interact with third-party vendor clusterware, if present, to coordinate cluster membership information.

In addition, while continuing to be required for RAC databases, Oracle Clusterware is also available for use with single-instance databases and applications that you deploy on clusters. The API libraries required for use with single-instance databases are provided with the Oracle Client installation media.

With this release, the following updates have been made to Oracle Clusterware:

- The Oracle Database cluster manager on database releases previous to 10*g* Release 1 was referred to as "Cluster Manager." In Oracle Database 10*g* releases, the cluster manager role is performed by Cluster Synchronization Services (CSS), a component of Oracle Clusterware, on all platforms. The Cluster Synchronization Service Daemon (CSSD) performs this function.
- With Oracle Database 10*g* Release 2 (10.2) with RAC, Oracle Clusterware provides for the creation of a mirrored Oracle Cluster Registry (OCR) file, enhancing cluster reliability.
- With Oracle Database 10*g* Release 2 (10.2) with RAC, CSS has been modified to allow you to configure CSS with multiple voting disks. In the 10g Release 1 (10.1), you could configure only one voting disk. By enabling multiple voting disk configuration, the redundant voting disks allow you to configure a RAC database with multiple voting disks on independent shared physical disks. This option facilitates the use of the iSCSI network protocol, and other Network Attached Storage (NAS) storage solutions.

  Note: To obtain the benefits of multiple voting disks, you must configure at least three voting disks.

- (10.1) There are new and changed pages and dialog boxes for Oracle Universal Installer (OUI), Database Configuration Assistant (DBCA), and Database Upgrade Assistant (DBUA). The Virtual Internet Protocol Configuration Assistant (VIPCA) is a new tool for the 10.*x* releases. These enhancements are described in the following:
  - (10.1)Database Creation Page: This page lets you select the type of starter database to create, and allows you to configure a database using DBCA after the software installation is complete.
  - (10.1) Database Backup and Recovery Page
  - (10.1) Database Security Page: You can now set the password for the ASM instance SYS user as well as the other database administration and monitoring accounts during installation.
  - (10.1) Database Management Page: This page provides the option to use Oracle Enterprise Manager Grid Control or Database Control to monitor the database.
  - (10.1) Create Disk Group Page: This page lets you create one disk group for ASM database data file management for backup and recovery.
  - (10.1) Existing ASM Disk Groups Page: This page provides assistance with placing disk groups in cases where you already have an ASM instance on your system.
  - (10.1) OUI Cluster Installation Mode Page: This page lets you select whether to perform a cluster or a single-instance Oracle Database 10*g* installation.

- (10.1) SYS and SYSTEM Passwords Page: This page has fields for entering and confirming the SYS and SYSTEM user passwords. This includes SYSMAN and DBSNMP if you use Oracle Enterprise Manager Database Control.
- (10.1) Storage Options Page: This page has storage options for selecting the storage type for the database files such as control files, data files, and redo logs.
- (10.1) DBCA Services Page: This page lets you create and configure services for your RAC environment.
- (10.1) DBCA Initialization Parameters Page: This page has two dialog boxes to display both Basic and Advanced parameter settings.
- (10.1) VIPCA: The pages for this assistant lets you configure virtual Internet protocol addresses for your RAC database.

- (10.1) A new auxiliary, system-managed tablespace called SYSAUX contains performance data and combines content that was stored in different tablespaces (some of which are no longer required) in earlier releases. This is a required tablespace for which you must plan disk space.
- (10.1) The `gsdctl` commands should be used only with Oracle9*i* databases. The Oracle Clusterware installation process stops any existing Global Service Daemon (GSD) processes. To start or stop the GSD processed manually, use `srvctl start nodeapps` or `srvctl stop nodeapps` respectively.
- (10.1) On some platforms, versions of the cluster manager previous to Oracle Database 10*g* were referred to as Cluster Manager. On all platforms in Oracle Database 10*g*, this function is performed by an Oracle Clusterware component known as Cluster Synchronization Services (CSS). Cluster Synchronization Service Daemon (CSSD) performs this function.
- (10.1) Oracle Database 10*g* provides cluster file system support for Linux and Windows-based platforms.

  See Also: For more information about Oracle Cluster File System on Linux, refer to the Linux Technology Center on OTN, which is located at the following URL:

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

- (10.1) RAC and Database Configuration Assistant (DBCA) support Automatic Storage Management (ASM) and Oracle Managed Files (OMF).

  See Also:

  - *Oracle Database Administrator's Guide* for information about Automatic Storage Management, a new database file management feature
  - *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about administering services and storage in RAC
  - *Oracle Database Upgrade Guide* for information about using DBUA

- (10.2) The Oracle Database 10*g* version of the `srvConfig.loc` file is the `ocr.loc` file. The Oracle9*i* version of `srvConfig.loc` still exists for backward compatibility.

Note: If OUI detects the Oracle9*i* `srvrconfig.loc` file, then the Oracle9*i* `srvConfig.loc` file is modified to point to `/dev/null`. Changing the `srvConfig.loc` file prevents Oracle9*i* clients such as `GSD` and `srvconfig` from overwriting contents of the device used for Oracle Cluster Registry in Oracle 10*g* Release 2 (10.2).

Part I

Oracle Clusterware and Oracle Real Application Clusters Installation Planning and Requirements

Part I describes how to plan your Oracle Clusterware and Oracle Real Application Clusters (RAC) installations, and describes Oracle Clusterware and RAC installation requirements. Part I contains the following topics:

Chapter 1, "Introduction to Installing and Configuring Oracle Clusterware and Oracle Real Application Clusters"

1

Introduction to Installing and Configuring Oracle Clusterware and Oracle Real Application Clusters

This chapter provides an overview of Oracle Clusterware and Oracle Real Application Clusters (RAC) installation and configuration procedures. It includes the following topics:

- Oracle Clusterware and Oracle Real Application Clusters Documentation Overview
- General System Installation Requirements
- Configuration Tasks for Oracle Clusterware and Oracle Real Application Clusters
- Pre-Installation, Installation, and Post-Installation Overview
- Oracle Universal Installer and Real Application Clusters
- Storage Considerations for Installing Oracle Database 10g Real Application Clusters
- Additional Considerations for Using Oracle Database 10g Features in RAC
- Oracle Database 10g and Real Application Clusters Components
- Oracle Database 10g Real Application Clusters Release Compatibility
- Required Operating System Groups
- Cloning Oracle Clusterware and RAC in Grid Environments

Oracle Clusterware and Oracle Real Application Clusters Documentation Overview

This section describes the Oracle Clusterware and RAC documentation set.

This book contains the pre-installation, installation, and post-installation information for Linux. Additional information for this release may be available in the Oracle Database 10*g* README or Release Notes. The platform-specific Oracle Database 10*g* installation media contains a copy of this book in both HTML and PDF formats.

The Server Documentation directory on the installation media contains *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*

Oracle Real Application Clusters Administration and Deployment Guide

Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide describes how to administer Oracle Clusterware components such as the voting disks and Oracle Cluster Registry (OCR) devices. This book also explains how to administer storage, how to use RAC scalability features to add and delete instances and nodes, how to use Recovery Manager (RMAN), and how to perform backup and recovery in RAC.

Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide describes RAC deployment topics such as services, high availability, and workload management. The book describes how the Automatic Workload Repository (AWR) tracks and reports service levels, and how you can use service level thresholds and alerts to improve high availability in your RAC environment. The book also describes how to make your applications highly available using Oracle Clusterware.

Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide also provides information about how to monitor and tune performance in RAC environments by using Oracle Enterprise Manager, and by using information in AWR and Oracle Database performance views. This book also provides some application-specific deployment techniques for online transaction processing and data warehousing environments.

General System Installation Requirements

Each node that you want to make part of your Oracle Clusterware or Oracle Clusterware and RAC installation must meet the hardware and software requirements specified in Part II of this book. You can use the new Cluster Verification Utility to assist you with verification of requirements.

If you are uncertain about concepts related to setting up and configuring a RAC database, then read*Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* to inform yourself about concepts such as services, setting up storage, and other information relevant to configuring your cluster.

Overview of Cluster Verification Utility

Cluster Verification Utility (CVU) is provided with Oracle Clusterware and Oracle Database 10*g* Release 2 (10.2) with Real Application Clusters. The purpose of CVU is to enable you or your hardware vendors to verify during setup and configuration that all components required for a successful installation of Oracle Clusterware or Oracle Clusterware and a RAC database are installed and configured correctly, and to provide you with ongoing assistance any time you need to make changes to your RAC cluster. You are provided with commands to use the CVU to verify completion of tasks in this guide.

There are two types of CVU commands:

- **Stage Commands** are CVU commands used to test system setup and readiness for successful software installation, database creation, or configuration change steps. These commands are also used to validate successful completion of specific cluster configuration steps.
- **Component Commands** are CVU commands used to check individual cluster components, and determine their state.

This guide provides stage and component CVU commands where appropriate to assist you with cluster verification.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for detailed information about Cluster Verification Utility.

Overview of Oracle Universal Installer

Oracle Universal Installer (OUI) is a graphical user interface (GUI) tool that assists you with installing and configuring Oracle Database. It can be run using different command options to perform installation pre-checks, specialized installation processes, and other tasks. To see an overview of OUI options, navigate to the directory path `oui/bin` in the Oracle home directory, and type the following command:

```
$ ./runInstaller -help
```

See Also: *Oracle Universal Installer and OPatch User's Guide* for more detailed information about OUI options

Overview of Upgrading and Multiple-release Systems

The path that you must take to upgrade to the new Oracle Database 10g release depends on the release number of your current database. It may not be possible to upgrade directly from your current release of Oracle Database to the latest release. Depending on your current release, you may need to upgrade through one or more intermediate releases to upgrade to the new Oracle Database 10g release.

For example, if the current database is running release 8.1.6, then first upgrade to release 8.1.7 using the instructions in Oracle8*i* Migration for release 8.1.7. The release 8.1.7 database can then be upgraded to the new Oracle Database 10g release.

Oracle9*i* database can coexist with Oracle Database 10*g* Release 2 (10.2). However, if you want separate releases of the database to coexist, then you must install Oracle Database 10*g* with Oracle9*i* already installed. You should not install Oracle9*i* after installing Oracle10*g*.

See Also: *Oracle Database Upgrade Guide* for more information about upgrading

Hardware Requirements for Oracle Clusterware and Oracle Real Application Clusters

Each node in a cluster requires the following hardware:

- External shared disks for storing the Oracle Clusterware (Oracle Cluster Registry and voting disk) files, and database files.

  Chapter 3 describes the storage disk configuration options that are available. Review these options before you decide which storage option to use in your RAC environment. However, note that when Database Configuration Assistant (DBCA) configures automatic disk backup, it uses a database recovery area that must be shared.

Note: Oracle Clusterware software cannot be installed on Oracle Cluster File System (OCFS). Oracle Clusterware can be installed on network-attached storage (NAS).

- One private internet protocol (IP) address for each node to serve as the private interconnect. The following must be true for each private IP address:
  - It must be separate from the public network
  - It must be accessible on the same network interface on each node
  - It must have a unique address on each node

  The private interconnect is used for internode communication by both Oracle Clusterware and RAC. The private IP address must be available in each node's `/etc/hosts` file.

  During Oracle Clusterware installation, the information you enter as the private IP address determines which private interconnects are used by Oracle Clusterware for its own communication. They must all be available, and capable of responding to a `ping` command.

  Oracle recommends that you use a logical Internet Protocol (IP) address that is available across all private networks, and that you take advantage of any available operating system-based failover mechanism by configuring it according to your third-party vendor's instructions for using their product to support failover.
- One public IP address for each node, to be used as the Virtual IP address for client connections and for connection failover.

  During installation this public virtual IP address (VIP) is associated with the same interface name on every node that is part of your cluster. The IP addresses that you use for all of the nodes that are part of a cluster must be from the same subnet. If you have a domain name server (DNS), then register the host names for the VIP with the DNS. The VIP should not be in use at the time of the installation, because this is a VIP that Oracle Clusterware manages.
- One public fixed host name address for each node, typically assigned by the system administrator during operating system installation. If you have a domain name server (DNS), then you can register both the fixed IP and the VIP address with the DNS. If you do not have a DNS, then you must make sure that both public IP addresses are in the node `etc/host` file (for all cluster nodes), and any client system's `etc/host` file that requires access to the database.

Note: In addition to these requirements, Oracle recommends the following:

- While installing and using Real Application Clusters software, you should attempt to keep the system clocks on all cluster nodes as close as possible to the same time. Oracle strongly recommends using the Network Time Protocol feature of most operating systems for this purpose, with all nodes using the same reference Network Time Protocol server.
- Use redundant switches as a standard configuration for all cluster sizes.

Software Requirements for Oracle Clusterware and Oracle Real Application Clusters

Each node in a cluster requires a supported interconnect software protocol to support Cache Fusion, and to support Oracle Clusterware polling. Your interconnect *must* be certified by Oracle for your platform. You should also have a Web browser, both to enable Oracle Enterprise Manager, and to view online documentation.

For Oracle Database 10*g* requirements, Oracle Clusterware provides the same functions as third-party vendor clusterware. Using Oracle Clusterware also reduces installation and support complications. However, you may require third-party vendor clusterware if you use a non-ethernet interconnect, or if you have deployed clusterware-dependent applications on the same cluster where you deploy RAC.

Configuration Tasks for Oracle Clusterware and Oracle Real Application Clusters

Before installing RAC, perform the following procedures:

1. Ensure that you have a certified combination of the operating system and an Oracle Database software release by referring to the Oracle*MetaLink* certification information, which is located at the following Web site:

   `http://metalink.oracle.com`

   Click **Certify & Availability**, and select **1.View Certifications by Product**.

   Note: The layout of the Oracle*MetaLink* site and the site's certification policies are subject to change.

2. Configure a high-speed interconnect that uses a private network. Some platforms support automatic failover to an additional interconnect.

3. Determine the storage option for your system and configure the shared disk. Oracle recommends that you use Automatic Storage Management (ASM) and Oracle Managed Files (OMF), or a cluster file system. If you use ASM or a cluster file system, then you can also take advantage of OMF and other Oracle Database 10*g* storage features. If you use RAC on Oracle Database 10*g* Standard Edition, then you must use ASM.

   When you start Oracle Universal Installer (OUI) to install Oracle Clusterware, you are asked to provide to provide the paths for voting disks, and for the Oracle Cluster Registry (OCR).

   For voting disks: Configure one disk, if you have existing redundancy support for the voting disk. If you intend to use multiple voting disks managed by Oracle Clusterware, then you must have at least three disks to provide sufficient redundancy, and you must ensure that each voting disk is located on physically independent storage.

   In addition, if you select multiple voting disks managed by Oracle Clusterware, then you should ensure that all voting disks are located on a secure network protected from external security threats, and you should ensure that all voting disks are on regularly maintained systems. If a voting disk fails, then you need to fix the physical hardware and bring it back online. The Cluster Synchronization Services (CSS) component of Oracle Clusterware continues to use the other voting disks, and automatically makes use of the restored drive when it is brought online again.

   For OCR: Configure one disk if you have existing redundancy support. If you intend to use OCR mirroring managed by Oracle Clusterware, then you must have two OCR locations, and you must ensure that each OCR is located on physically independent storage.

In addition, if you select mirrored OCRs managed by Oracle Clusterware, then you should ensure that all OCRs are located on a secure network protected from external security threats, and you should ensure that all OCRs are on regularly maintained systems. If an OCR copy fails or becomes inaccessible, then you can use the `ocrconfig` tool to replace the OCR.

4. Install the operating system patch updates that are listed in the pre-installation chapter in this book in Part II.
5. Use the Cluster Verification Utility (CVU) to help you to verify that your system meets requirements for installing Oracle Database with RAC.

Pre-Installation, Installation, and Post-Installation Overview

The following describes the installation procedures that are covered in Part II and Part III of this book.

Pre-Installation Overview for Oracle Database 10*g* Real Application Clusters

The pre-installation procedures in Part II explain how to verify user equivalence, perform network connectivity tests, how to set directory and file permissions, and other required pre-installation tasks. Complete all pre-installation tasks and verify that your system meets all pre-installation requirements before proceeding to the install phase.

Installation Overview for Oracle Database 10*g* Real Application Clusters

Oracle Database 10*g* Real Application Clusters installation is a two-phase installation. In phase one, use Oracle Universal Installer (OUI) to install Oracle Clusterware as described in Chapter 4, "Installing Oracle Clusterware". Note that the Oracle home in phase one is a home for the Oracle Clusterware software, which *must be different* from the Oracle home that you use in phase two for the installation of the Oracle database software with RAC components. The Oracle Clusterware installation starts the Oracle Clusterware processes in preparation for installing Oracle Database 10*g* with RAC, as described in Chapter 5, "Installing Oracle Database 10g with Real Application Clusters". Use OUI in this phase to install the RAC software.

You must install Oracle Clusterware and Oracle Database in a separate home directories. If you will use multiple Oracle Database homes with ASM, then you should install a separate Oracle Database home for ASM. You should create the listener in the Oracle database Oracle home.

If OUI detects a previous release of Oracle Clusterware (previously known as Oracle Cluster Ready Services), then you are prompted to select either a rolling upgrade, or a full upgrade.

If OUI detects a previous release of the Oracle database, then OUI provides you with the option to start Database Upgrade Assistant (DBUA) to upgrade your database to Oracle Database 10*g* Release 2 (10.2). In addition, DBUA displays a Service Configuration page for configuring services in your RAC database.

> **See Also:** *Oracle Database Upgrade Guide* for additional information about preparing for upgrades

After the installation completes, OUI starts the Oracle Database assistants, such as Database Configuration Assistant (DBCA), to configure your environment and create your RAC database. You can later use the DBCA Instance Management feature to add

or modify services and instances as described in Chapter 6, "Creating Real Application Clusters Databases with Database Configuration Assistant".

Post-Installation Overview for Oracle Database 10*g* Real Application Clusters

After you create your database, download and install the most recent patch set for your Oracle Database 10*g* release, as described in Chapter 8, "Oracle Real Application Clusters Post-Installation Procedures". If you are using other Oracle products with your RAC database, then you must also configure them.

You must also perform several post-installation configuration tasks to use certain Oracle Database 10*g* features.

On the installation media, you can select additional Oracle Database 10*g* software that may improve performance or extend database capabilities. Examples: Oracle JAccelerator, Oracle *inter*Media, and Oracle Text.

> **See Also:** *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*, and *Oracle Universal Installer and OPatch User's Guide* for more information about using the RAC scalability features of adding and deleting nodes and instances from RAC databases

Oracle Universal Installer and Real Application Clusters

Oracle Universal Installer (OUI) facilitates the installation of Oracle Clusterware and Oracle Database 10*g* software. In most cases, you use the graphical user interface (GUI) provided by OUI to install the software. However, you can also use OUI to complete non-interactive (or *silent*) installations, without using the GUI. Refer to Appendix B for information about non-interactive installations.

The Oracle Inventory maintains records of Oracle software releases and patches. Each installation has a central inventory where the Oracle home is registered. Oracle software installations have a local inventory directory, whose path location is recorded in the central inventory Oracle home. The local inventory directory for each Oracle software installation contains a list of components and applied interim patches associated with that software. Because your Oracle software installation can be corrupted by faulty inventory information, OUI must perform all read and write operations on Oracle inventories.

When you install Oracle Clusterware or RAC, OUI copies this Oracle software onto the node from which you are running it. If your Oracle home is not on a shared file system, then OUI propagates the software onto the other nodes that you have selected to be part of your OUI installation session. The Oracle Inventory maintains a list of each node that is a member of the RAC database, and lists the paths to each node's Oracle home. This is used to maintain software patches and updates for each member node of the RAC database.

If you create your RAC database using OUI, or if you create it later using DBCA, then Oracle Enterprise Manager Database Control is configured for your RAC database. Database Control can manage your RAC database, all its instances, and the hosts where instances are configured.

You can also configure Enterprise Manager Grid Control to manage multiple databases and application servers from a single console. To manage RAC databases in Grid Control, you must install a Grid Control agent on each of the nodes of your cluster. The Agent installation is designed to recognize a cluster environment and install

across all cluster nodes; you need to perform the installation on only one of the cluster nodes to install Grid Control agent on all cluster nodes.

When OUI installs the Oracle Clusterware or Oracle Database software, Oracle recommends that you select a preconfigured database, or use Database Configuration Assistant (DBCA) interactively to create your RAC database. You can also manually create your database as described in procedures posted on the Oracle Technical Network, which is at the following URL:

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

Oracle recommends that you use Automatic Storage Management (ASM). If you are not using ASM, or if you are not using a cluster file system or an NFS system, then configure shared raw devices before you create your database.

See Also:

- *Oracle Universal Installer and OPatch User's Guide* for more details about OUI
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for information about using Oracle Enterprise Manager to administer RAC environments
- The Grid Technology Center on the Oracle Technology Network (OTN), which is available at the following URL:

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

Storage Considerations for Installing Oracle Database 10*g* Real Application Clusters

This section discusses storage configuration options that you should consider before installing Oracle Database 10*g* Release 2 (10.2) with Real Application Clusters.

Overview of Automatic Storage Management

Oracle recommends using Automatic Storage Management (ASM) or a cluster file system with Oracle Managed Files (OMF) for database storage. This section provides an overview of ASM.

Note that RAC installations using Oracle Database Standard Edition must use ASM for database file storage.

You can use ASM to simplify the administration of Oracle database files. Instead of having to manage potentially thousands of database files, using ASM, you need to manage only a small number of disk groups. A **disk group** is a set of disk devices that ASM manages as a single logical unit. You can define a particular disk group as the default disk group for a database, and Oracle Database will automatically allocate storage for, create, or delete, the files associated with the appropriate database object. When administering the database, you need to refer to database objects only by name, rather than by file name.

When using ASM with a single Oracle home for database instances on a node, the ASM instance can run from that same home. If you are using ASM with Oracle Database instances from multiple database homes on the same node, then Oracle recommends that you run the ASM instance from an Oracle home that is distinct from the database homes. In addition, the ASM home should be installed on every cluster node. Following this recommendation prevents the accidental removal of ASM

instances that are in use by databases from other homes during the de-installation of a database's Oracle home.

Benefits of Automatic Storage Management

ASM provides many of the same benefits as storage technologies such as a redundant array of independent disks (RAID) or a logical volume manager (LVM). Like these technologies, ASM lets you create a single disk group from a collection of individual disk devices. It balances input and output (I/O) loads to the disk group across all of the devices in the disk group. It also implements striping and mirroring to improve I/O performance and data reliability.

However, unlike RAID or LVMs, ASM implements striping and mirroring at the file level. This implementation lets you specify different storage attributes for individual files in the same disk group.

Disk Groups and Failure Groups

A disk group can contain between 1 to 10000 disk devices. Each disk device can be an individual physical disk, a multiple disk device such as a RAID storage array or logical volume, or even a partition on a physical disk. However, in most cases, disk groups consist of one or more individual physical disks. To enable ASM to balance I/O and storage appropriately within the disk group, all devices in the disk group should have similar, if not identical, storage capacity and performance.

Note: Do not put more than one partition of a single disk into the same disk group. You can put separate partitions of a single disk into separate disk groups.

Logical volume managers are not supported on Linux.

When you add a device to a disk group, you can specify a failure group for that device. Failure groups define ASM disks that share a common potential failure mechanism. An example of a failure group is a set of SCSI disks sharing the same SCSI controller. Failure groups are used to determine which ASM disks to use for storing redundant copies of data. For example, if two-way mirroring is specified for a file, then ASM automatically stores redundant copies of file extents in separate failure groups. Failure groups apply only to normal and high redundancy disk groups. You define the failure groups in a disk group when you create or alter the disk group.

Redundancy Levels

ASM provides three levels of mirroring, called redundancy levels, that you can specify when creating a disk group. The redundancy levels are:

- External redundancy

  In disk groups created with external redundancy, the contents of the disk group are not mirrored by ASM. Choose this redundancy level when:

  - The disk group contains devices, such as RAID devices, that provide their own data protection
  - Your use of the database does not require uninterrupted access to data. For example: a development environment where you have a suitable backup strategy

- Normal redundancy

In disk groups created with normal redundancy, the contents of the disk group are two-way mirrored by default. However, you can choose to create certain files that are three-way mirrored, or that are not mirrored. To create a disk group with normal redundancy, you must specify at least two failure groups (a minimum of two devices).

The effective disk space of a disk group that uses normal redundancy is half the total disk space of all of its devices.

- High redundancy

  In disk groups created with high redundancy, the contents of the disk group are three-way mirrored by default. To create a disk group with high redundancy, you must specify at least three failure groups (a minimum of three devices).

  The effective disk space of a disk group that uses high redundancy is one-third of the total disk space of all of its devices.

See Also: *Oracle Database Administrator's Guide* for additional information about ASM and redundancy

ASM and Installation Types

The type and number of disk groups that you can create when installing Oracle Database software depends on the type of database you choose to create during the installation, as follows:

- Preconfigured database

  If you choose to create the default preconfigured database that uses ASM, then OUI prompts you to specify one or more disk device names and redundancy. By default, OUI creates a disk group named DATA, with normal redundancy.

- Advanced database

  If you choose to create an advanced database that uses ASM, then you can create one or more disk groups. These disk groups can use one or more devices. For each disk group, you can specify the redundancy level that suits your requirements.

- Configure Automatic Storage Management

  If you choose to create an ASM instance only, then OUI prompts you to create Disk Group. If OUI finds a Grid Control service is found on the system, then OUI prompts you to indicate if the ASM instance shall be managed by Grid Control. The Management Service box lists the available Oracle Management Services.

Shared Storage for Database Recovery Area

When you configure a database recovery area in a RAC environment, the database recovery area must be on shared storage. When Database Configuration Assistant (DBCA) configures automatic disk backup, it uses a database recovery area that must be shared.

If the database files are stored on a cluster file system, then the recovery area can also be shared through the cluster file system.

If the database files are stored on an Automatic Storage Management (ASM) disk group, then the recovery area can also be shared through ASM.

Note: ASM disk groups are always valid recovery areas, as are cluster file systems. Recovery area files do not have to be in the same location where data files are stored. For instance, you can store data files on raw devices, but use ASM for the recovery area.

Additional Considerations for Using Oracle Database 10*g* Features in RAC

Oracle recommends that you use the following Oracle Database 10*g* features to simplify RAC database management:

- Oracle Enterprise Manager—Use Enterprise Manager to administer your *entire* processing environment, not just the RAC database. Enterprise Manager lets you manage a RAC database with its instance targets, listener targets, host targets, and a cluster target, as well as ASM targets if you are using ASM storage for your database.
- Automatic undo management—This feature automatically manages undo processing.
- Automatic segment-space management—This feature automatically manages segment freelists and freelist groups.
- Locally managed tablespaces—This feature enhances space management performance.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about these features in RAC environments

Oracle Database 10*g* and Real Application Clusters Components

Oracle Database 10*g* provides single-instance database software and the additional components to operate RAC databases. Some of the RAC-specific components include the following:

- Oracle Clusterware
- A RAC-enabled Oracle home

Oracle Clusterware

OUI installs Oracle Clusterware on each node of the cluster. If third-party vendor clusterware is not present, then you must use OUI to enter the nodes on which you want Oracle Clusterware to be installed. The Oracle Clusterware home can either be shared by all nodes, or private to each node, depending on your responses when you run OUI. The home that you select for Oracle Clusterware *must* be different from the RAC-enabled Oracle home.

When third-party vendor clusterware is present, Oracle Clusterware may interact with the third-party vendor clusterware. For Oracle Database 10*g* on Linux and Windows, Oracle Clusterware coexists with but does not interact with previous Oracle clusterware releases. In using third-party vendor clusterware, note the following:

- Oracle Clusterware can integrate with third-party vendor clusterware for all operating systems except Linux and Windows.

Note: The Oracle Database cluster manager in database releases previous to 10*g* Release 1 was referred to as "Cluster Manager." In Oracle Database 10*g*, the cluster manager role is performed by Cluster Synchronization Services (CSS), a component of Oracle Clusterware, on all platforms. The Cluster Synchronization Service Daemon (CSSD) performs this function.

The Installed Real Application Clusters Components

All instances in RAC environments share the control file, server parameter file, redo log files, and all data files. These files reside on a shared cluster file system or on shared disks. Either of these types of file configurations are accessed by all the cluster database instances. Each instance also has its own set of redo log files. During failures, shared access to redo log files enables surviving instances to perform recovery.

Oracle Database 10*g* Real Application Clusters Release Compatibility

You can install and operate different releases of Oracle Database software on the same computer:

- With Oracle Database 10*g* Release 2 (10.2) if you have an existing Oracle home, then you must install the database into the existing Oracle home. You should install Oracle Clusterware in a separate Oracle Clusterware home. Each node can have only one Oracle Clusterware home.

  During installation, Oracle Universal Installer (OUI) prompts you to install additional Oracle Database 10*g* components if you have not already installed all of them.
- OUI lets you de-install and re-install Oracle Database 10*g* Real Application Clusters if needed.
- If you want to install Oracle9*i* and Oracle Database 10*g* Release 2 (10.2) on the same system, then you must install Oracle9i first. You cannot install Oracle9i on a system with Oracle Database 10*g*.
- If OUI detects an earlier database release, then OUI asks you about your upgrade preferences. You have the option to upgrade one of the previous release databases with DBUA or to create a new database using DBCA. The information collected during this dialog is passed to DBUA or DBCA after the software is installed.

  Note: Do not move Oracle binaries from the Oracle home to another location. Doing so can cause dynamic link failures.

- You can run different releases of Oracle Database and Automatic Storage Management (ASM). If the Oracle Database release and the ASM release are the same release, then they can both run out of the same Oracle home. If they are different releases, then the Oracle Database release and the ASM release must be in their separate release homes. For example, you can install an ASM release 10g Release 2 (10.2) instance and use it with Oracle Database 10g Release 1 (10.1) database, or you can install an Oracle 10g Release 2 (10.2) database and use it with an ASM 10g Release 1 (10.1) instance.

Note: When using different release ASM and Oracle Database releases, the functionality of each is dependent on the functionality of the earlier software release. For example, an Oracle Database 10*g* release 10.1.0.2 using an ASM 10.1.0.3 instance will not be able to use new features available for ASM in the 10.1.0.3 release, but instead only ASM 10.1.0.2 features. Conversely, an Oracle Database 10*g* release 10.1.0.3 using an ASM instance release 10.1.0.2 will function like a release 10.1.0.2 database.

Required Operating System Groups

Depending on whether this is the first time that you are installing Oracle server software on your system, you may need to create several groups and a user account to own Oracle software, as described later in the pre-installation procedures. The required groups and user are:

- The Oracle Inventory group (`oinstall`)

  You must create this group the first time you install Oracle software on the system. The usual name for this group is `oinstall`. Members of this group own the Oracle inventory, which is a catalog of all of the Oracle software installed on the system. Group membership in `oinstall` is also required to perform some tasks involving Oracle Cluster Registry (OCR) keys that are created during Oracle Clusterware installation.

- The OSDBA group (`dba`)

  You must create the OSDBA group the first time you install Oracle software on the system.

  The OSDBA group provides operating system verification of users that have database administrative privileges (the SYSDBA and SYSOPER privileges). The default name for this group is `dba`. During installation, if you want to specify a group name other than the default, then you are prompted during installation for the name of the OSDBA group.

  You must create a new OSDBA group if you have an existing OSBDA group, but you want to give a different group of users database administrative privileges in a new Oracle server installation.

- The OSOPER group (`oper`)

  The OSOPER group is optional. Create this group if you want a separate group of users to have a limited set of database administrative privileges (the SYSOPER privilege), to perform such operations as backing up, recovering, starting up, and shutting down the database. The default name for this group is `oper`. To use this group, choose the **Custom** installation type to install the software. To use an OSOPER group, you must create it in the following circumstances:

  - If an OSOPER group does not exist (for example, if this is the first installation of Oracle server software on the system)
  - If an OSOPER group exists, but you want to give a different group of users database operator privileges in a new Oracle server installation

- The Oracle Software Owner user (`oracle`)

  You must create the Oracle user the first time you install Oracle software on the system. The Oracle user owns all of the software installed during the installation. The usual name for the Oracle user is `oracle`. The Oracle user must have the

Oracle Inventory group as its primary group and the OSDBA group as its secondary group. It must also have the OSOPER group as a secondary group if you choose to create that group.

If an Oracle software owner user exists, but you want to use a different user with different group membership in a new Oracle server installation, then you must give database administrative privileges to those groups.

A single Oracle Inventory group is required for all installations of Oracle software on the system. However, you can create different Oracle software owner users, OSDBA groups, and OSOPER groups (other than `oracle`, `dba`, and `oper`) for separate installations. In addition, you can create a separate owner for Oracle Clusterware. Using different groups lets you grant `DBA` privileges to a particular operating system user on one database, which they would not have on another database on the same system.

> **See Also:** *Oracle Database Administrator's Reference, 10g* Release 2 (10.2) *for UNIX Systems* and *Oracle Database 10g Administrator's Guide* for Linux for additional information about the OSDBA and OSOPER groups, and the `SYSDBA` and `SYSOPER` privileges.

Cloning Oracle Clusterware and RAC in Grid Environments

The following section provides a summary of the procedure for deployments of RAC in grid environments with large numbers of nodes using cloned Clusterware and RAC images.

> **See Also:** For detailed information about cloning RAC and Oracle Clusterware images, refer to the following documents:
>
> Cloning, and adding and deleting nodes:
>
> *Oracle Universal Installer and OPatch User's Guide*
>
> Additional information about adding and deleting nodes:
>
> *Oracle10i Real Application Clusters Administration*

This section contains the following:

- Cloning Oracle Clusterware Homes
- Cloning RAC Homes

Cloning Oracle Clusterware Homes

Complete the following tasks to clone an Oracle Clusterware home on multiple nodes:

1. On the source node, install Oracle Clusterware software. All required root scripts must run successfully.
2. As root, create a tar file of the Oracle Clusterware home
3. On the target node, create an Oracle Clusterware home, and copy the Oracle Clusterware tar file from the source node to the target node Oracle Clusterware home.
4. As root, uncompress the tar file.
5. Run OUI in clone mode, as described in *Oracle Universal Installer and OPatch User's Guide*.

6. Run root scripts.
7. Repeat steps 1 through 6 on each node that you want to add to the cluster. On the last node that you install, run the tool `oifcfg` to configure the network interfaces.

Cloning RAC Homes

Complete the following tasks to Clone a RAC database image on multiple nodes:

1. On the source node, install a RAC database Oracle home. All required root scripts must run successfully. Do not create a database, and do not run any configuration tools.
2. As root, create a tar file of the RAC database Oracle home.
3. On the target node, create an Oracle home directory for the RAC database, and copy the RAC database tar from the source node to the target node Oracle home.
4. Create required Oracle users and groups, ensuring that you use the same names, user ID numbers and group ID numbers as those on the source node.
5. As root, uncompress the tar file.
6. Run OUI in clone mode, as described in *Oracle Universal Installer and OPatch User's Guide*.
7. Run root scripts.
8. Repeat steps 1 through 7 for on each node that you want to add to the cluster.
9. Run the configuration assistant NetCA on a local node of the cluster, and provide a list when prompted of all nodes that are part of the cluster. This procedure creates the listener.
10. Run the configuration assistant DBCA to create the database.
11. Follow post-cloning phase instructions as provided in *Oracle Universal Installer and OPatch User's Guide*.

Part II

Oracle Clusterware and Oracle Real Application Clusters Pre-Installation Procedures

Part II describes the pre-installation procedures for Linux that you must complete before installing Oracle Clusterware and Oracle Database 10*g* Real Application Clusters (RAC). Part II contains the following chapters:

- Chapter 2, "Pre-Installation Tasks"
- Chapter 3, "Configuring Oracle Clusterware and Oracle Database Storage"

2

Pre-Installation Tasks

This chapter describes the system configuration tasks that you must complete before you start Oracle Universal Installer. It includes information about the following tasks:

- Understanding and Using Cluster Verification Utility
- Logging In to the System as root
- Creating Required Operating System Groups and User
- Configuring the oracle User's Environment
- Checking the Hardware Requirements
- Checking the Network Requirements
- Checking the Network Setup with CVU
- Checking the Hardware and Operating System Setup with CVU
- Checking the Operating System Requirements Setup with CVU
- Identifying Required Software Directories
- Identifying or Creating an Oracle Base Directory
- Creating the Oracle Clusterware Home Directory
- Stopping Existing Oracle Processes for Upgrades or Co-existing Databases

Understanding and Using Cluster Verification Utility

Cluster Verification Utility (CVU) is a tool that performs system checks. This guide provides CVU commands to assist you with confirming that your system is properly configured for Oracle Clusterware and Oracle Real Application Clusters installation.

This section describes the following topics:

- Prerequisites for Using Cluster Verification Utility
- Using CVU to Determine if Installation Prerequisites are Complete
- Entering Cluster Verification Utility Commands
- Using the Cluster Verification Utility Help
- Using Cluster Verification Utility with Oracle 10g Release 1
- Verbose Mode and "Unknown" Output

Prerequisites for Using Cluster Verification Utility

You must have JDK 1.4.2 installed on your system before you can run CVU.

If you do not have JDK 1.4.2 installed on your system, and you attempt to run CVU, then you receive an error message similar to the following:

```
ERROR. Either CV_JDKHOME environment variable should be set
       or /stagepath/cluvfy/jrepack.zip should exist.
```

To correct this error, use the following command to determine if you have an installed JDK 1.4.2 home:

```
$ find . -name "*j2*" -print | more
```

This command searches for the directory name created by the Sun bin script. You can also search for other names, such as "JDK" or "JRE". If you do not have JDK 1.4.2 installed, then download it from the Sun Web site, and use the Sun instructions to install it. JDK 1.4.2 is available as a download from the following Web site:

```
http://www.sun.com/java
```

If you do have JDK 1.4.2 installed, then you must define the user environment variable CV_JDKHOME for the path to the JDK.

For example, if JDK 1.4.2 is installed in `/usr/local/j2re1.4.2_08`, then log in as the user that you plan to use to run CVU, and enter the following commands

```
CV_JDKHOME=/usr/local/j2re1.4.2_08
export CV_JDKHOME
```

Entering Cluster Verification Utility Commands

To enter a CVU command, use the following syntax:

```
$ /mountpoint/clusterware/cluvfy/runcluvfy.sh options
```

In the preceding syntax example, the variable *mountpoint* represents the mountpoint path for the installation media, and the variable *`options`* represents the CVU command options that you select. For example:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh comp nodereach -n node1,node2 -verbose
```

By default, when you enter a CVU command, CVU provides a summary of the test. During pre-installation, Oracle recommends that you obtain detailed output by using the `-verbose` argument with the CVU command. The `-verbose` argument produces detailed output of individual checks. Where applicable, it shows results for each node in a tabular layout.

Using CVU to Determine if Installation Prerequisites are Complete

You can use CVU to determine which system prerequisites for installation are already completed. Use this option if you are installing Oracle 10*g* Release 2 (10.2) on a system with a pre-existing Oracle software installation. In using this option, note the following:

- You must complete the prerequisites for using CVU
- CVU can assist you by finding pre-installation steps that need to be completed, but it cannot perform pre-installation tasks

Use the following syntax to determine what pre-installation steps are completed, and what pre-installation steps must be performed

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh stage -pre crsinst -n node_list
```

In the preceding syntax example, replace the variable *mountpoint* with the installation media mountpoint, and replace the variable *node_list* with the names of the nodes in your cluster, separated by commas.

For example, for a cluster with mountpoint /dev/dvdrom/, and with nodes node1, node2, and node3, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh stage -pre crsinst -n
node1,node2,node3
```

Review the CVU report, and proceed to the sections of the pre-installation chapter to complete additional steps as needed.

Using the Cluster Verification Utility Help

The `cluvfy` commands have context-sensitive help that shows correct syntax usage based on the command line arguments that you enter.

If you enter an invalid CVU command, then CVU shows the correct usage for that command. For example, if you type `runcluvfy.sh stage -pre dbinst`, then CVU shows the correct syntax for the database pre-installation checks that CVU performs with the `dbinst` stage option. The following is a list of context help commands.

- `cluvfy`: CVU displays high-level generic usage text describing the stage and component syntax.
- `cluvfy -help`: CVU displays detailed CVU command information.
- `cluvfy comp -list`: CVU displays a list of components that can be checked, and brief descriptions of how each component is checked.
- `cluvfy comp -help`: CVU displays detailed syntax for each of the valid component checks.
- `cluvfy stage -list`: CVU displays a list of valid stages.
- `cluvfy stage -help`: CVU displays detailed syntax for each of the valid stage checks.

Using Cluster Verification Utility with Oracle 10g Release 1

You can use CVU on the Oracle 10*g* Release 2 (10.2) media to check system requirements for Oracle 10*g* Release 1 (10.1) installations. To use CVU to check 10. 1 installations, append the command flag -r 10gR1 to the standard CVU system check commands.

For example, to perform a verification check for a Cluster Ready Services 10. 1 installation, on a system where the media mountpoint is /dev/dvdrom/, and the cluster nodes are node1, node2, and node 3, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh stage -pre crsinst -n
node1,node2,node3 -r 10gR1
```

Verbose Mode and "Unknown" Output

If you run CVU using the `-verbose` argument, and a CVU command responds with `UNKNOWN` for a particular node, then this is because the CVU cannot determine whether a check passed or failed. The following is a list of possible causes for an "Unknown" response:

- The node is down
- Executables required by CVU are missing in the CRS_home `/bin` or Oracle home directory
- The user account starting CVU does not have privileges to run common operating system executables on the node
- The node is missing an operating system patch, or a required package
- The node has exceeded the maximum number of processes or maximum number of open files, or there is a problem with IPC segments, such as shared memory or semaphores

Logging In to the System as root

Before you install the Oracle software, you must complete several tasks as the `root` user. To log in as the `root` user, complete one of the following procedures:

- If you are installing the software from an X Window System workstation or X terminal, then:

  1. Start a local terminal session, for example, an X terminal (`xterm`).
  2. If you are not installing the software on the local system, then enter the following command to enable remote hosts to display X applications on the local X server:

     ```
     $ xhost + hostname
     ```

     The *hostname* is the name of the local host.
  3. If you are not installing the software on the local system, then use the `ssh`, `rlogin`, or `telnet` command to connect to the system where you want to install the software:

     ```
     $ telnet remote_host
     ```

  4. If you are not logged in as the `root` user, then enter the following command to switch the user to `root`:

     ```
     $ su - root
     password:
     #
     ```

- If you are installing the software from a PC or other system with X server software installed, then:

  Note: If necessary, refer to your X server documentation for more information about completing this procedure. Depending on the X server software that you are using, you may need to complete the tasks in a different order.

  1. Start the X server software.

2. Configure the security settings of the X server software to permit remote hosts to display X applications on the local system.
3. Connect to the remote system where you want to install the software and start a terminal session on that system, for example, an X terminal (`xterm`).
4. If you are not logged in as the `root` user on the remote system, then enter the following command to switch user to `root`:

```
$ su - root
password:
#
```

Creating Required Operating System Groups and User

Depending on whether this is the first time Oracle software is being installed on this system and on the products that you are installing, you may need to create several operating system groups and an operating system user account.

The following operating system groups and user are required if you are installing Oracle Database:

- The OSDBA group (typically, `dba`)

  You must create this group the first time you install Oracle Database software on the system. This group identifies operating system user accounts that have database administrative privileges (the SYSDBA privilege). The default name for this group is `dba`.

  If you want to specify a group name other than the default `dba` group, then you must choose the Custom installation type to install the software or start Oracle Universal Installer as a user that is not a member of this group. In this case, Oracle Universal Installer prompts you to specify the name of this group.

- The OSOPER group (typically, `oper`)

  This is an optional group. Create this group if you want a separate group of operating system users to have a limited set of database administrative privileges (the SYSOPER privilege). By default, members of the OSDBA group also have the SYSOPER privilege.

  If you want to specify a separate OSOPER group, other than the default `dba` group, then you must choose the Custom installation type to install the software or start Oracle Universal Installer as a user that is not a member of the `dba` group. In this case, Oracle Universal Installer prompts you to specify the name of this group. The usual name chosen for this group is `oper`.

- An unprivileged user

  Verify that the unprivileged user `nobody` exists on the system. The `nobody` user must own the external jobs (extjob) executable after the installation.

The following operating system group and user are required for all installations:

- The Oracle Inventory group (typically, `oinstall`)

  You must create this group the first time you install Oracle software on the system. The usual name chosen for this group is `oinstall`. This group owns the Oracle Inventory, which is a catalog of all Oracle software installed on the system.

> **Note:** If Oracle software is already installed on the system, then the existing Oracle Inventory group must be the primary group of the operating system user that you use to install new Oracle software. The following sections describe how to identify an existing Oracle Inventory group.

- The Oracle software owner user (typically, `oracle`)

  You must create this user the first time you install Oracle software on the system. This user owns all of the software installed during the installation. The usual name chosen for this user is `oracle`. This user must have the Oracle Inventory group as its primary group. It must also have the OSDBA and OSOPER groups as secondary groups. In Oracle documentation, when this user account is referred to, it is called the `oracle` user.

A single Oracle Inventory group is required for all installations of Oracle software on the system. After the first installation of Oracle software, you must use the same Oracle Inventory group for all subsequent Oracle software installations on that system. However, you can choose to create different Oracle software owner users, OSDBA groups, and OSOPER groups (other than `oracle`, `dba`, and `oper`) for separate installations. By using different groups for different installations, members of these different groups have DBA privileges only on the associated databases rather than on all databases on the system.

> **See Also:** *Oracle Database Administrator's Reference for UNIX Systems* and *Oracle Database Administrator's Guide* for more information about the OSDBA and OSOPER groups and the SYSDBA and SYSOPER privileges

The following sections describe how to create the required operating system user and groups:.

- Creating the Oracle Inventory Group
- Creating the OSDBA Group
- Creating an OSOPER Group (Optional)
- Creating the Oracle Software Owner User
- Verifying That the User nobody Exists
- Creating Identical Users and Groups on Other Cluster Nodes
- Configuring SSH on All Cluster Nodes

> **Note:** The following sections describe how to create local users and groups. As an alternative to creating local users and groups, you can create the appropriate users and groups in a directory service, such as Network Information Services (NIS). For information about using directory services, contact your system administrator or refer to your operating system documentation.

Creating the Oracle Inventory Group

Oracle Universal Installer (OUI) helps you to choose a group to use as the Oracle Inventory group. If you have an existing Oracle Inventory group, then provide this group name and path when prompted.

The following subsections describe how to determine the Oracle Inventory group name, if it exists, and how to create it if necessary.

Determining If the Oracle Inventory Exists

When you install Oracle software on the system for the first time, Oracle Universal Installer creates the `oraInst.loc` file. This file identifies the name of the Oracle Inventory group (typically, `oinstall`), and the path of the Oracle Inventory directory.

If you have an existing Oracle Inventory, then ensure that you use the same Oracle Inventory for all Oracle software installations.

If you do not have an existing Oracle Inventory, then you should create an Oracle Inventory group.

To determine whether you have an Oracle Inventory on your system, enter the following command:

```
# more /etc/oraInst.loc
```

If the `oraInst.loc` file exists, then the output from this command is similar to the following:

```
inventory_loc=/u01/app/oracle/oraInventory
inst_group=oinstall
```

In the previous output example:

- The `inventory_loc` group shows the location of the Oracle Inventory
- The `inst_group` parameter shows the name of the Oracle Inventory group (in this example, `oinstall`).

Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist

If the `oraInst.loc` file does not exist, then create the Oracle Inventory group by entering a command similar to the following:

```
# /usr/sbin/groupadd oinstall
```

Creating the OSDBA Group

You must create an OSDBA group in the following circumstances:

- An OSDBA group does not exist, for example, if this is the first installation of Oracle Database software on the system
- An OSDBA group exists, but you want to give a different group of operating system users database administrative privileges for a new Oracle Database installation

If the OSDBA group does not exist or if you require a new OSDBA group, then create it as follows. In the following procedure, use the group name `dba` unless a group with that name already exists:

```
# /usr/sbin/groupadd dba
```

Creating an OSOPER Group (Optional)

Create an OSOPER group only if you want to identify a group of operating system users with a limited set of database administrative privileges (SYSOPER operator privileges). For most installations, it is sufficient to create only the OSDBA group. If you want to use an OSOPER group, then you must create it in the following circumstances:

- If an OSOPER group does not exist; for example, if this is the first installation of Oracle Database software on the system
- If an OSOPER group exists, but you want to give a different group of operating system users database operator privileges in a new Oracle installation

If you require a new OSOPER group, then create it as follows. In the following, use the group name `oper` unless a group with that name already exists.

```
# /usr/sbin/groupadd oper
```

Creating the Oracle Software Owner User

You must create an Oracle software owner user in the following circumstances:

- If an Oracle software owner user does not exist; for example, if this is the first installation of Oracle software on the system
- If an Oracle software owner user exists, but you want to use a different operating system user, with different group membership, to give database administrative privileges to those groups in a new Oracle Database installation

> **Note:** If you intend to use multiple Oracle software owners for different Oracle homes, then you should create a separate Oracle software owner for Oracle Clusterware, and install Oracle Clusterware using the Oracle Clusterware software owner.

Determining if an Oracle Software Owner User Exists

To determine whether an Oracle software owner user named `oracle` exists, enter the following command:

```
# id oracle
```

If the `oracle` user exists, then the output from this command is similar to the following:

```
uid=440(oracle) gid=200(oinstall) groups=201(dba),202(oper)
```

If the user exists, then determine whether you want to use the existing user, or create another `oracle` user. If you want to use the existing user, then ensure that the user's primary group is the Oracle Inventory group and that it is a member of the appropriate OSDBA and OSOPER groups. Refer to one of the following sections for more information:

> **Note:** If necessary, contact your system administrator before using or modifying an existing user.

- To modify an existing user, refer to the "Modifying an Existing Oracle Software Owner User" section on page 2-9.
- To create a user, refer to the following section.

Creating an Oracle Software Owner User

If the Oracle software owner user does not exist, or if you require a new Oracle software owner user, then create it as follows. In the following procedure, use the user name `oracle` unless a user with that name already exists.

1. To create the `oracle` user, enter a command similar to the following:

   ```
   # /usr/sbin/useradd -u 200 -g oinstall -G dba[,oper] oracle
   ```

   In the preceding command:

   - The -u option specifies the user ID. Using this command flag is optional, as you can allow the system to provide you with an automatically generated user ID number. However, you must make note of the oracle user ID number, as you require it later during pre-installation.
   - The `-g` option specifies the primary group, which must be the Oracle Inventory group--for example, `oinstall`
   - The `-G` option specifies the secondary groups, which must include the OSDBA group, and, if required, the OSOPER group. For example: `dba`, or `dba,oper`

2. Set the password of the `oracle` user:

   ```
   # passwd oracle
   ```

   Refer to the section "Verifying That the User nobody Exists" on page 2-9.

Modifying an Existing Oracle Software Owner User

If the `oracle` user exists, but its primary group is not `oinstall`, or it is not a member of the appropriate OSDBA or OSOPER groups, then enter a command similar to the following to modify it. Specify the primary group using the `-g` option and any required secondary group using the `-G` option:

```
# /usr/sbin/usermod -g oinstall -G dba[,oper] oracle
```

Repeat this procedure on all of the other nodes in the cluster.

Verifying That the User nobody Exists

Before installing the software, complete the following procedure to verify that the user nobody exists on the system:

1. To determine if the user exists, enter the following command:

   ```
   # id nobody
   ```

   If this command displays information about the `nobody` user, then you do not have to create that user.

2. If the `nobody` user does not exist, then enter the following command to create it:

   ```
   # /usr/sbin/useradd nobody
   ```

3. Repeat this procedure on all the other nodes in the cluster.

Creating Identical Users and Groups on Other Cluster Nodes

> **Note:** You must complete the following procedures only if you are using local users and groups. If you are using users and groups defined in a directory service such as NIS, then they are already identical on each cluster node.

The Oracle software owner user and the Oracle Inventory, OSDBA, and OSOPER groups must exist and be identical on all cluster nodes. To create these identical users and groups, you must identify the user ID and group IDs assigned them on the node where you created them, then create the user and groups with the same name and ID on the other cluster nodes.

Identifying the User and Group IDs

To determine the user ID (UID) of the Oracle software owner user and the group IDs (GID) of the Oracle Inventory, OSDBA, and OSOPER groups, follow these steps:

1. Enter following command:

   ```
   # id oracle
   ```

   The output from this command is similar to the following:

   ```
   uid=440(oracle) gid=200(oinstall) groups=201(dba),202(oper)
   ```

2. From the output, identify the user ID (UID) for the `oracle` user and the group identities (GIDs) for the groups to which it belongs.

Creating the User and Groups on the Other Cluster Nodes

To create the user and groups on the other cluster nodes, repeat the following procedure on each node:

1. Log in to the next cluster node as `root`.
2. Enter commands similar to the following to create the `oinstall` and `dba` groups, and if required, the `oper` group. Use the `-g` option to specify the correct GID for each group.

   ```
   # /usr/sbin/groupadd -g 200 oinstall
   # /usr/sbin/groupadd -g 201 dba
   # /usr/sbin/groupadd -g 202 oper
   ```

   > **Note:** If the group already exists, then use the `groupmod` command to modify it if necessary. If you cannot use the same group ID for a particular group on this node, then view the `/etc/group` file on all nodes to identify a group ID that *is* available on every node. You must then specify that ID for the group on all of the nodes.

3. To create the `oracle` user, enter a command similar to the following:

   ```
   # /usr/sbin/useradd -u 200 -g oinstall -G dba[,oper] oracle
   ```

   In the preceding command:

- The `-u` option specifies the user ID, which must be the user ID that you identified in the previous subsection
- The `-g` option specifies the primary group, which must be the Oracle Inventory group, for example `oinstall`
- The `-G` option specifies the secondary groups, which must include the OSDBA group and if required, the OSOPER group. For example: `dba` or `dba,oper`

Note: If the `oracle` user already exists, then use the `usermod` command to modify it if necessary. If you cannot use the same user ID for the `oracle` user on this node, then view the `/etc/passwd` file on all nodes to identify a user ID that *is* available on every node. You must then specify that ID for the user on all of the nodes.

---

4. Set the password of the `oracle` user:

```
# passwd oracle
```

Configuring SSH on All Cluster Nodes

Before you install and use Oracle Real Application clusters, you must configure secure shell (SSH) for the `oracle` user on all cluster nodes. Oracle Universal Installer uses the `ssh` and `scp` commands during installation to run remote commands on and copy files to the other cluster nodes. You must configure SSH so that these commands do not prompt for a password.

Note: This section describes how to configure OpenSSH version 3. If SSH is not available, then Oracle Universal Installer attempts to use `rsh` and `rcp` instead. However, these services are disabled by default on most Linux systems.

To determine if SSH is running, enter the following command:

```
$ pgrep sshd
```

If SSH is running, then the response to this command is process ID numbers. To find out more about SSH, enter the following command:

```
$ man ssh
```

Configuring SSH on Cluster Member Nodes

To configure SSH, you must first create RSA and DSA keys on each cluster node, and then copy the keys from all cluster node members into an authorized keys file on each node. To do this task, complete the following steps:

Create RSA and DSA keys on each node: Complete the following steps on each node:

1. Log in as the `oracle` user.
2. If necessary, create the `.ssh` directory in the `oracle` user's home directory and set the correct permissions on it:

```
$ mkdir ~/.ssh
```

```
$ chmod 700 ~/.ssh
```

3. Enter the following commands to generate an RSA key for version 2 of the SSH protocol:

   ```
   $ /usr/bin/ssh-keygen -t rsa
   ```

   At the prompts:

   - Accept the default location for the key file.
   - Enter and confirm a pass phrase that is different from the `oracle` user's password.

   This command writes the public key to the `~/.ssh/id_rsa.pub` file and the private key to the `~/.ssh/id_rsa` file. *Never distribute the private key to anyone.*

4. Enter the following commands to generate a DSA key for version 2 of the SSH protocol:

   ```
   $ /usr/bin/ssh-keygen -t dsa
   ```

   At the prompts:

   - Accept the default location for the key file
   - Enter and confirm a pass phrase that is different from the `oracle` user's password

   This command writes the public key to the `~/.ssh/id_dsa.pub` file and the private key to the `~/.ssh/id_dsa` file. Never distribute the private key to anyone.

Add keys to an authorized key file: Complete the following steps:

1. On the local node, determine if you have an authorized key file (`~/.ssh/authorized_keys`). If the authorized key file already exists, then proceed to step 2. Otherwise, enter the following commands:

   ```
   $ touch ~/.ssh/authorized_keys
   $ cd ~/.ssh
   $ ls
   ```

   You should see the `id_dsa.pub` and `id_rsa.pub` keys that you have created.

2. Using SSH, copy the *contents* of the `~/.ssh/id_rsa.pub` and `~/.ssh/id_dsa.pub` files to the file `~/.ssh/authorized_keys`, and provide the Oracle user password as prompted. This process is illustrated in the following syntax example with a two-node cluster, with nodes node1 and node2, where the Oracle user path is `/home/oracle:`

   ```
   [oracle@node1 .ssh]$ ssh node1 cat /home/oracle/.ssh/id_rsa.pub >>
   authorized_keys
   oracle@node1's password:
   [oracle@node1 .ssh]$ ssh node1 cat /home/oracle/.ssh/id_dsa.pub >>
   authorized_keys
   [oracle@node1 .ssh$ ssh node2 cat /home/oracle/.ssh/id_rsa.pub >>
   authorized_keys
   oracle@node2's password:
   [oracle@node1 .ssh$ ssh node2 cat /home/oracle/.ssh/id_dsa.pub
   >>authorized_keys
   oracle@node2's password:
   ```

Note: Repeat this process for each node in the cluster

3. Use SCP (Secure Copy) or SFTP (Secure FTP) to copy the `authorized_keys` file to the Oracle user .ssh directory on a remote node. The following example is with SCP, on a node called node2, where the Oracle user path is `/home/oracle`:

   ```
   [oracle@node1 .ssh]scp authorized_keys node2:/home/oracle/.ssh/
   ```

4. Repeat step 2 and 3 for each cluster node member. When you have added keys from each cluster node member to the authorized_keys file on the last node you want to have as a cluster node member, then use SCP to copy the complete authorized_keys file back to each cluster node member

   Note: The Oracle user's `/.ssh/authorized_keys` file on every node must contain the contents from all of the `/.ssh/id_rsa.pub` and `/.ssh/id_dsa.pub` files that you generated on all cluster nodes.

5. Change the permissions on the Oracle user's `/.ssh/authorized_keys` file on all cluster nodes:

   ```
   $ chmod 600 ~/.ssh/authorized_keys
   ```

   At this point, if you use `ssh` to log in to or run a command on another node, you are prompted for the pass phrase that you specified when you created the DSA key.

Enabling SSH User Equivalency on Cluster Member Nodes

To enable Oracle Universal Installer to use the `ssh` and `scp` commands without being prompted for a pass phrase, follow these steps:

1. On the system where you want to run Oracle Universal Installer, log in as the `oracle` user.
2. Enter the following commands:

   ```
   $ exec /usr/bin/ssh-agent $SHELL
   $ /usr/bin/ssh-add
   ```

3. At the prompts, enter the pass phrase for each key that you generated.

   If you have configured SSH correctly, then you can now use the `ssh` or `scp` commands without being prompted for a password or a pass phrase.

4. If you are on a remote terminal, and the local node has only one visual (which is typical), then use the following syntax to set the DISPLAY environment variable:

   Bourne, Korn, and Bash shells

   ```
   $ export DISPLAY=hostname:0
   ```

   C shell:

   ```
   $ setenv DISPLAY 0
   ```

   For example, if you are using the Bash shell, and if your hostname is node1, then enter the following command:

```
$ export DISPLAY=node1:0
```

5. To test the SSH configuration, enter the following commands from the same terminal session, testing the configuration of each cluster node, where *nodename1*, *nodename2*, and so on, are the names of nodes in the cluster:

```
$ ssh nodename1 date
$ ssh nodename2 date
    .
    .
    .
```

These commands should display the date set on each node.

If any node prompts for a password or pass phrase, then verify that the `~/.ssh/authorized_keys` file on that node contains the correct public keys.

If you are using a remote client to connect to the local node, and you see a message similar to "Warning: No xauth data; using fake authentication data for X11 forwarding," then this means that your authorized keys file is configured correctly, but your ssh configuration has X11 forwarding enabled. To correct this, proceed to step 6.

Note: The first time you use SSH to connect to a node from a particular system, you may see a message similar to the following:

```
The authenticity of host 'node1 (140.87.152.153)' can't be
established.
RSA key fingerprint is
7z:ez:e7:f6:f4:f2:4f:8f:9z:79:85:62:20:90:92:z9.
Are you sure you want to continue connecting (yes/no)?
```

Enter `yes` at the prompt to continue. You should not see this message again when you connect from this system to that node.

If you see any other messages or text, apart from the date, then the installation can fail. Make any changes required to ensure that only the date is displayed when you enter these commands.

You should ensure that any parts of login scripts that generate any output, or ask any questions, are modified so that they act only when the shell is an interactive shell

---

6. To ensure that X11 forwarding will not cause the installation to fail, create a user-level SSH client configuration file for the Oracle software owner user, as follows:

   a. Using any text editor, edit or create the `~oracle/.ssh/config` file.

   b. Make sure that the ForwardX11 attribute is set to `no`. For example:

```
Host *
      ForwardX11 no
```

7. You must run Oracle Universal Installer from this session or remember to repeat steps 2 and 3 before you start Oracle Universal Installer from a different terminal session.

Preventing Oracle Clusterware Installation Errors Caused by stty Commands

During an Oracle Clusterware installation, Oracle Universal Installer uses SSH (if available) to run commands and copy files to the other nodes. During the installation,

hidden files on the system (for example, .bashrc or .cshrc) will cause installation errors if they contain stty commands.

To avoid this problem, you must modify these files to suppress all output on STDERR, as in the following examples:

- Bourne, Bash, or Korn shell:

```
if [ -t 0 ]; then
   stty intr ^C
fi
```

- C shell:

```
test -t 0
if ($status == 0) then
   stty intr ^C
endif
```

Note: When SSH is not available, the Installer uses the `rsh` and `rcp` commands instead of `ssh` and `scp`.

If there are hidden files that contain stty commands that are loaded by the remote shell, then OUI indicates an error and stops the installation.

Configuring the oracle User's Environment

You run Oracle Universal Installer from the `oracle` account. However, before you start Oracle Universal Installer you must configure the environment of the `oracle` user.

To configure the environment, you must:

- Set the default file mode creation mask (umask) to 022 in the shell startup file
- Set the DISPLAY, ORACLE_BASE, and ORACLE_HOME environment variables in preparation for the Oracle Clusterware installation

You should also ensure that the PATH variable contains `$ORACLE_HOME/bin` before `/usr/X11R6/bin`

To set the `oracle` user's environment, follow these steps:

1. Start a new terminal session; for example, start an X terminal (`xterm`).
2. Enter the following command to ensure that X Window applications can display on this system:

   ```
   $ xhost + hostname
   ```

   The *hostname* is the name of the local host.
3. If you are not already logged in to the system where you want to install the software, then log in to that system as the `oracle` user.
4. If you are not logged in as the `oracle` user, then switch user to `oracle`:

   ```
   $ su - oracle
   ```

5. To determine the default shell for the `oracle` user, enter the following command:

   ```
   $ echo $SHELL
   ```

6. Open the `oracle` user's shell startup file in any text editor:

   Note: On Red Hat Linux, `.bash_profile` is the user startup file for the Bash shell.

   - Bourne shell (`sh`), Bash shell (`bash`) or Korn shell (`ksh`):

     ```
     % vi .bash_profile
     ```

   - C shell (`csh` or `tcsh`):

     ```
     % vi .login
     ```

7. Enter or edit the following line, specifying a value of 022 for the default file mode creation mask:

   ```
   umask 022
   ```

8. If the `ORACLE_SID`, `ORACLE_HOME`, or `ORACLE_BASE` environment variable is set in the file, then remove the appropriate lines from the file.
9. Save the file, and exit from the text editor.
10. To run the shell startup script, enter one of the following commands:

    - Bash shell on Red Hat Enterprise Linux:

      ```
      $ . ./.bash_profile
      ```

    - Bourne, Bash, or Korn shell:

      ```
      $ . ./.profile
      ```

    - C shell:

      ```
      % source ./.login
      ```

11. If you are not installing the software on the local system, then enter a command similar to the following to direct X applications to display on the local system:

    - Bourne, Bash, or Korn shell:

      ```
      $ DISPLAY=local_host:0.0 ; export DISPLAY
      ```

    - C shell:

      ```
      % setenv DISPLAY local_host:0.0
      ```

    In this example, *`local_host`* is the host name or IP address of the system that you want to use to display Oracle Universal Installer (your workstation or PC).

12. If you determined that the `/tmp` directory has less than 400 MB of free disk space, then identify a file system with at least 400 MB of free space and set the `TEMP` and `TMPDIR` environment variables to specify a temporary directory on this file system:

    Note: You cannot use a shared file system as the location of the `/tmp` file. `/tmp` for RAC installation. If you place /tmp on a shared file system, then the installation fails.

a. Use the `df -h` command to identify a suitable file system with sufficient free space.

b. If necessary, enter commands similar to the following to create a temporary directory on the file system that you identified, and set the appropriate permissions on the directory:

```
$ su - root
# mkdir /mount_point/tmp
# chmod 775 /mount_point/tmp
# exit
```

c. Enter commands similar to the following to set the TEMP and TMPDIR environment variables:

* Bourne, Bash, or Korn shell:

```
$ TEMP=/mount_point/tmp
$ TMPDIR=/mount_point/tmp
$ export TEMP TMPDIR
```

* C shell:

```
% setenv TEMP /mount_point/tmp
% setenv TMPDIR /mount_point/tmp
```

Installing the cvuqdisk Package for Red Hat Linux

If you are using Red Hat Linux, then you must download and install the Red Hat operating system package `cvuqdisk`. Without `cvuqdisk`, CVU is unable to discover shared disks, and you receive the error message "Package cvuqdisk not installed" when you run CVU.

To install the cvuqdisk RPM, complete the following procedure:

Note: If you prefer, you can choose to disable CVU shared disk checks by adding the following line to the file *CRS_home*`/cv/admin/cvuconfig`:

```
CV_RAW_CHECK_ENABLED=FALSE
```

---

1. Copy the cvuqdisk package to each node on the cluster. You should ensure that each node is running the same version of Linux.
2. Log in as root.
3. Using the following command, check to see if you have an existing version of the cvuqdisk package:

   ```
   # rpm -qi cvuqdisk
   ```

   If you have an existing version, then enter the following command:

   ```
   rpm -e cvuqdisk
   ```

4. Set the environment variable CVUDISK_GRP to point to the group that will own cvuqdisk, typically `oinstall`.
5. Use a command similar to the following to install the `cvuqdisk` package:

```
rpm -iv cvuqdisk-1.0.0-1.i386.rpm
```

Note that the package you install depends on your operating system.

Checking the Hardware Requirements

Each system must meet the following minimum hardware requirements:

- At least 1 GB of physical RAM
- Swap space equivalent to the multiple of the available RAM, as indicated in the following table:

Available RAM	Swap Space Required
Between 1 GB and 2 GB	1.5 times the size of RAM
More than 2 GB	Equal to the size of RAM

- 400 MB of disk space in the `/tmp` directory
- Up to 4 GB of disk space for the Oracle software, depending on the installation type and platform
- 1.2 GB of disk space for a preconfigured database that uses file system storage (optional)

> **Note:** The disk space requirement for databases that use Automatic Storage Management or raw device storage is described later in this chapter.

Additional disk space, either on a file system or in an Automatic Storage Management disk group, is required for the flash recovery area if you choose to configure automated backups.

To ensure that each system meets these requirements, follow these steps:

1. To determine the physical RAM size, enter the following command:

   ```
   # grep MemTotal /proc/meminfo
   ```

   If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.

2. To determine the size of the configured swap space, enter the following command:

   ```
   # grep SwapTotal /proc/meminfo
   ```

   If necessary, refer to your operating system documentation for information about how to configure additional swap space.

3. To determine the amount of disk space available in the `/tmp` directory, enter the following command:

   ```
   # df -k /tmp
   ```

   If there is less than 400 MB of disk space available in the `/tmp` directory, then complete one of the following steps:

   - Delete unnecessary files from the `/tmp` directory to make available the disk space required.

- Set the TEMP and TMPDIR environment variables when setting the `oracle` user's environment (described later).
- Extend the file system that contains the `/tmp` directory. If necessary, contact your system administrator for information about extending file systems.

4. To determine the amount of free disk space on the system, enter the following command:

```
# df -k
```

The following table shows the approximate disk space requirements for software files for each installation type:

Installation Type	Requirement for Software Files (GB)
Enterprise Edition	1.4
Standard Edition	1.4
Custom (maximum)	1.4

5. To determine whether the system architecture can run the software, enter the following command:

```
# grep "model name" /proc/cpuinfo
```

Note: This command displays the processor type. Verify that the processor architecture meets or exceeds the minimum requirement for the Oracle software release that you want to install. If it does not meet or exceed this requirement, then you cannot install the software on this system.

Checking the Network Requirements

Check that you have the networking hardware and internet protocol (IP) addresses required for an Oracle Real Application Clusters installation.

Note: For the most up-to-date information about supported network protocols and hardware for RAC installations, refer to the Certify pages on the Oracle*MetaLink* Web site at

`http://metalink.oracle.com`

Network Hardware Requirements

Each node in the cluster must meet the following requirements:

- Each node must have at least two network adapters: one for the public network interface, and one for the private network interface (the interconnect).
- The public interface names associated with the network adapters for each network must be the same on all nodes, and the private interface names associated with the network adaptors should be the same on all nodes.

  For example: With a two-node cluster, you cannot configure network adapters on node1 with `eth0` as the public interface, but on node2 have `eth1` as the public interface. Public interface names must be the same, so you must configure eth0 as

public on *both* nodes. You should configure the private interfaces on the same network adapters as well. If `eth1` is the private interface for node 1, then `eth1` should be the private interface for node 2.

- For increased reliability, configure redundant public and private network adapters for each node.
- For the public network, each network adapter must support TCP/IP.
- For the private network, the interconnect must support the user datagram protocol (UDP) using high-speed network adapters and switches that support TCP/IP (Gigabit Ethernet or better recommended).

  Note: UDP is the default interconnect protocol for RAC, and TCP is the interconnect protocol for Oracle Clusterware.

  Token-Ring is not supported for the interconnect.

- For the private network, the endpoints of all designated interconnect interfaces must be completely reachable on the network. There should be no node that is not connected to every private network. You can test whether an interconnect interface is reachable using a ping command.

Network Parameter Requirements

If you are using NFS for your shared storage, then you must set the values for the NFS buffer size parameters `rsize` and `wsize` to at least 16384. Oracle recommends that you use the value 32768.

For example, if you decide to use rsize and wsize buffer settings with the value 16384, then update the `/etc/fstab` file on each node with an entry similar to the following:

```
clusternode:/vol/DATA/oradata  /home/oracle/netapp     nfs
rw,hard,nointr,rsize=16384,wsize=16384,tcp,noac,vers=3  1 2
```

IP Address Requirements

Before starting the installation, you must identify or obtain the following IP addresses for each node:

- An IP address and an associated host name registered in the domain name service (DNS) for each public network interface
- One unused virtual IP address (VIP) and an associated virtual host name registered in DNS, or resolved in the hostfile, or both, that you will configure for the primary public network interface

  The VIP address must be in the same subnet as the associated public interface. After installation, you can configure clients to use the virtual host name or IP address. If a node fails, then its VIP address fails over to another node.

- A private IP address and optional host name for each private interface

  Oracle recommends that you use non-routable IP addresses for the private interfaces; for example: 10.*.*.* or 192.168.*.*. You can use the `/etc/hosts` file on each node to associate private host names with private IP addresses.

In the example described in the following table, the node rac1 has two public interfaces and one private interface. This is a proper configuration of host names and IP addresses. Each node of the cluster on which node rac1 is a member must have similar host names and IP addresses:

Host Name	Type	IP Address	Registered In
rac1.mydomain.com	Public	143.47.43.100	DNS (optional)
rac1-2.mydomain.com	Public	143.46.51.101	DNS (optional)
rac1-vip.mydomain.com	Virtual	143.47.43.104	DNS (optional)
rac1-priv1	Private	10.0.0.1	/etc/hosts

Node Time Requirements

Before starting the installation, ensure that each member node of the cluster is set as closely as possible to the same date and time. Oracle strongly recommends using the Network Time Protocol feature of most operating systems for this purpose, with all nodes using the same reference Network Time Protocol server.

Configuring the Network Requirements

To verify that each node meets the requirements, follow these steps:

1. If necessary, install the network adapters for the public and private networks and configure them with either public or private IP addresses.
2. Register the host names and IP addresses for the public network interfaces in DNS.
3. For each node, register one virtual host name and IP address in DNS.
4. For each private interface on every node, add a line similar to the following to the `/etc/hosts` file on all nodes, specifying the private IP address and associated private host name:

   ```
   10.0.0.1     rac1-priv1
   ```

5. To identify the interface name and associated IP address for every network adapter, enter the following command:

   ```
   # /sbin/ifconfig
   ```

   From the output, identify the interface name and IP address for all network adapters that you want to specify as public or private network interfaces.

   Note: When you install Oracle Clusterware and RAC, you will require this information.

6. To prevent network hangs with failovers from public to virtual IP addresses with RAC databases using NAS devices or NFS mounts, enter the following command as root to enable the Name Service Cache Daemon (`nscd`):

   ```
   # /sbin/service  nscd start
   ```

Checking the Network Setup with CVU

As the Oracle user, enter a command using the following syntax to verify node connectivity among all of the nodes for which your cluster is configured:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh comp nodecon -n node_list [-verbose]
```

In the preceding syntax example, the variable *node_list* is a comma-separated list of nodes in your cluster. This command detects all the network interfaces available on the cluster nodes, and verifies the connectivity among all the nodes through the network interfaces it finds.

Select the option `-verbose` to receive progress updates as the CVU performs its system checks, and detailed reporting of the test results.

For example, to verify node connectivity on a two-node cluster with nodes node1 and node2, with the mountpoint /dev/dvdrom, and with updates and a summary of the verification checks the CVU performs, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh comp nodecon -n node1,node2 -verbose
```

Note: You can use this command to obtain a list of all the interfaces available on the nodes that are suitable for use as VIPs, as well as a list of private interconnects that are connecting successfully on all nodes.

Checking the Software Requirements

Depending on the products that you intend to install, verify that the following software is installed on the system. The procedure following the table describes how to check these requirements.

Note: Oracle Universal Installer performs checks on your system to verify that it meets the listed requirements. To ensure that these checks pass, verify the requirements before you start Oracle Universal Installer.

Item	Requirement
Operating system	■ Red Hat Enterprise Linux AS/ES 3 (Update 3 or later) ■ SUSE Linux Enterprise Server 9
Kernel version	The system must be running the following kernel version (or a later version): **Red Hat Enterprise Linux 3 (Update 2):** 2.4.21-15.EL **SUSE Linux Enterprise Server 9:** 2.6.5-7.97

Item	Requirement

Packages	The following packages (or later versions) must be installed:
	Red Hat Enterprise Linux 3 (Update 2):
	`binutils-2.14`
	`compat-db-4.0.14-5`
	`compat-gcc-7.3-2.96.128`
	`compat-gcc-c++-7.3-2.96.128`
	`compat-libstdc++-7.3-2.96.128`
	`compat-libstdc++-devel-7.3-2.96.128`
	`gcc-3.2`
	`glibc-2.3.2-95.27`
	`make-3.79`
	`openmotif-2.2.3`
	`setarch-1.3-1`
	SUSE Linux Enterprise Server 9:
	`gcc-3.3`
	`gcc-c++-3.3.3-43`
	`glibc-2.3.3-98.28`
	`libaio-0.3.98-18`
	`libaio-devel-0.3.98-18`
	`make-3.80`
	`openmotif-libs-2.2.2-519.1`

PL/SQL native compilation, Pro*C/C++, Oracle Call Interface, Oracle C++ Call Interface, Oracle XML Developer's Kit (XDK)	The version of GNU C and C++ compilers listed previously for your distribution is supported for use with these products.
Oracle JDBC/OCI Drivers	On x86-64 systems, you can use the following optional JDK version with the Oracle JDBC/OCI drivers; however, it is not required for the installation: ■ JDK 1.4.2_03 with the JNDI extension. http://www.sun.com/java

Oracle Real Application Clusters	Oracle Cluster File System (OCFS)
	■ Version 1.0.13-1 or later for SUSE Linux Enterprise Server 8 on Itanium systems
	■ Version 1.0.11-1 or later for other installations
	OCFS requires the following kernel packages:
	`ocfs-support`
	`ocfs-tools`
	`ocfs-`*`kernel_version`*
	In the preceding list, the variable *`kernel_version`* represents the kernel version of the operating system on which you are installing OCFS.
	Note: OCFS is required only if you want to use a cluster file system for database file storage. If you want to use Automatic Storage Management or raw devices for database file storage, then you do not need to install OCFS.
	Obtain OCFS kernel packages, installation instructions, and additional information about OCFS from the following URL:
	`http://oss.oracle.com/projects/ocfs/`

To ensure that the system meets these requirements, follow these steps:

1. To determine which distribution and version of Linux is installed, enter the following command:

   ```
   # cat /proc/version
   ```

   Note: Only the distributions and versions listed in the previous table are supported.

2. To determine whether the required kernel errata is installed on Red Hat Enterprise Linux 3 systems, enter the following command:

   ```
   # uname -r
   2.4.21-15.EL
   ```

   In this example, the output shows the kernel version (`2.4.21`) and errata level (`15.EL`) on the system. If the errata level is less than 15, then install the latest kernel update. Kernel updates are available from the Red Hat Network.

3. To determine whether the required kernel is installed on SUSE Linux Enterprise Server 9 systems, enter the following command:

   ```
   # uname -r
   2.6.5-7.97
   ```

   In this example, the output shows the kernel version (`2.6.5-7.97`) on the system. If the kernel version is less than 2.6.5-7.97, then contact SUSE for information about obtaining and installing the kernel updates.

4. To determine whether the required packages are installed, enter commands similar to the following:

   ```
   # rpm -q package_name
   ```

   If a package is not installed, then install it from your Linux distribution media or download the required package version from your Linux vendor's Web site.

5. To determine whether OCFS is installed, enter the following command:

   ```
   # rpm -qa | grep ocfs
   ```

   If you want to install the Oracle Database files on an OCFS file system and the packages are not installed, then download them from the following Web site. Follow the instructions listed with the kit to install the packages and configure the file system:

   http://oss.oracle.com/projects/ocfs/

Configuring Kernel Parameters

> **Note:** The kernel parameter and shell limit values shown in the following section are recommended values only. For production database systems, Oracle recommends that you tune these values to optimize the performance of the system. Refer to your operating system documentation for more information about tuning kernel parameters.

Verify that the kernel parameters shown in the following table are set to values greater than or equal to the recommended value shown. The procedure following the table describes how to verify and set the values.

On all cluster nodes, verify that the kernel parameters shown in the following table are set to values greater than or equal to the recommended value shown. The procedure following the table describes how to verify and set the values.

Parameter	Value	File
semmsl semmns semopm semmni	250 32000 100 128	`/proc/sys/kernel/sem`
shmmax	Half the size of physical memory (in bytes)	`/proc/sys/kernel/shmmax`
shmmni	4096	`/proc/sys/kernel/shmmni`
shmall	2097152	`/proc/sys/kernel/shmall`
file-max	65536	`/proc/sys/fs/file-max`
ip_local_port_range	Minimum: 1024 Maximum: 65000	`/proc/sys/net/ipv4/ip_local_port_range`
rmem_default	262144	`/proc/sys/net/core/rmem_default`
rmem_max	262144	`/proc/sys/net/core/rmem_max`
wmem_default	262144	`/proc/sys/net/core/wmem_default`
wmem_max	262144	`/proc/sys/net/core/wmem_max`

> **Note:** If the current value for any parameter is later than the value listed in this table, then do not change the value of that parameter.

To view the current value specified for these kernel parameters, and to change them if necessary, follow these steps:

1. Enter the commands shown in the following table to view the current values of the kernel parameters:

> **Note:** Make a note of the current values and identify any values that you must change.

Parameter	Command
semmsl, semmns, semopm, and semmni	`# /sbin/sysctl -a | grep sem` This command displays the value of the semaphore parameters in the order listed.
shmall, shmmax, and shmmni	`# /sbin/sysctl -a | grep shm`
file-max	`# /sbin/sysctl -a | grep file-max`
ip_local_port_range	`# /sbin/sysctl -a | grep ip_local_port_range` This command displays a range of port numbers.
rmem_default, rmem_max, wmem_default, and wmem_max	`# /sbin/sysctl -a | grep net.core`

2. If the value of any kernel parameter is different from the recommended value, then complete the following process:

   Using any text editor, create or edit the `/etc/sysctl.conf` file, and add or edit lines similar to the following:

   > **Note:** Include lines only for the kernel parameter values that you want to change. For the semaphore parameters (`kernel.sem`), you must specify all four values. However, if any of the current system parameter values are greater than the recommended values, then keep using the larger values.

```
kernel.shmall = 2097152
kernel.shmmax = 2147483648
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
fs.file-max = 65536
net.ipv4.ip_local_port_range = 1024 65000
net.core.rmem_default = 262144
net.core.rmem_max = 262144
net.core.wmem_default = 262144
net.core.wmem_max = 262144
```

   By specifying the values in the `/etc/sysctl.conf` file, they persist when you reboot the system.

   On Red Hat systems, to have these changes take effect immediately so that you do not have to restart the system, enter the following command:

```
/sbin/sysctl -p
```

3. Repeat steps 1 and 2 on all other nodes in the cluster.

   On SUSE systems only, enter the following command to cause the system to read the `/etc/sysctl.conf` file when it reboots:

```
# /sbin/chkconfig boot.sysctl on
```

Setting Shell Limits for the oracle User

To improve the performance of the software on Linux systems, you must increase the following shell limits for the `oracle` user:

Shell Limit	Item in limits.conf	Hard Limit
Maximum number of open file descriptors	`nofile`	65536
Maximum number of processes available to a single user	`nproc`	16384

To increase the shell limits:

1. Add the following lines to the `/etc/security/limits.conf` file:

```
oracle              soft    nproc   2047
oracle               hard    nproc   16384
oracle               soft    nofile  1024
oracle               hard    nofile  65536
```

2. Add or edit the following line in the `/etc/pam.d/login` file, if it does not already exist:

```
session    required     /lib/security/pam_limits.so
```

3. Depending on the `oracle` user's default shell, make the following changes to the default shell startup file:

   - For the Bourne, Bash, or Korn shell, add the following lines to the `/etc/profile` file (or the file on SUSE systems)`/etc/profile.local`:

```
if [ $USER = "oracle" ]; then
        if [ $SHELL = "/bin/ksh" ]; then
              ulimit -p 16384
              ulimit -n 65536
        else
              ulimit -u 16384 -n 65536
        fi
fi
```

   - For the C shell (`csh` or `tcsh`), add the following lines to the `/etc/csh.login` file (or the file on SUSE systems)`/etc/csh.login.local`:

```
if ( $USER == "oracle" ) then
        limit maxproc 16384
        limit descriptors 65536
endif
```

4. Repeat this procedure on all other nodes in the cluster.

Refer to the "Identifying Required Software Directories" section to continue.

Checking the Hardware and Operating System Setup with CVU

As the `oracle` user, use the following command syntax to start Cluster Verification Utility (CVU) stage verification to check hardware and operating system setup:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh stage -post hwos -n node_list
[-verbose]
```

In the preceding syntax example, replace the variable *node_list* with the names of the nodes in your cluster, separated by commas. For example, to check the hardware and operating system of a two-node cluster with nodes node1 and node2, with the mountpoint `/dev/dvdrom/` and with the option to limit the output to the test results, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh stage -post hwos -n node1,node2
```

Select the option `-verbose` to receive detailed reports of the test results, and progress updates about the system checks performed by Cluster Verification Utility.

Checking the Operating System Requirements Setup with CVU

As the `oracle` user, use the following command syntax to check if your system meets the operating system requirement pre-installation tasks:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh comp sys -n node_list -p
{crs|database} -osdba osdba_group -orainv orainv_group -verbose
```

In the preceding syntax example:

- The variable mountpoint is the mountpoint of the Oracle 10*g* Release 2 (10.2) installation media
- The variable *node_list* is the list of nodes in your cluster, separated by commas
- The `-p` flag identifies either `crs` or `database`, and indicates that checks are performed for Oracle Clusterware or Oracle Database system requirements
- The variable *osdba_group* is the name of your OSDBA group, typically `dba`
- The variable *orainv_group* is the name of your Oracle Inventory group, typically `oinstall`

You can select the option `-verbose` to receive progress updates as the CVU performs its system checks, and detailed reporting of the test results.

For example, to perform a system check for an Oracle Clusterware installation on a two-node cluster with nodes node1 and node2, with the OSDBA `dba` and Oracle inventory group `oinstall`, and with the media mountpoint `/dev/dvdrom/`, then enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh comp sys -n node1,node2 -p crs -osdba
crs -orainv oinstall
```

Identifying Required Software Directories

You must identify or create the following directories for the Oracle software, as follows:

- Oracle Base Directory
- Oracle Inventory Directory
- Oracle Clusterware Home Directory
- Oracle Home Directory

The following subsections describe the requirements for these directories.

Oracle Base Directory

The Oracle base directory acts as a top-level directory for Oracle software installations. On Linux systems, the Optimal Flexible Architecture (OFA) guidelines recommend that you use a path similar to the following for the Oracle base directory:

/mount_point/app/*oracle_sw_owner*

In this example:

- *mount_point* is the mount point directory for the file system that will contain the Oracle software.

  The examples in this guide use `/u01` for the mount point directory. However, you could choose another mount point directory, `/oracle` or `/opt/oracle` for example.

- *oracle_sw_owner* is the operating system user name of the Oracle software owner, for example `oracle`.

You can use the same Oracle base directory for more than one installation or you can create separate Oracle base directories for different installations. If different operating system users install Oracle software on the same system, then each user must create a separate Oracle base directory. The following example Oracle base directories could all exist on the same system:

```
/u01/app/oracle
/u01/app/orauser
/opt/oracle/app/oracle
```

The following sections describe how to identify existing Oracle base directories that might be suitable for your installation and how to create an Oracle base directory if necessary.

Regardless of whether you create an Oracle base directory or decide to use an existing one, you must set the ORACLE_BASE environment variable to specify the full path to the Oracle base directory.

Note: The Oracle base directory can be on a local file system or on an NFS file system on a certified NAS device. Do not create the Oracle base directory on an OCFS version 1 file system.

Oracle Inventory Directory

The Oracle Inventory directory (`oraInventory`) stores an inventory of all software installed on the system. It is required by, and shared by, all Oracle software installations on a single system. The first time you install Oracle software on a system, Oracle Universal Installer prompts you to specify the path to this directory. If you are installing the software on a local file system, then Oracle recommends that you choose the following path:

oracle_base/oraInventory

If the Oracle base directory is on a cluster file system, or on an NFS file system on an NAS device, then you must specify a path for the Oracle Inventory directory on a local file system. The Oracle base directory must be on a local file system to enable all of the nodes to have separate inventories.

Oracle Universal Installer creates the directory that you specify, and sets the correct owner, group, and permissions for it. You do not need to create it.

Note: All Oracle software installations rely on the Oracle base directory. Make sure that you back it up regularly.

Do not delete the Oracle base directory unless you have completely removed all Oracle software from the system.

Oracle Clusterware Home Directory

The Oracle Clusterware home directory is the directory where you choose to install the software for Oracle Clusterware. You must install Oracle Clusterware in a separate home directory. When you run Oracle Universal Installer, it prompts you to specify the path to this directory, as well as a name that identifies it. Oracle recommends that you specify a path similar to the following for the Oracle Clusterware home directory:

```
/u01/app/oracle/product/crs
```

Note: Because you must change the permissions of all of the parent directories of the Oracle Clusterware home directory after installing the software to grant write access only to the `root` user, the Oracle Clusterware home directory must not be a subdirectory of the Oracle base directory.

Oracle Home Directory

The Oracle home directory is the directory where you choose to install the software for a particular Oracle product. You must install different Oracle products, or different releases of the same Oracle product, in separate Oracle home directories. When you run Oracle Universal Installer, it prompts you to specify the path to this directory, as well as a name that identifies it. The directory that you specify must be a subdirectory of the Oracle base directory. Oracle recommends that you specify a path similar to the following for the Oracle home directory:

```
oracle_base/product/10.2.0/db_1
```

Oracle Universal Installer creates the directory path that you specify under the Oracle base directory. It also sets the correct owner, group, and permissions on it. You do not need to create this directory.

Identifying or Creating an Oracle Base Directory

Before starting the installation, you must either identify an existing Oracle base directory or if required, create one. This section contains information about the following:

- Identifying an Existing Oracle Base Directory
- Creating an Oracle Base Directory

Note: You can choose to create an Oracle base directory, even if other Oracle base directories exist on the system.

Identifying an Existing Oracle Base Directory

Existing Oracle base directories might not have paths that comply with OFA guidelines. However, if you identify an existing Oracle Inventory directory or existing

Oracle home directories, you can usually identify the Oracle base directories, as follows:

- Identifying an existing Oracle Inventory directory

  Enter the following command on all nodes in the cluster to view the contents of the `oraInst.loc` file:

  ```
  # more /etc/oraInst.loc
  ```

  If the `oraInst.loc` file exists, then the output from this command is similar to the following:

  ```
  inventory_loc=/u01/app/oracle/oraInventory
  inst_group=oinstall
  ```

  The `inventory_loc` parameter identifies the Oracle Inventory directory (`oraInventory`) on that system. The parent directory of the `oraInventory` directory is typically an Oracle base directory. In the previous example, `/u01/app/oracle` is an Oracle base directory.

- Identifying existing Oracle home directories

  Enter the following command on all nodes in the cluster to view the contents of the `oratab` file:

  ```
  # more /etc/oratab
  ```

  If the `oratab` file exists, then it contains lines similar to the following:

  ```
  *:/u03/app/oracle/product/10.2.0/db_1:N
  *:/opt/orauser/infra_904:N
  *:/oracle/9.2.0:N
  ```

  The directory paths specified on each line identify Oracle home directories. Directory paths that end with the user name of the Oracle software owner that you want to use are valid choices for an Oracle base directory. If you intend to use the `oracle` user to install the software, then you could choose one of the following directories from the previous example:

  ```
  /u03/app/oracle
  /oracle
  ```

  Note: If possible, choose a directory path similar to the first (`/u03/app/oracle`). This path complies with the OFA guidelines.

Before deciding to use an existing Oracle base directory for this installation, make sure that it satisfies the following conditions:

- It should not be on the same file system as the operating system.
- It must have an identical path on all nodes in the cluster, or it must be an NFS file system on a certified NAS device.

  If you are not using an NFS file system, then create identical Oracle base directories on the other nodes.

- It must have 1.4 GB free disk space on all the nodes in the cluster

  To determine the free disk space on the file system where the Oracle base directory is located, enter the following command:

  ```
  # df -h oracle_base_path
  ```

When you are configuring the `oracle` user's environment in the section "Creating the Oracle Clusterware Home Directory" on page 2-32, set the ORACLE_BASE environment variable to specify the directory you choose.

If an Oracle base directory does not exist on the system, or if you want to create an Oracle base directory, then refer to the following section.

Creating an Oracle Base Directory

Before you create an Oracle base directory, you must identify an appropriate file system. The Oracle base directory requires 1.5 GB of free disk space.

To identify an appropriate file system, follow these steps:

1. Use the `df -h` command to determine the free disk space on each mounted file system.
2. From the display, identify a file system that has appropriate free space.

   Note: The Oracle base directory can be on a local file system or on an NFS file system on a certified NAS device. Do not create the Oracle base directory on an OCFS version 1 file system.

   The path to the Oracle base directory must be the same on all nodes.
3. Note the name of the mount point directory for the file system that you identified.

To create the Oracle base directory and specify the correct owner, group, and permissions for it, follow these steps:

1. Enter commands similar to the following to create the recommended subdirectories in the mount point directory that you identified, and to set the appropriate owner, group, and permissions on them:

   ```
   # mkdir -p /mount_point/app/oracle_sw_owner
   # chown -R oracle:oinstall /mount_point/app/oracle_sw_owner
   # chmod -R 775 /mount_point/app/oracle_sw_owner
   ```

   For example, if the mount point you identify is `/u01`, and `oracle` is the user name of the Oracle software owner, then the recommended Oracle base directory path is as follows:

   ```
   /u01/app/oracle
   ```
2. If necessary, repeat the commands listed in the previous step to create the same directory on the other nodes in the cluster.
3. When you configure the `oracle` user's environment later in this chapter, set the ORACLE_BASE environment variable to specify the Oracle base directory you have created in this task.

Creating the Oracle Clusterware Home Directory

Oracle Universal Installer (OUI) creates the Oracle Clusterware home directory for you. Ensure before you start the installation that you provide sufficient disk space on a file system for the Oracle Clusterware directory, and the parent directory of the Oracle Clusterware directory space is writable by the Oracle user.

To identify an appropriate file system, follow these steps:

1. Use the `df -h` command to determine the free disk space on each mounted file system.
2. From the display, identify a file system that has at least 120 MB of free disk space.

   If you are using the same file system for the Oracle base directory, then this 120 MB of disk space is additional to the free disk space requirement that you identified previously.

   Note: The file system can be a local file system or an NFS file system on a certified NAS device. Do not create the Oracle Clusterware home directory on an OCFS file system.

   The path to the Oracle Clusterware home directory must be the same on all nodes.
3. Note the name of the mount point directory for the file system that you identified.

To create the Oracle Clusterware home directory and specify the correct owner, group, and permissions for it, follow these steps:

1. Enter commands similar to the following to create the recommended subdirectories in the mount point directory that you identified and set the appropriate owner, group, and permissions on them:

   ```
   # mkdir -p /mount_point/crs/oracle_sw_owner/product/10/app
   # chown -R root:oinstall /mount_point/crs
   # chmod -R 775 /mount_point/crs/oracle_sw_owner
   ```

   If the mount point you identified is `/u01`, then the recommended Oracle Clusterware home directory path is as follows:

   ```
   /u01/crs/oracle/product/10/crs
   ```

   Note: After installation, you should change permissions so that only the root user can write to the Oracle Clusterware home directory.
2. If necessary, repeat the commands listed in the previous step to create the same directory on the other nodes in the cluster.
3. Enter commands similar to the following to set the ORACLE_BASE and ORACLE_HOME environment variable in preparation for the Oracle Clusterware installation:
   - Bourne, Bash, or Korn shell:

     ```
     $ ORACLE_BASE=/u01/app/oracle
     $ ORACLE_HOME=/u01/crs/oracle/product/10/app
     $ export ORACLE_BASE
     $ export ORACLE_HOME
     ```

   - C shell:

     ```
     % setenv ORACLE_BASE /u01/app/oracle
     % setenv ORACLE_HOME /u01/crs/oracle/product/10/app
     ```
4. Enter the following commands to ensure that the TNS_ADMIN environment variable is not set:
   - Bourne, Bash, or Korn shell:

```
$ unset TNS_ADMIN
```

- C shell:

```
% unsetenv TNS_ADMIN
```

5. To verify that the environment has been set correctly, enter the following commands:

```
$ umask
$ env | more
```

Verify that the `umask` command displays a value of `22`, `022`, or `0022` and the environment variables that you set in this section have the correct values.

Stopping Existing Oracle Processes for Upgrades or Co-existing Databases

Table 2–1 provides an overview of what you need to do if you have an existing Oracle database on the system where you plan to install Oracle Database 10*g* Release 2 (10.2). Review the table, and perform tasks as required.

> **See Also:** *Oracle Database Upgrade Guide* for additional information about preparing for and performing upgrades.

Table 2–1 Overview of System Preparation for Upgrades or Co-existing Databases

Installation Scenario	What you need to do
Upgrading from Oracle Database 10g Release 1 (10.1) to 10*g* Release 2 (10.2)	No additional tasks. Refer to Installing Oracle 10g Release 2 on a System with Oracle 10g Release 1 on page 2-35
installing Oracle Database 10*g* Release 2 (10.2) on a system to co-exist with Oracle Database 10*g* Release 1 (10.1)	No additional tasks. Refer to Installing Oracle 10g Release 2 on a System with Oracle 10g Release 1 on page 2-35
Upgrading from Oracle9*i* Release 9.2 to Oracle Database 10*g* Release 2 (10.2)	Shut down the Global Service Daemon, and shut down a default listener on port 1521, if present. Refer to Installing Oracle 10g Release 2 on a System with Oracle9i Release 2 on page 2-35
Installing Oracle Database 10*g* Release 2 (10.2) on a system to co-exist with Oracle9*i* Release 9.2	Shut down a default listener on port 1521, if present, and shut down the Global Service Daemon. Refer to Installing Oracle 10g Release 2 on a System with Oracle9i Release 2 on page 2-35

Installing Oracle 10g Release 2 on a System with Oracle 10*g* Release 1

If your system has an Oracle Database Release 10*g* Release 10. 1 installation, and you install an Oracle Database 10*g* Release 2 (10.2) either to coexist with or to upgrade the 10.1, then most installation types configure and start a default Oracle Net listener using TCP/IP port 1521 and the IPC key value EXTPROC. One of the following occurs:

- During a co-existing installation, Database Configuration Assistant (DBCA) automatically migrates the listener and related files from the 10.1 Oracle home to the 10.2 Oracle home.

- During an upgrade, Oracle Database Upgrade Assistant (DBUA) automatically locates the Oracle 10*g* release 1 (10.1) listener, and migrates it to Oracle 10g release 2.

Proceed to Chapter 3.

Installing Oracle 10*g* Release 2 on a System with Oracle9*i* Release 2

This section provides instructions for preparing

Explanation of Tasks If you are installing an Oracle Database 10*g* Release 2 (10.2) on a system with an existing Oracle9*i* Release 2 (9.2) database, and the Oracle Net listener process is using the *same* port or key value as the default used with the Oracle 10*g* Release 2 (10.2) installation, port 1521, then Oracle Universal Installer can only configure the new listener; it cannot start it. To ensure that the new listener process starts during the installation, you must shut down any existing listeners before starting Oracle Universal Installer. To do this, refer to "Shutting Down the Listener" on page 2-35

You must shut down the Global Services Daemon (GSD), because otherwise, during 10*g* Release 2 (10.2) installation, the Oracle9i Release 9.2 SRVM shared data is upgraded into an Oracle Cluster Registry that the 9.2 GSD will not be able to use. The 10.2 Oracle Clusterware installation starts a 10*g* Release 2 (10.2) GSD to serve the Oracle9*i* 9.2 clients. To do this, refer to "Shutting down the Global Services Daemon" on page 2-36

Shutting Down the Listener To determine whether an existing Oracle9*i* listener process is running and to shut it down if necessary, follow these steps:

1. Switch user to `oracle`:

   ```
   # su - oracle
   ```

2. Enter the following command to determine whether an Oracle9*i* listener process is running and to identify its name and the Oracle home directory in which it is installed:

   ```
   $ ps -ef | grep tnslsnr
   ```

   This command displays information about the Oracle Net listeners running on the system:

   ```
   ... oracle_home1/bin/tnslsnr LISTENER -inherit
   ```

   In this example, *`oracle_home1`* is the Oracle home directory where the listener is installed and `LISTENER` is the listener name.

   > **Note:** If no Oracle Net listeners are running, then proceed to Chapter 3.

3. Set the ORACLE_HOME environment variable to specify the appropriate Oracle home directory for the listener:

   - Bourne, Bash, or Korn shell:

     ```
     $ ORACLE_HOME=oracle_home1
     $ export ORACLE_HOME
     ```

   - C or tcsh shell:

```
% setenv ORACLE_HOME oracle_home1
```

4. Enter the following command to identify the TCP/IP port number and IPC key value that the listener is using:

   ```
   $ $ORACLE_HOME/bin/lsnrctl status listenername
   ```

   Note: If the listener uses the default name LISTENER, then you do not have to specify the listener name in this command.

5. Enter a command similar to the following to stop the listener process:

   ```
   $ $ORACLE_HOME/bin/lsnrctl stop listenername
   ```

6. Repeat this procedure to stop all listeners running on this system and on all other nodes in the cluster.

Shutting down the Global Services Daemon As the Oracle user, on each node of the cluster, use the following syntax to shut down the GSD:

```
$ cd 92_Oracle_home
$ bin/gsdctl stop
```

In the preceding syntax example, the variable 92_Oracle_home is the Oracle9i Release 2 (9.2) database home.

3

Configuring Oracle Clusterware and Oracle Database Storage

This chapter describes the storage configuration tasks that you must complete before you start Oracle Universal Installer. It includes information about the following tasks:

- Reviewing Storage Options for Oracle Clusterware, Database, and Recovery Files
- Configuring Storage for Oracle Clusterware Files on a Supported Shared File System
- Configuring Storage for Oracle Clusterware Files on Raw Devices
- Choosing a Storage Option for Oracle Database Files
- Configuring Disks for Automatic Storage Management
- Configuring Database File Storage on Raw Devices

Reviewing Storage Options for Oracle Clusterware, Database, and Recovery Files

This section describes supported options for storing Oracle Clusterware files, Oracle Database files, and data files. It includes the following sections:

- Overview of Storage Options
- Checking for Available Shared Storage with CVU

Overview of Storage Options

Use the following overview to help you select your storage option:

Overview of Oracle Clusterware Storage Options

There are two ways of storing Oracle Clusterware files:

- **A supported shared file system**: Supported file systems include the following:
  - **Oracle Cluster File System (OCFS)**: The cluster file system Oracle provides for the Linux community
  - **Network File System (NFS)**: A file-level protocol that enables access and sharing of files
- **Raw partitions**: Raw partitions are disk partitions that are not mounted and written to using the Linux file system, but instead are accessed directly by the application.

Overview of Oracle Database and Recovery File Options

There are three ways of storing Oracle Database and recovery files:

- **Automatic Storage Management**: Automatic Storage Management (ASM) is an integrated, high-performance database file system and disk manager for Oracle files.
- **A supported shared file system**: Supported file systems include the following:
  - **Oracle Cluster File System (OCFS)**: Note that if you intend to use OCFS for your data files, then you should create partitions large enough for the database files when you create partitions for Oracle Clusterware. If you intend to store Oracle Clusterware files on OCFS, then you must ensure that OCFS volume sizes are at least 500 MB each.
  - **OSCP-Certified NAS Network File System (NFS)**: Note that if you intend to use NFS for your data files, then you should create partitions large enough for the database files when you create partitions for Oracle Clusterware
- **Raw partitions** (database files only): A raw partition is required for each database file.

See Also: For information about certified compatible storage optinos, refer to the Oracle Storage Compatibility Program (OSCP) Web site, which is at the following URL:

`http://www.oracle.com/technology/deploy/availability/htdocs/oscp.html`

General Storage Considerations

For all installations, you must choose the storage option that you want to use for Oracle Clusterware files and Oracle Database files. If you want to enable automated backups during the installation, then you must also choose the storage option that you want to use for recovery files (the flash recovery area). You do not have to use the same storage option for each file type.

For single-instance Oracle Database installations using Oracle Clusterware for failover, you must use OCFS, ASM, or shared raw disks if you do not want the failover processing to include dismounting and remounting disks.

The following table shows the storage options supported for storing Oracle Clusterware files, Oracle Database files, and Oracle Database recovery files. Oracle Database files include data files, control files, redo log files, the server parameter file, and the password file. Oracle Clusterware files include the Oracle Cluster Registry (OCR), a mirrored OCR file (optional), the Oracle Clusterware voting disk, and additional voting disk files (optional).

Note: For the most up-to-date information about supported storage options for RAC installations, refer to the Certify pages on the Oracle*MetaLink* Web site:

`http://metalink.oracle.com`

	File Types Supported			
Storage Option	**OCR and Voting Disk**	**Oracle Clusterware**	**Database**	**Recovery**
Automatic Storage Management	No	No	Yes	Yes
Cluster file system (OCFS)	Yes	No	Yes	Yes
Local storage	No	Yes	No	No
NFS file system **Note:** Requires a certified NAS device	Yes	Yes	Yes	Yes
Shared raw partitions	Yes	No	Yes	No

Use the following guidelines when choosing the storage options that you want to use for each file type:

- You can choose any combination of the supported storage options for each file type provided that you satisfy any requirements listed for the chosen storage options.
- Oracle recommends that you choose Automatic Storage Management (ASM) as the storage option for database and recovery files.
- For Standard Edition RAC installations, ASM is the only supported storage option for database or recovery files.
- You cannot use ASM to store Oracle Clusterware files, because these files must be accessible before any Oracle instance starts.
- If you intend to use ASM with RAC, and you are configuring a new ASM instance, then you must ensure that your system meets the following conditions:
  - All nodes on the cluster have the release 2 (10.2) version of Oracle Clusterware installed
  - Any existing ASM instance on any node in the cluster is shut down
- If you intend to upgrade an existing RAC database, or a RAC database with ASM instances, then you must ensure that your system meets the following conditions:
  - The RAC database or RAC database with ASM instance is running on the node from which the Oracle Universal Installer (OUI) and Database Configuration Assistant (DBCA) is run
  - The RAC database or RAC database with an ASM instance is running on the *same* nodes that you intend to make members of the new cluster installation. For example, if you have an existing RAC database running on a three-node cluster, then you must install the upgrade on all three nodes. You cannot upgrade only 2 nodes of the cluster, removing the third instance in the upgrade.

    See Also: *Oracle Database Upgrade Guide* for information about how to prepare for upgrading an existing database

- If you do not have a storage option that provides external file redundancy, then you must configure at least three voting disk areas to provide voting disk redundancy.

After You Have Selected Disk Storage Options

When you have determined your disk storage options, you must perform the following tasks in the following order:

1: Check for available shared storage with CVU

Refer to Checking for Available Shared Storage with CVU on page 3-4

2: Configure shared storage for Oracle Clusterware files

- **To use a file system (NFS or OCFS) for Oracle Clusterware files**, refer to Configuring Storage for Oracle Clusterware Files on a Supported Shared File System on page 3-5
- **To use raw devices (partitions) for Oracle Clusterware files**, refer to "Configuring Storage for Oracle Clusterware Files on Raw Devices" on page 3-8

3: Configure storage for Oracle Database files and recovery files

- **To use a file system for database or recovery file storage**, refer to Configuring Storage for Oracle Clusterware Files on a Supported Shared File System on page 3-5, and ensure that in addition to the volumes you create for Oracle Clusterware files, you also create additional volumes with sizes sufficient to store database files.
- **To use Automatic Storage Management for database or recovery file storage**, refer to "Configuring Database File Storage on ASM and Raw Devices" on page 3-20.
- **To use raw devices (partitions) for database file storage**, refer to "Configuring Database File Storage on Raw Devices" on page 3-21.

Checking for Available Shared Storage with CVU

To check for all shared file systems available across all nodes on the cluster, either with OCFS or with an NFS file system, use the following command:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh comp ssa -n node_list
```

If you want to check the shared accessibility of a specific shared storage type to specific nodes in your cluster, then use the following command syntax:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh comp ssa -n node_list -s storageID_
list
```

In the preceding syntax examples, the variable *mountpoint* is the mountpoint path of the installation media, the variable *node_list* is the list of nodes you want to check, separated by commas, and the variable *storageID_list* is the list of storage device IDs for the storage devices managed by the file system type that you want to check.

For example, if you want to check the shared accessibility from node1 and node2 of storage devices `/dev/sdb` and `/dev/sdc`, and your mountpoint is `/dev/dvdrom/`, then enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh comp ssa -n node1,node2 -s
/dev/sdb,/dev/sdc
```

If you do not specify specific storage device IDs in the command, then the command searches for all available storage devices connected to the nodes on the list

Configuring Storage for Oracle Clusterware Files on a Supported Shared File System

Oracle Universal Installer (OUI) does not suggest a default location for the Oracle Cluster Registry (OCR) or the Oracle Clusterware voting disk. If you choose to create these files on a file system, then review the following sections to complete storage requirements for Oracle Clusterware files:

- Requirements for Using a file system for Oracle Clusterware Files
- Checking NFS Buffer Size Parameters
- Creating Required Directories for Oracle Clusterware Files on Shared File Systems

Requirements for Using a file system for Oracle Clusterware Files

To use a file system for Oracle Clusterware files, the file system must comply with the following requirements

- To use a cluster file system on Linux, you must use OCFS
- To use an NFS file system, it must be on a certified NAS device

> **Note:** If you are using a shared file system on a NAS device to store a shared Oracle home directory for Oracle Clusterware or RAC, then you must use the same NAS device for Oracle Clusterware file storage.

- If you choose to place your Oracle Cluster Registry (OCR) files on a shared file system, then you should ensure one of the following is true:
  - Either ensure that the disks used for the file system are on a highly available storage device, (for example, a RAID device that implements file redundancy), or mount at least two file systems, and use the features of Oracle Database 10*g* Release 2 (10.2) to provide redundancy for the OCR.
- If you intend to use a shared file system to store database files, then you should ensure that you use at least two independent file systems, with the database files on one file system, and the recovery files on a different file system.
- The `oracle` user must have write permissions to create the files in the path that you specify.

> **Note:** If you are upgrading from Oracle9*i* release 2, then you can continue to use the raw device or shared file that you used for the SRVM configuration repository instead of creating a new file for the OCR.

Use Table 3–1 to determine the partition size for shared file systems:

Table 3–1 Shared file system Volume Size Requirements

File Types Stored	Number of Volumes	Volume Size
Oracle Clusterware files (OCR and voting disks) with external redundancy	1	At least 120 MB for each volume

Table 3–1 (Cont.) Shared file system Volume Size Requirements

File Types Stored	Number of Volumes	Volume Size
Oracle Clusterware files (OCR and voting disks) with redundancy provided by Oracle	1	At least 120 MB for each volume
Redundant Oracle Clusterware files with redundancy provided by Oracle (mirrored OCR and two additional voting disks)	1	At least 140 MB (100 MB for the mirrored OCR, and 20 MB each for the additional voting disks)
Oracle Database files	1	At least 1.2 GB for each volume
Recovery files **Note**: Recovery files must be on a different volume than database files	1	At least 2 GB for each volume

In Table 3–1, the total required volume size is cumulative. For example, to store all files on the shared file system, you should have at least 3.4 GB of storage available over a minimum of two volumes.

To determine whether OCFS is installed, enter the following command:

```
# rpm -qa | grep ocfs
```

If you want to install the Oracle Database files on an OCFS file system and the packages are not installed, then download them from the following Web site. Follow the instructions listed with the kit to install the packages and configure the file system:

http://oss.oracle.com/projects/ocfs/

Checking NFS Buffer Size Parameters

If you are using NFS, then you must set the values for the NFS buffer size parameters `rsize` and `wsize` to at least 16384. Oracle recommends that you use the value 32768.

For example, if you decide to use rsize and wsize buffer settings with the value 16384, then update the `/etc/fstab` file on each node with an entry similar to the following:

```
nfs_server:/vol/DATA/oradata  /home/oracle/netapp     nfs\
rw,hard,nointr,rsize=16384,wsize=16384,tcp,noac,vers=3  1 2
```

Creating Required Directories for Oracle Clusterware Files on Shared File Systems

Use the following instructions to create directories for Oracle Clusterware files. If you intend to use a file system to store Oracle Clusterware files, then you can also configure file systems for the Oracle Database and recovery files.

> **Note:** For both NFS and OCFS storage, you must complete this procedure only if you want to place the Oracle Clusterware files on a separate file system to the Oracle base directory.

To create directories for the Oracle Clusterware files on separate file systems to the Oracle base directory, follow these steps:

1. If necessary, configure the shared file systems that you want to use and mount them on each node.

Note: The mount point that you use for the file system must be identical on each node. Make sure that the file systems are configured to mount automatically when a node reboots.

2. Use the `df -h` command to determine the free disk space on each mounted file system.
3. From the display, identify the file systems that you want to use:

File Type	File System Requirements
Oracle Clusterware files	Choose a file system with at least 120 MB of free disk space
Database files	Choose either: ■ A single file system with at least 1.2 GB of free disk space ■ Two or more file systems with at least 1.2 GB of free disk space in total
Recovery files	Choose a file system with at least 2 GB of free disk space.

   If you are using the same file system for more than one type of file, then add the disk space requirements for each type to determine the total disk space requirement.
4. Note the names of the mount point directories for the file systems that you identified.
5. If the user performing installation (typically, `oracle`) has permissions to create directories on the disks where you plan to install Oracle Clusterware and Oracle Database, then OUI creates the Oracle Clusterware file directory, and DBCA creates the Oracle Database file directory, and the Recovery file directory.

   If the user performing the installation does not have write access, then you must create these directories manually using commands similar to the following to create the recommended subdirectories in each of the mount point directories and set the appropriate owner, group, and permissions on them:

   - Oracle Clusterware file directory:

     ```
     # mkdir /mount_point/oracrs
     # chown oracle:oinstall /mount_point/oracrs
     # chmod 775 /mount_point/oracrs
     ```

   - Database file directory:

     ```
     # mkdir /mount_point/oradata
     # chown oracle:oinstall /mount_point/oradata
     # chmod 775 /mount_point/oradata
     ```

   - Recovery file directory (flash recovery area):

     ```
     # mkdir /mount_point/flash_recovery_area
     # chown oracle:oinstall /mount_point/flash_recovery_area
     # chmod 775 /mount_point/flash_recovery_area
     ```

By making the Oracle user the owner of these directories, this permits them to be read by multiple Oracle homes, including those with different OSDBA groups.

When you have completed creating subdirectories in each of the mount point directories, and set the appropriate owner, group, and permissions, you have completed OCFS or NFS configuration.

Configuring Storage for Oracle Clusterware Files on Raw Devices

The following subsections describe how to configure Oracle Clusterware files on raw partitions.

- Clusterware File Restrictions for Logical Volume Manager on Linux
- Identifying Required Raw Partitions for Clusterware Files
- Creating the Required Raw Partitions on IDE, SCSI, or RAID Devices
- Binding the Partitions to Raw Devices

Clusterware File Restrictions for Logical Volume Manager on Linux

The procedures contained in this section describe how to create raw partitions for Oracle Clusterware. Although Red Hat Enterprise Linux 3 and SUSE Linux Enterprise Server provide a Logical Volume Manager (LVM), this LVM is not cluster-aware. For this reason, on x86 and Itanium systems, Oracle does not support the use of logical volumes with RAC for either Oracle Clusterware or database files.

Note: The use of logical volumes for raw devices is supported only for single-instance databases. Their use is not supported for RAC databases.

Identifying Required Raw Partitions for Clusterware Files

Table 3–2 lists the number and size of the raw partitions that you must configure for Oracle Clusterware files.

Table 3–2 Raw Partitions Required for Oracle Clusterware Files on Linux

Number	Size for Each Partition (MB)	Purpose
2 (or 1, if you have external redundancy support for this file)	100	Oracle Cluster Registry **Note:** You need to create these raw partitions only once on the cluster. If you create more than one database on the cluster, then they all share the same Oracle Cluster Registry (OCR). You should create two partitions: One for the OCR, and one for a mirrored OCR. If you are upgrading from Oracle9*i* release 2, then you can continue to use the raw device that you used for the SRVM configuration repository instead of creating this new raw device.
3 (or 1, if you have external redundancy support for this file)	20	Oracle Clusterware voting disks **Note:** You need to create these raw partitions only once on the cluster. If you create more than one database on the cluster, then they all share the same Oracle Clusterware voting disk. You should create three partitions: One for the voting disk, and two for additional voting disks.

Note: If you put Oracle Clusterware files on Oracle Cluster File System (OCFS) then you should ensure that OCFS volumes are at least 500 GB in size.

Creating the Required Raw Partitions on IDE, SCSI, or RAID Devices

If you intend to use IDE, SCSI, or RAID devices for the raw devices, then follow these steps:

1. If necessary, install or configure the shared disk devices that you intend to use for the raw partitions and reboot the system.

   Note: Because the number of partitions that you can create on a single device is limited, you might need to create the required raw partitions on more than one device.

2. To identify the device name for the disks that you want to use, enter the following command:

   ```
   # /sbin/fdisk -l
   ```

   Depending on the type of disk, the device name can vary:

Disk Type	Device Name Format	Description
IDE disk	/dev/hd*xn*	In this example, *x* is a letter that identifies the IDE disk and *n* is the partition number. For example, /dev/hda is the first disk on the first IDE bus.
SCSI disk	/dev/sd*xn*	In this example, *x* is a letter that identifies the SCSI disk and *n* is the partition number. For example, /dev/sda is the first disk on the first SCSI bus.
RAID disk	/dev/rd/c*x*d*y*p*z* /dev/ida/c*x*d*y*p*z*	Depending on the RAID controller, RAID devices can have different device names. In the examples shown, *x* is a number that identifies the controller, *y* is a number that identifies the disk, and *z* is a number that identifies the partition. For example, /dev/ida/c0d1 is the second logical drive on the first controller.

   You can create the required raw partitions either on new devices that you added or on previously partitioned devices that have unpartitioned free space. To identify devices that have unpartitioned free space, examine the start and end cylinder numbers of the existing partitions and determine whether the device contains unused cylinders.

3. To create raw partitions on a device, enter a command similar to the following:

   ```
   # /sbin/fdisk devicename
   ```

   Use the following guidelines when creating partitions:

   – Use the p command to list the partition table of the device.

   – Use the n command to create a partition.

- After you have created the required partitions on this device, use the `w` command to write the modified partition table to the device.
- Refer to the `fdisk` man page for more information about creating partitions.

Binding the Partitions to Raw Devices

After you have created the required partitions, you must bind the partitions to raw devices on every node. However, you must first determine what raw devices are already bound to other devices. The procedure that you must follow to complete this task varies, depending on the Linux distribution that you are using:

Note: If the nodes are configured differently, then the disk device names might be different on some nodes. In the following procedure, make sure that you specify the correct disk device names on each node.

- Red Hat:

1. To determine what raw devices are already bound to other devices, enter the following command on every node:

   ```
   # /usr/bin/raw -qa
   ```

   Raw devices have device names in the form `/dev/raw/raw`*n*, where *n* is a number that identifies the raw device.

   For each device that you want to use, identify a raw device name that is unused on all nodes.

2. Open the `/etc/sysconfig/rawdevices` file in any text editor and add a line similar to the following for each partition that you created:

   ```
   /dev/raw/raw1 /dev/sdb1
   ```

   Specify an unused raw device for each partition.

3. For the raw device that you created for the Oracle Cluster Registry (OCR), enter commands similar to the following to set the owner, group, and permissions on the device file:

   ```
   # chown root:oinstall /dev/raw/raw<n>
   # chmod 640 /dev/raw/raw<n>
   ```

   By making the `oinstall` group the owner of the OCR, this permits the OCR to be read by multiple Oracle homes, including those with different OSDBA groups.

4. To bind the partitions to the raw devices, enter the following command:

   ```
   # /sbin/service rawdevices restart
   ```

   The system automatically binds the devices listed in the `rawdevices` file when it reboots.

5. Repeat steps 2 to 4 on the other nodes in the cluster.

- SUSE:

1. To determine what raw devices are already bound to other devices, enter the following command on every node:

```
# /usr/sbin/raw -qa
```

Raw devices have device names in the form `/dev/raw/raw`*n*, where *n* is a number that identifies the raw device.

For each device that you want to use, identify a raw device name that is unused on all nodes.

2. Open the `/etc/raw` file in any text editor and add a line similar to the following to associate each partition with an unused raw device:

```
raw1:sdb1
```

3. For the raw device that you created for the Oracle Cluster Registry, enter commands similar to the following to set the owner, group, and permissions on the device file:

```
# chown root:oinstall /dev/raw/rawn
# chmod 660 /dev/raw/rawn
```

4. To bind the partitions to the raw devices, enter the following command:

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

5. To ensure that the raw devices are bound when the system reboots, enter the following command:

```
# /sbin/chkconfig raw on
```

6. Repeat steps 2 to 5 on the other nodes in the cluster.

When you have completed creating subdirectories in each of the mount point directories, and set the appropriate owner, group, and permissions, you have completed OCFS or NFS configuration.

Choosing a Storage Option for Oracle Database Files

Database files consist of the files that make up the database, and the recovery area files. There are four options for storing database files:

- Oracle Cluster File System (OCFS)
- Network File System (NFS)
- Automatic Storage Management (ASM)
- Raw partitions (Database files only--not for the recovery area)

During configuration of Oracle Clusterware, if you selected OCFS or NFS, and the volumes that you created are large enough to hold the database files and recovery files, then you have completed required pre-installation steps. You can proceed to Chapter 4, "Installing Oracle Clusterware" on page 4-1.

If you want to place your database files on ASM, then proceed to Configuring Disks for Automatic Storage Management on page 3-12.

If you want to place your database files on raw devices, and manually provide storage management for your database and recovery files, then proceed to "Configuring Database File Storage on Raw Devices" on page 3-21.

Note: Databases can consist of a mixture of ASM files and non-ASM files. Refer to *Oracle Database Administrator's Guide* for additional information about ASM.

Configuring Disks for Automatic Storage Management

This section describes how to configure disks for use with Automatic Storage Management. Before you configure the disks, you must determine the number of disks and the amount of free disk space that you require. The following sections describe how to identify the requirements and configure the disks on each platform:

- Identifying Storage Requirements for Automatic Storage Management
- Using an Existing Automatic Storage Management Disk Group
- Configuring Disks for Automatic Storage Management with ASMLIB
- Configuring Database File Storage on ASM and Raw Devices

Note: For Automatic Storage Management installations:

- Although this section refers to disks, you can also use zero-padded files on a certified NAS storage device in an Automatic Storage Management disk group. Refer to *Oracle Database Installation Guide* for Linux for information about creating and configuring NAS-based files for use in an Automatic Storage Management disk group.
- You can run ASM using ASMLIB, or run ASM using raw partitions.

Identifying Storage Requirements for Automatic Storage Management

To identify the storage requirements for using Automatic Storage Management, you must determine how many devices and the amount of free disk space that you require. To complete this task, follow these steps:

1. Determine whether you want to use Automatic Storage Management for Oracle Database files, recovery files, or both.

   Note: You do not have to use the same storage mechanism for database files and recovery files. You can use the file system for one file type and Automatic Storage Management for the other.

   If you choose to enable automated backups and you do not have a shared file system available, then you must choose Automatic Storage Management for recovery file storage.

   If you enable automated backups during the installation, you can choose Automatic Storage Management as the storage mechanism for recovery files by specifying an Automatic Storage Management disk group for the flash recovery area. Depending on how you choose to create a database during the installation, you have the following options:

   - If you select an installation method that runs Database Configuration Assistant in interactive mode (for example, by choosing the Advanced

database configuration option) then you can decide whether you want to use the same Automatic Storage Management disk group for database files and recovery files, or you can choose to use different disk groups for each file type.

The same choice is available to you if you use Database Configuration Assistant after the installation to create a database.

- If you select an installation method that runs Database Configuration Assistant in noninteractive mode, then you must use the same Automatic Storage Management disk group for database files and recovery files.

2. Choose the Automatic Storage Management redundancy level that you want to use for the Automatic Storage Management disk group.

   The redundancy level that you choose for the Automatic Storage Management disk group determines how Automatic Storage Management mirrors files in the disk group and determines the number of disks and amount of disk space that you require, as follows:

   - External redundancy

     An external redundancy disk group requires a minimum of one disk device. The effective disk space in an external redundancy disk group is the sum of the disk space in all of its devices.

     Because Automatic Storage Management does not mirror data in an external redundancy disk group, Oracle recommends that you use only RAID or similar devices that provide their own data protection mechanisms as disk devices in this type of disk group.

   - Normal redundancy

     In a normal redundancy disk group, Automatic Storage Management uses two-way mirroring by default, to increase performance and reliability. A normal redundancy disk group requires a minimum of two disk devices (or two failure groups). The effective disk space in a normal redundancy disk group is *half* the sum of the disk space in all of its devices.

     For most installations, Oracle recommends that you select normal redundancy disk groups.

   - High redundancy

     In a high redundancy disk group, Automatic Storage Management uses three-way mirroring to increase performance and provide the highest level of reliability. A high redundancy disk group requires a minimum of three disk devices (or three failure groups). The effective disk space in a high redundancy disk group is *one-third* the sum of the disk space in all of its devices.

     While high redundancy disk groups do provide a high level of data protection, you must consider the greater cost of additional storage devices before deciding to select high redundancy disk groups.

3. Determine the total amount of disk space that you require for the database files and recovery files.

   Use the following table to determine the minimum number of disks and the minimum disk space requirements for the installation:

Redundancy Level	Minimum Number of Disks	Database Files	Recovery Files	Both File Types
External	1	1.15 GB	2.3 GB	3.45 GB

Redundancy Level	Minimum Number of Disks	Database Files	Recovery Files	Both File Types
Normal	2	2.3 GB	4.6 GB	6.9 GB
High	3	3.45 GB	6.9 GB	10.35 GB

For RAC installations, you must also add additional disk space for the Automatic Storage Management metadata. You can use the following formula to calculate the additional disk space requirements (in MB):

15 + (2 * *number_of_disks*) + (126 * *number_of_Automatic_Storage_Management_instances*)

For example, for a four-node RAC installation, using three disks in a high redundancy disk group, you require an additional 525 MB of disk space:

15 + (2 * 3) + (126 * 4) = 525

If an Automatic Storage Management instance is already running on the system, then you can use an existing disk group to meet these storage requirements. If necessary, you can add disks to an existing disk group during the installation.

The following section describes how to identify existing disk groups and determine the free disk space that they contain.

4. Optionally, identify failure groups for the Automatic Storage Management disk group devices.

> **Note:** You need to complete this step only if you intend to use an installation method that runs Database Configuration Assistant in interactive mode, for example, if you intend to choose the Custom installation type or the Advanced database configuration option. Other installation types do not enable you to specify failure groups.

If you intend to use a normal or high redundancy disk group, then you can further protect your database against hardware failure by associating a set of disk devices in a custom failure group. By default, each device comprises its own failure group. However, if two disk devices in a normal redundancy disk group are attached to the same SCSI controller, then the disk group becomes unavailable if the controller fails. The controller in this example is a single point of failure.

To protect against failures of this type, you could use two SCSI controllers, each with two disks, and define a failure group for the disks attached to each controller. This configuration would enable the disk group to tolerate the failure of one SCSI controller.

> **Note:** If you define custom failure groups, then you must specify a minimum of two failure groups for normal redundancy disk groups and three failure groups for high redundancy disk groups.

5. If you are sure that a suitable disk group does not exist on the system, then install or identify appropriate disk devices to add to a new disk group. Use the following guidelines when identifying appropriate disk devices:

   - All of the devices in an Automatic Storage Management disk group should be the same size and have the same performance characteristics.

- Do not specify more than one partition on a single physical disk as a disk group device. Automatic Storage Management expects each disk group device to be on a separate physical disk.
- Although you can specify a logical volume as a device in an Automatic Storage Management disk group, Oracle does not recommend their use. Logical volume managers can hide the physical disk architecture, preventing Automatic Storage Management from optimizing I/O across the physical devices.

See Also: The "Configuring Disks for Automatic Storage Management" section on page 3-12 for information about completing this task

Using an Existing Automatic Storage Management Disk Group

If you want to store either database or recovery files in an existing Automatic Storage Management disk group, then you have the following choices, depending on the installation method that you select:

- If you select an installation method that runs Database Configuration Assistant in interactive mode (for example, by choosing the Advanced database configuration option), then you can decide whether you want to create a disk group, or use an existing one.

  The same choice is available to you if you use Database Configuration Assistant after the installation to create a database.

- If you select an installation method that runs Database Configuration Assistant in noninteractive mode, then you must choose an existing disk group for the new database; you cannot create a disk group. However, you can add disk devices to an existing disk group if it has insufficient free space for your requirements.

Note: The Automatic Storage Management instance that manages the existing disk group can be running in a different Oracle home directory.

To determine whether an existing Automatic Storage Management disk group exists, or to determine whether there is sufficient disk space in a disk group, you can use Oracle Enterprise Manager Grid Control or Database Control. Alternatively, you can use the following procedure:

1. View the contents of the `oratab` file to determine whether an Automatic Storage Management instance is configured on the system:

   ```
   # more /etc/oratab
   ```

   If an Automatic Storage Management instance is configured on the system, then the `oratab` file should contain a line similar to the following:

   ```
   +ASM2:oracle_home_path
   ```

   In this example, `+ASM2` is the system identifier (SID) of the Automatic Storage Management instance, with the node number appended, and *`oracle_home_path`* is the Oracle home directory where it is installed. By convention, theSIDfor an Automatic Storage Management instance begins with a plus sign.

2. Set the ORACLE_SID and ORACLE_HOME environment variables to specify the appropriate values for the Automatic Storage Management instance that you want to use.
3. Connect to the Automatic Storage Management instance as the SYS user with SYSDBA privilege and start the instance if necessary:

```
# $ORACLE_HOME/bin/sqlplus "SYS/SYS_password as SYSDBA"
SQL> STARTUP
```

4. Enter the following command to view the existing disk groups, their redundancy level, and the amount of free disk space in each one:

```
SQL> SELECT NAME,TYPE,TOTAL_MB,FREE_MB FROM V$ASM_DISKGROUP;
```

5. From the output, identify a disk group with the appropriate redundancy level and note the free space that it contains.
6. If necessary, install or identify the additional disk devices required to meet the storage requirements listed in the previous section.

Note: If you are adding devices to an existing disk group, then Oracle recommends that you use devices that have the same size and performance characteristics as the existing devices in that disk group.

Configuring Disks for Automatic Storage Management with ASMLIB

If you do not already have Automatic Storage Management disks, then Oracle provides an Automatic Storage Management library driver (ASMLIB).

ASMLIB simplifies the configuration and management of the disk devices that you want to use with Automatic Storage Management. A disk that is configured for use with Automatic Storage Management is known as a candidate disk.

If you intend to use Automatic Storage Management for database storage for Linux, then Oracle recommends that you install the ASMLIB driver and associated utilities, and use them to configure candidate disks.

Note: If you choose to configure disks using the Automatic Storage Management library driver, then you must use Database Configuration Assistant in interactive mode to create the database. You can run Database Configuration Assistant in interactive mode by choosing the Custom installation type or the Advanced database configuration option. You must also change the default disk discovery string to `ORCL:*`.

Note: If you do not use the Automatic Storage Management library driver, then you must bind each disk device that you want to use to a raw device, as described in Configuring Database File Storage on ASM and Raw Devices on page 3-20.

To use the Automatic Storage Management library driver (ASMLIB) to configure Automatic Storage Management devices, complete the following tasks.

- Installing and Configuring the Automatic Storage Management Library Driver Software
- Configuring the Disk Devices to Use the Automatic Storage Management Library Driver
- Administering the Automatic Storage Management Library Driver and Disks

Installing and Configuring the Automatic Storage Management Library Driver Software

To install and configure the ASMLIB driver software, follow these steps:

1. Enter the following command to determine the kernel version and architecture of the system:

   ```
   # uname -rm
   ```

2. If necessary, download the required ASMLIB packages from the OTN Web site:

   `http://www.oracle.com/technology/tech/linux/asmlib/index.html`

   Note: Automatic Storage Management library driver packages for some kernel versions are available in the Oracle Clusterware directory on the 10*g* Release 2 (10.2) DVD-ROM, in the `crs/RPMS/asmlib` directory. However, Oracle recommends that you check the OTN Web site for the most up-to-date packages.

   You must install the following packages, where *`version`* is the version of the Automatic Storage Management library driver, *`arch`* is the system architecture, and *`kernel`* is the version of the kernel that you are using:

   ```
   oracleasm-support-version.arch.rpm
   oracleasm-kernel-version.arch.rpm
   oracleasmlib-version.arch.rpm
   ```

3. Switch user to the `root` user:

   ```
   $ su -
   ```

4. Enter a command similar to the following to install the packages:

   ```
   # rpm -Uvh oracleasm-support-version.arch.rpm \
             oracleasm-kernel-version.arch.rpm \
             oracleasmlib-version.arch.rpm
   ```

   For example, if you are using the Red Hat Enterprise Linux AS 2.1 enterprise kernel on an x86 system, then enter a command similar to the following:

   ```
   # rpm -Uvh oracleasm-support-1.0.0-1.i386.rpm \
             oracleasm-2.4.9-e-enterprise-1.0.0-1.i686.rpm \
             oracleasmlib-1.0.0-1.i386.rpm
   ```

5. Enter the following command to run the `oracleasm` initialization script with the `configure` option:

   ```
   # /etc/init.d/oracleasm configure
   ```

6. Enter the following information in response to the prompts that the script displays:

Prompt	Suggested Response
Default user to own the driver interface:	Specify the Oracle software owner user (typically, `oracle`).
Default group to own the driver interface:	Specify the OSDBA group (typically `dba`).
Start Oracle Automatic Storage Management Library driver on boot (y/n):	Enter `y` to start the Oracle Automatic Storage Management library driver when the system starts.

The script completes the following tasks:

- Creates the `/etc/sysconfig/oracleasm` configuration file
- Creates the `/dev/oracleasm` mount point
- Loads the `oracleasm` kernel module
- Mounts the Automatic Storage Management library driver file system

Note: The Automatic Storage Management library driver file system is not a regular file system. It is used only by the Automatic Storage Management library to communicate with the Automatic Storage Management driver.

7. Repeat this procedure on all nodes in the cluster where you want to install Oracle Real Application Clusters.

Configuring the Disk Devices to Use the Automatic Storage Management Library Driver

To configure the disk devices that you want to use in an Automatic Storage Management disk group, follow these steps:

1. If you intend to use IDE, SCSI, or RAID devices in the Automatic Storage Management disk group, then follow these steps:
   a. If necessary, install or configure the shared disk devices that you intend to use for the disk group and reboot the system.
   b. To identify the device name for the disks that you want to use, enter the following command:

      ```
      # /sbin/fdisk -l
      ```

      Depending on the type of disk, the device name can vary:

Disk Type	Device Name Format	Description
IDE disk	`/dev/hd`*x*`n`	In this example, *x* is a letter that identifies the IDE disk and *n* is the partition number. For example, `/dev/hda` is the first disk on the first IDE bus.
SCSI disk	`/dev/sd`*x*`n`	In this example, *x* is a letter that identifies the SCSI disk and *n* is the partition number. For example, `/dev/sda` is the first disk on the first SCSI bus.

Disk Type	Device Name Format	Description
RAID disk	`/dev/rd/cxdypz` `/dev/ida/cxdypz`	Depending on the RAID controller, RAID devices can have different device names. In the examples shown, *x* is a number that identifies the controller, *y* is a number that identifies the disk, and *z* is a number that identifies the partition. For example, `/dev/ida/c0d1` is the second logical drive on the first controller.

To include devices in a disk group, you can specify either whole-drive device names or partition device names.

Note: Oracle recommends that you create a single whole-disk partition on each disk that you want to use.

c. Use either `fdisk` or `parted` to create a single whole-disk partition on the disk devices that you want to use.

2. Enter a command similar to the following to mark a disk as an Automatic Storage Management disk:

```
# /etc/init.d/oracleasm createdisk DISK1 /dev/sdb1
```

In this example, `DISK1` is a name that you want to assign to the disk.

Note: The disk names that you specify can contain uppercase letters, numbers, and the underscore character. They must start with an uppercase letter.

If you are using a multi-pathing disk driver with Automatic Storage Management, then make sure that you specify the correct logical device name for the disk.

3. To make the disk available on the other nodes in the cluster, enter the following command as `root` on each node:

```
# /etc/init.d/oracleasm scandisks
```

This command identifies shared disks attached to the node that are marked as Automatic Storage Management disks.

Administering the Automatic Storage Management Library Driver and Disks

To administer the Automatic Storage Management library driver and disks, use the `oracleasm` initialization script with different options, as follows:

Option	Description
`configure`	Use the `configure` option to reconfigure the Automatic Storage Management library driver, if necessary: `# /etc/init.d/oracleasm configure`

Option	Description
`enable` `disable`	Use the `disable` and `enable` options to change the behavior of the Automatic Storage Management library driver when the system boots. The `enable` option causes the Automatic Storage Management library driver to load when the system boots: `# /etc/init.d/oracleasm enable`

`start`
`stop`
`restart`
	Use the `start`, `stop`, and `restart` options to load or unload the Automatic Storage Management library driver without rebooting the system:
	`# /etc/init.d/oracleasm restart`

`createdisk`	Use the `createdisk` option to mark a disk device for use with the Automatic Storage Management library driver and give it a name: `# /etc/init.d/oracleasm createdisk` *`DISKNAME devicename`*
`deletedisk`	Use the `deletedisk` option to unmark a named disk device: `# /etc/init.d/oracleasm deletedisk` *`DISKNAME`* **Caution:** Do not use this command to unmark disks that are being used by an Automatic Storage Management disk group. You must drop the disk from the Automatic Storage Management disk group before you unmark it.
`querydisk`	Use the `querydisk` option to determine whether a disk device or disk name is being used by the Automatic Storage Management library driver: `# /etc/init.d/oracleasm querydisk {`*`DISKNAME`*` | `*`devicename`*`}`
`listdisks`	Use the `listdisks` option to list the disk names of marked Automatic Storage Management library driver disks: `# /etc/init.d/oracleasm listdisks`
`scandisks`	Use the `scandisks` option to enable cluster nodes to identify which shared disks have been marked as Automatic Storage Management library driver disks on another node: `# /etc/init.d/oracleasm scandisks`

When you have completed creating and configuring Automatic Storage Management, with ASMLIB, proceed to Chapter 4, "Installing Oracle Clusterware" on page 4-1

Configuring Database File Storage on ASM and Raw Devices

Note: For improved performance and easier administration, Oracle recommends that you use the Automatic Storage Management library driver (ASMLIB) instead of raw devices to configure Automatic Storage Management disks.

To configure disks for Automatic Storage Management (ASM) using raw devices, complete the following tasks:

1. To use ASM with raw partitions, you must create sufficient partitions for your data files, and then bind the partitions to raw devices. To do this, follow the instructions provided for Oracle Clusterware in the section "Configuring Storage for Oracle Clusterware Files on Raw Devices" on page 3-8.
2. Make a list of the raw device names you create for the data files, and have it available during database installation.

When you have completed creating and configuring ASM with raw partitions, proceed to Chapter 4, "Installing Oracle Clusterware" on page 4-1

Configuring Database File Storage on Raw Devices

The following subsections describe how to configure raw partitions for database files.

- Database File Restrictions for Logical Volume Manager on Linux
- Identifying Required Raw Partitions for Database Files
- Creating Required Raw Partitions for Database Files on IDE, SCSI, or RAID Devices
- Binding Partitions to Raw Devices for Database Files
- Creating the Database Configuration Assistant Raw Device Mapping File

Database File Restrictions for Logical Volume Manager on Linux

The procedures contained in this section describe how to create raw partitions for Oracle Clusterware and database file storage. Although Red Hat Enterprise Linux 3 and SUSE Linux Enterprise Server provide a Logical Volume Manager (LVM), this LVM is not cluster aware. For this reason, on x86 and Itanium systems, Oracle does not support the use of logical volumes with RAC for either Oracle Clusterware or database files.

Note: The use of logical volumes for raw devices is supported only for single-instance databases. Their use is not supported for RAC databases.

Identifying Required Raw Partitions for Database Files

Table 3–3 lists the number and size of the raw partitions that you must configure for database files.

Table 3–3 Raw Partitions Required for Database Files on Linux

Number	Partition Size (MB)	Purpose
1	500	SYSTEM tablespace
1	300 + (Number of instances * 250)	SYSAUX tablespace
Number of instances	500	UNDOTBS*n* tablespace (One tablespace for each instance)
1	250	TEMP tablespace
1	160	EXAMPLE tablespace
1	120	USERS tablespace
2 * number of instances	120	Two online redo log files for each instance
2	110	First and second control files
1	5	Server parameter file (SPFILE)
1	5	Password file

Note: If you prefer to use manual undo management, instead of automatic undo management, then, instead of the UNDOTBS*n* raw devices, you must create a single rollback segment tablespace (RBS) raw device that is at least 500 MB in size.

Creating Required Raw Partitions for Database Files on IDE, SCSI, or RAID Devices

If you intend to use IDE, SCSI, or RAID devices for the database raw devices, then follow these steps:

1. If necessary, install or configure the shared disk devices that you intend to use for the raw partitions and reboot the system.

   Note: Because the number of partitions that you can create on a single device is limited, you might need to create the required raw partitions on more than one device.

2. To identify the device name for the disks that you want to use, enter the following command:

   ```
   # /sbin/fdisk -l
   ```

   Depending on the type of disk, the device name can vary:

Disk Type	Device Name Format	Description
IDE disk	`/dev/hd`*`xn`*	In this example, *x* is a letter that identifies the IDE disk and *n* is the partition number. For example, `/dev/hda` is the first disk on the first IDE bus.
SCSI disk	`/dev/sd`*`xn`*	In this example, *x* is a letter that identifies the SCSI disk and *n* is the partition number. For example, `/dev/sda` is the first disk on the first SCSI bus.
RAID disk	`/dev/rd/c`*`x`*`d`*`y`*`p`*`z`* `/dev/ida/c`*`x`*`d`*`y`*`p`*`z`*	Depending on the RAID controller, RAID devices can have different device names. In the examples shown, *x* is a number that identifies the controller, *y* is a number that identifies the disk, and *z* is a number that identifies the partition. For example, `/dev/ida/c0d1` is the second logical drive on the first controller.

   You can create the required raw partitions either on new devices that you added or on previously partitioned devices that have unpartitioned free space. To identify devices that have unpartitioned free space, examine the start and end cylinder numbers of the existing partitions and determine whether the device contains unused cylinders.

3. To create raw partitions on a device, enter a command similar to the following:

   ```
   # /sbin/fdisk devicename
   ```

   Use the following guidelines when creating partitions:

   – Use the `p` command to list the partition table of the device.

   – Use the `n` command to create a partition.

- After you have created the required partitions on this device, use the `w` command to write the modified partition table to the device.
- Refer to the `fdisk` man page for more information about creating partitions.

Binding Partitions to Raw Devices for Database Files

After you have created the required partitions for database files, you must bind the partitions to raw devices on every node. However, you must first determine what raw devices are already bound to other devices. The procedure that you must follow to complete this task varies, depending on the Linux distribution that you are using:

Note: If the nodes are configured differently, then the disk device names might be different on some nodes. In the following procedure, make sure that you specify the correct disk device names on each node.

- Red Hat:

  1. To determine what raw devices are already bound to other devices, enter the following command on every node:

     ```
     # /usr/bin/raw -qa
     ```

     Raw devices have device names in the form `/dev/raw/raw`*n*, where *n* is a number that identifies the raw device.

     For each device that you want to use, identify a raw device name that is unused on all nodes.

  2. Open the `/etc/sysconfig/rawdevices` file in any text editor and add a line similar to the following for each partition that you created:

     ```
     /dev/raw/raw1 /dev/sdb1
     ```

     Specify an unused raw device for each partition.

  3. For each raw device that you specified in the `rawdevices` file, enter commands similar to the following to set the owner, group, and permissions on the device file:

     ```
     # chown oracle:dba /dev/raw/rawn
     # chmod 660 /dev/raw/rawn
     ```

  4. To bind the partitions to the raw devices, enter the following command:

     ```
     # /sbin/service rawdevices restart
     ```

     The system automatically binds the devices listed in the `rawdevices` file when it reboots.

  5. Repeat steps 2 to 4 on the other nodes in the cluster.

- SUSE:

  1. To determine what raw devices are already bound to other devices, enter the following command on every node:

     ```
     # /usr/sbin/raw -qa
     ```

Raw devices have device names in the form `/dev/raw/rawn`, where *n* is a number that identifies the raw device.

For each device that you want to use, identify a raw device name that is unused on all nodes.

2. Open the `/etc/raw` file in any text editor and add a line similar to the following to associate each partition with an unused raw device:

```
raw1:sdb1
```

3. For each raw device that you specified in the `/etc/raw` file, enter commands similar to the following to set the owner, group, and permissions on the device file:

```
# chown oracle:dba /dev/raw/rawn
# chmod 660 /dev/raw/rawn
```

4. To bind the partitions to the raw devices, enter the following command:

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

5. To ensure that the raw devices are bound when the system reboots, enter the following command:

```
# /sbin/chkconfig raw on
```

6. Repeat steps 2 to 5 on the other nodes in the cluster.

Creating the Database Configuration Assistant Raw Device Mapping File

Note: You must complete this procedure only if you are using raw devices for database files. You do not specify the raw devices for the Oracle Clusterware files in the Database Configuration Assistant raw device mapping file.

To allow Database Configuration Assistant to identify the appropriate raw device for each database file, you must create a raw device mapping file, as follows:

1. Set the ORACLE_BASE environment variable to specify the Oracle base directory that you identified or created previously:
   - Bourne, Bash, or Korn shell:

```
$ ORACLE_BASE=/u01/app/oracle ; export ORACLE_BASE
```

   - C shell:

```
% setenv ORACLE_BASE /u01/app/oracle
```

2. Create a database file subdirectory under the Oracle base directory and set the appropriate owner, group, and permissions on it:

```
# mkdir -p $ORACLE_BASE/oradata/dbname
# chown -R oracle:oinstall $ORACLE_BASE/oradata
# chmod -R 775 $ORACLE_BASE/oradata
```

   In this example, *dbname* is the name of the database that you chose previously.

3. Change directory to the `$ORACLE_BASE/oradata/`*dbname* directory.

4. Edit the *dbname*_raw.conf file in any text editor to create a file similar to the following:

   Note: The following example shows a sample mapping file for a two-instance RAC cluster.

```
system=/dev/raw/raw1
sysaux=/dev/raw/raw2
example=/dev/raw/raw3
users=/dev/raw/raw4
temp=/dev/raw/raw5
undotbs1=/dev/raw/raw6
undotbs2=/dev/raw/raw7
redo1_1=/dev/raw/raw8
redo1_2=/dev/raw/raw9
redo2_1=/dev/raw/raw10
redo2_2=/dev/raw/raw11
control1=/dev/raw/raw12
control2=/dev/raw/raw13
spfile=/dev/raw/raw14
pwdfile=/dev/raw/raw15
```

   Use the following guidelines when creating or editing this file:

   - Each line in the file must have the following format:

     database_object_identifier=raw_device_path

   - For a single-instance database, the file must specify one automatic undo tablespace data file (undotbs1), and at least two redo log files (redo1_1, redo1_2).
   - For a RAC database, the file must specify one automatic undo tablespace data file (undotbs*n*) and two redo log files (redo*n*_1, redo*n*_2) for each instance.
   - Specify at least two control files (control1, control2).
   - To use manual instead of automatic undo management, specify a single RBS tablespace data file (rbs) instead of the automatic undo management tablespace data files.

5. Save the file, and note the file name that you specified.
6. If you are using raw devices for database storage, then set the DBCA_RAW_CONFIG environment variable to specify the full path to the raw device mapping file:

   Bourne, Bash, or Korn shell:

```
$ DBCA_RAW_CONFIG=$ORACLE_BASE/oradata/dbname/dbname_raw.conf
$ export DBCA_RAW_CONFIG
```

   C shell:

```
$ setenv DBCA_RAW_CONFIG=$ORACLE_BASE/oradata/dbname/dbname_raw.conf
```

Part III

Installing Oracle Clusterware and Oracle Database 10*g* with Oracle Real Application Clusters, Creating Oracle Real Application Clusters Databases, and Performing Post-Installation Tasks

Part III describes the two-phase installation process of how to install Oracle Clusterware and Oracle Database 10*g* with Real Application Clusters (RAC). It also explains how to create RAC databases, and it describes the post-installation tasks. The chapters in Part III are:

- Chapter 4, "Installing Oracle Clusterware"
- Chapter 5, "Installing Oracle Database 10g with Real Application Clusters"
- Chapter 6, "Creating Real Application Clusters Databases with Database Configuration Assistant"
- Chapter 7, "Configuring the Server Parameter File in Oracle Real Application Clusters Environments"

4

Installing Oracle Clusterware

This chapter describes the procedures for installing Oracle Clusterware on Linux. If you are installing Oracle Database 10g Real Application Clusters installation, then this is phase one of a two-phase installation. The topics in this chapter are:

- Verifying Oracle Clusterware Requirements with CVU
- Preparing to Install Oracle Clusterware with OUI
- Installing Oracle Clusterware with OUI

Verifying Oracle Clusterware Requirements with CVU

Using the following command syntax, start the Cluster Verification Utility (CVU) to check system requirements for installing Oracle Clusterware:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh stage -pre crsinst -n node_list
```

In the preceding syntax example, replace the variable *mountpoint* with the installation media mountpoint, and replace the variable *node_list* with the names of the nodes in your cluster, separated by commas.

For example, for a cluster with mountpoint /dev/dvdrom/, and with nodes node1, node2, and node3, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh stage -pre crsinst -n
node1,node2,node3
```

The Cluster Verification Utility Oracle Clusterware pre-installation stage check verifies the following:

- **Node Reachability**: All of the specified nodes are reachable from the local node
- **User Equivalence**: Required user equivalence exists on all of the specified nodes
- **Node Connectivity**: Connectivity exists between all the specified nodes through the public and private network interconnections, and at least one subnet exists that connects each node and contains public network interfaces that are suitable for use as virtual IPs (VIPs)
- **Administrative Privileges**: The `oracle` user has proper administrative privileges to install Oracle Clusterware on the specified nodes
- **Shared Storage Accessibility**: If specified, the OCR device and voting disk are shared across all the specified nodes
- **System Requirements**: All system requirements are met for installing Oracle Clusterware software, including kernel version, kernel parameters, memory, swap directory space, temp directory space, and required users and groups

- **Kernel Packages**: All required operating system software packages are installed
- **Node Applications**: The virtual IP (VIP), Oracle Notification Service (ONS) and Global Service Daemon (GSD) node applications are functioning on each node

Troubleshooting Oracle Clusterware Setup

If the CVU report indicates that your system fails to meet the requirements for Oracle Clusterware installation, then use the topics in this section to correct the problem or problems indicated in the report, and run the CVU command again.

User Equivalence Check Failed

Cause: Failure to establish user equivalency across all nodes. This can be due to not creating the required users, or failing to complete SSH configuration properly.

Action: The CVU provides a list of nodes on which user equivalence failed. For each node listed as a failure node, review the Oracle user configuration to ensure that the user configuration is properly completed, and that SSH configuration is properly completed.

> **See Also:** "Creating Identical Users and Groups on Other Cluster Nodes" and "Configuring SSH on All Cluster Nodes" in Chapter 2 for user equivalency configuration instructions.

Use the command su - oracle and check user equivalence manually by running the `ssh` command on the local node with the `date` command argument using the following syntax:

```
ssh node_name date
```

The output from this command should be the timestamp of the remote node identified by the value that you use for `node_name`. If `ssh` is in the default location, the `/usr/bin` directory, then use `ssh` to configure user equivalence. You can also use `rsh` to confirm user equivalence.

If `ssh` is in a location other than the default, `/usr/bin`, then CVU reports a user equivalence check failure. To avoid this error, navigate to the directory `$CV_HOME/cv/admin`, open the file `cvu_config` with a text editor, and add or update the key `ORACLE_SRVM_REMOTESHELL` to indicate the ssh path location on your system. For example:

```
# Locations for ssh and scp commands
ORACLE_SRVM_REMOTESHELL=/usr/local/bin/ssh
ORACLE_SRVM_REMOTECOPY=/usr/local/bin/scp
```

Note the following rules for modifying the `cvu_config` file:

- Key entries have the syntax *name=value*
- Each key entry and the value assigned to the key defines one property only
- Lines beginning with the number sign (#) are comment lines, and are ignored
- Lines that do not follow the syntax *name=value* are ignored

When you have changed the path configuration, run the CVU check again. If ssh is in another location than the default, you also need to start OUI with additional arguments to specify a different location for the remote shell and remote copy commands. Enter `runInstaller -help` to obtain information about how to use these arguments.

Note: When you or the Oracle Universal Installer run ssh or rsh commands, including any login or other shell scripts they start, you may see errors about invalid arguments or standard input if the scripts generate any output. You should correct the cause of these errors.

To stop the errors, remove all commands from the Oracle user's login scripts that generate output when you run ssh or rsh commands.:

If you see messages about X11 forwarding, then perform step 6 in Chapter 2, "Enabling SSH User Equivalency on Cluster Member Nodes" on page 2-13 to resolve this issue.

If you see errors similar to the following:

```
stty: standard input: Invalid argument
stty: standard input: Invalid argument
```

These errors are produced if hidden files on the system (for example, .bashrc or .cshrc) contain stty commands. If you see these errors, then refer to Chapter 2, "Preventing Oracle Clusterware Installation Errors Caused by stty Commands" on page 2-14 to correct the cause of these errors.

Node Reachability Check or Node Connectivity Check Failed

Cause: One or more nodes in the cluster cannot be reached using TCP/IP protocol, through either the public or private interconnects.

Action: Use the command /usr/sbin/ping *address* to check each node address. When you find an address that cannot be reached, check your list of public and private addresses to make sure that you have them correctly configured. If you use third-party vendor clusterware, then refer to the vendor documentation for assistance. Ensure that the public and private network interfaces have the same interface names on each node of your cluster.

User Existence Check or User-Group Relationship Check Failed

Cause: The administrative privileges for users and groups required for installation are missing or incorrect.

Action: Use the `id` command on each node to confirm that the Oracle user is created with the correct group membership. Ensure that you have created the required groups, and create or modify the user account on affected nodes to establish required group membership.

See Also: "Creating Required Operating System Groups and User" in Chapter 2 for instructions about how to create required groups, and how to configure the Oracle user.

Preparing to Install Oracle Clusterware with OUI

Before you install Oracle Clusterware, use the following checklist to ensure that you have all the information you will need during installation, and you have completed all tasks that must be done before starting to install Oracle Clusterware. Mark the check box for each task as you complete it, and write down the information needed, so that you can provide it during installation.

❑ **Shut Down Running Oracle Processes**

If you are installing Oracle Clusterware on a node that already has a single-instance Oracle Database 10g installation, then stop the existing ASM instances. After Oracle Clusterware is installed, start up the ASM instances again. When you restart the single-instance Oracle database and then the ASM instances, the ASM instances use the Cluster Synchronization Services (CSSD) Daemon instead of the daemon for the single-instance Oracle database.

You can upgrade some or all nodes of an existing Cluster Ready Services installation. For example, if you have a six-node cluster, then you can upgrade two nodes each in three upgrading sessions.Base the number of nodes that you upgrade in each session on the load the remaining nodes can handle. This is called a "rolling upgrade."

If a Global Services Daemon (GSD) from Oracle9*i* Release 9.2 or earlier is running, then stop it before installing Oracle Database 10*g* Oracle Clusterware by running the following command from the GSD's Oracle home:

```
ORACLE_home/bin/gsdctl stop
```

where *Oracle_home* is the home that is running the GSD.

Caution: If you have an existing 9.2 Oracle Cluster Manager (Oracle CM) installation, then *do not* shut down the Oracle CM service. Doing so prevents the Oracle Clusterware 10*g* Release 2 (10.2) software from detecting the Oracle 9.2 nodelist, and causes failure of the Oracle Clusterware installation.

Note: If you receive a warning to stop all Oracle services after starting OUI, then run the command

```
Oracle_home/bin/localconfig delete
```

where *Oracle_home* is the home that is running CSS.

- **Prepare for Clusterware Upgrade If You Have Existing Oracle Cluster Ready Services Software**

  During an Oracle Clusterware installation, if OUI detects an existing Oracle Database 10*g* Release 1 (10.1) Cluster Ready Services, then you are given the option to perform a rolling upgrade by installing Oracle Database 10*g* Release 2 (10.2) Oracle Clusterware on a subset of cluster member nodes.

  If you intend to perform a rolling upgrade, then you should shut down the CRS stack on the nodes you intend to upgrade, and unlock the CRS home using the script *mountpoint*`/clusterware/upgrade/preupdate.sh`, which is available on the 10*g* Release 2 (10.2) installation media.

  When you run OUI and select the option to install Oracle Clusterware on a subset of nodes, OUI installs Oracle Database 10*g* Release 2 (10.2) Oracle Clusterware software into the existing CRS home on the local and remote node subset. When you run the root script, it starts the Oracle Clusterware 10*g* Release 2 (10.2) stack on the subset cluster nodes, but lists it as an inactive version.

  When all member nodes of the cluster are running Oracle Clusterware 10*g* Release 2 (10.2), then the new clusterware becomes the active version.

If you intend to install RAC, then you must first complete the upgrade to Oracle Clusterware 10*g* Release 2 (10.2) on all cluster member nodes before you install the Oracle Database 10*g* Release 2 (10.2) version of RAC.

❑ **Prevent Oracle Clusterware Installation Errors Caused by stty Commands**

During an Oracle Clusterware installation, OUI uses SSH (if available) to run commands and copy files to the other nodes. If you see errors similar to the following, then you have hidden files on the system (for example, .bashrc or .cshrc) that contain stty commands:

```
stty: standard input: Invalid argument
stty: standard input: Invalid argument
```

If you see these errors, then stop the installation, and refer to Chapter 2, "Preventing Oracle Clusterware Installation Errors Caused by stty Commands" on page 2-14 to correct the cause of these errors.

❑ **Determine the Oracle Inventory location**

If you have already installed Oracle software on your system, then OUI detects the existing Oracle Inventory directory from the `/etc/oraInst.loc` file, and uses this location.

If you are installing Oracle software for the first time on your system, and your system does not have an Oracle inventory, then you are asked to provide a path for the Oracle inventory, and you are also asked the name of the Oracle Inventory group (typically, `oinstall`).

> **See Also:** The pre-installation chapters in Part II for information about creating the Oracle Inventory, and completing required system configuration.

❑ **Obtain root account access**

During installation, you are asked to run configuration scripts as the root user. You must run these scripts as root, or be prepared to have your system administrator run them for you.

❑ **Decide if you want to install other languages**

During installation, you are asked if you want to install a language other than the default.

❑ **Determine your cluster name, public node names, private node names, and virtual node names for each node in the cluster**

If you install the clusterware during installation, then you are asked to provide a public node name and a private node name for each node.

When you enter the public node name, use the primary host name of each node. In other words, use the name displayed by the `hostname` command. This node name can be either the permanent or the virtual host name.

In addition, ensure that the following are true:

– Determine a cluster name with the following characteristics:
  * It must be globally unique throughout your host domain
  * It must be at least one character long and less than 15 characters long

  * It must consist of the same character set used for host names: underscores (_), hyphens (-), and single-byte alphanumeric characters (a to z, A to Z, and 0 to 9).

- Determine a private node name or private IP address for each node. The private IP address is an address that is only accessible by the other nodes in this cluster. Oracle uses private IP addresses for inter-node, or instance-to-instance Cache Fusion traffic. Oracle recommends that you provide a name in the format *public_hostname*-priv. For example: myclstr2-priv.
- Determine a virtual host name for each node. A virtual host name is a public node name that is used to reroute client requests sent to the node if the node is down. Oracle uses VIPs for client-to-database connections, so the VIP address must be publicly accessible. Oracle recommends that you provide a name in the format *public_hostname*-vip. For example: myclstr2-vip

---

Note: The following is a list of additional information about node IP addresses:

The IP addresses that you use for all of the nodes in the current installation process must be from the same subnet.

If you create private and virtual host names in the format *nodename*-priv and *nodename*-vip, then OUI looks up these addresses and fills them in by default during the installation. Otherwise, you are required to provide node names using the default formats.

Host names, private names, and virtual host names are not domain-qualified. If you provide a domain in the address field during installation, then OUI removes the domain from the address.

Private IP addresses should not be accessible as public interfaces. Using public interfaces for Cache Fusion can cause performance problems.

❑ **Identify shared storage for Oracle Clusterware files and prepare disk partitions if necessary**

During installation, you are asked to provide paths for two files that must be shared across all nodes of the cluster, either on a shared raw device, or a shared file system file:

- The voting disk is a partition that Oracle Clusterware uses to verify cluster node membership and status.

  The voting disk must be owned by the oracle user, must be in the `dba` group, and must have permissions set to 644. Provide at least 20 MB disk space for the voting disk.
- The Oracle Cluster Registry (OCR) contains cluster and database configuration information for the RAC database and for Oracle Clusterware, including the node list, and other information about cluster configuration and profiles.

  The OCR disk must be owned by root, must be in the `oinstall` group, and must have permissions set to 640. Provide at least 100 MB disk space for the OCR.

If your disks do not have external storage redundancy, then Oracle recommends that you provide one additional location for the OCR disk, and two additional locations for the voting disk. Creating redundant storage locations protects the OCR and voting disk in the event of a disk failure on the partitions you choose for the OCR and the voting disk.

See Also: The pre-installation chapter in Part II for information about the minimum raw device sizes

Installing Oracle Clusterware with OUI

This section provides you with information about how to use the Oracle Universal Installer (OUI) to install Oracle Clusterware.

Running OUI to Install Oracle Clusterware

Complete the following steps to install Oracle Clusterware on your cluster. At any time during installation, if you have a question about what you are being asked to do, click the **Help** button on the OUI page.

1. Start the `runInstaller` command from the `clusterware` directory on the10*g* Release 2 (10.2) installation media. When OUI displays the Welcome page, click **Next**.
2. Provide information or run scripts as root when prompted by OUI. If you need assistance during installation, click **Help**.
3. After you run `root.sh` on all of the nodes, OUI runs the Oracle Notification Server Configuration Assistant, Oracle Private Interconnect Configuration Assistant, and Cluster Verification Utility. These assistants run without user intervention.

When you have verified that your Oracle Clusterware installation is completed successfully, Oracle Clusterware installation is complete.

If you intend to install Oracle Database 10*g* with RAC, then continue to Chapter 5, "Installing Oracle Database 10g with Real Application Clusters". If you intend to use Oracle Clusterware by itself, then refer to the single-instance Oracle Database installation guide.

Troubleshooting Oracle Clusterware Installation Verification

If the CVU report indicates that your Oracle Clusterware installation has a component issue, then use the topics in this section to correct the problem or problems indicated in the report, and run the CVU command again.

CSS is probably working with a non-clustered, local-only configuration on nodes:

Cause: OCR configuration error. The error message specifies the nodes on which this error is found.

This error occurs when, for each specified node, either the contents of the OCR configuration file `ocr.loc` cannot be retrieved, or the configuration key `local_only` is set to TRUE in the configuration file of nodes listed in the error message.

Action: Confirm that Oracle Clusterware was installed on the node. Correct the OCR configuration, if it is incorrect. Also, ensure that you have typed the node name correctly when entering the CVU command.

Unable to obtain OCR integrity details from nodes:

Cause: Unable to run the `ocrcheck` tool successfully on the nodes listed in the error message.

Action: If the ocrcheck tool indicates an error on only some nodes in the cluster, then OCR is not configured on that set of nodes. If the ocrcheck tool indicates that the OCR integrity check failed on all nodes, then the OCR storage area is

corrupted. Refer to *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for instructions about how to use ocrconfig -repair to resolve this issue.

To configure OCR, you can use `ocrconfig -repair`, as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*, or you can configure OCR manually.

To configure OCR manually, as the Oracle user, enter the following command from the bin directory in the CRS home:

```
$ ./ocrcheck
```

To test whether the OCR storage area is corrupted, complete the following task:

1. Enter the following command:

   ```
   ocrconfig -showbackups
   ```

2. View the contents of the OCR file using the following command syntax:

   ```
   ocrdump -backupfile OCR_filename
   ```

3. Select a backup file, and use the following command to attempt to restore the file:

   ```
   ocrconfig -restore backupfile
   ```

   if the command returns a failure message, then both the primary OCR and the OCR mirror have failed.

   See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for additional information about testing and restoring the Oracle Cluster Registry.

OCR version is inconsistent amongst the nodes.

Cause: The OCR version does not match on all the cluster member nodes. Either all nodes are not part of the same cluster, or nodes do not point to the same OCR, or an OCR configuration file has been changed manually to an invalid configuration on one or more nodes

Action: Perform the following checks:

1. Ensure that all listed nodes are part of the cluster.
2. Use the `ocrcheck` utility (`/crs/home/bin/ocrcheck`) to find the location of OCR on each node. Start `ocrcheck` with one of the following command:

   As root:

   ```
   # ocrcheck
   ```

   As the Oracle user, or as a user with OSDBA group privileges, from the user home directory:

   ```
   $ /crs/home/bin/ocrcheck
   ```

3. Repair invalid OCR configurations by logging into a node you suspect has a faulty configuration, stopping the CRS daemon, and entering the following command:

   ```
   ocrconfig –repair ocrmirror device_name
   ```

the `ocrconfig -repair` command only changes the OCR configuration on the node from which you run the command.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for information about how to use the `ocrconfig` tool to repair OCR files.

Incorrect OCR version found for nodes:

Cause: the OCR version on the specified nodes does not match the version required for 10*g* Release 2 (10.2).

Action: Follow the same actions described in the preceding error message, "OCR version is inconsistent amongst the nodes." on page 4-8.

OCR integrity is invalid.

Cause: The data integrity of the OCR is invalid, which indicates that OCR storage is corrupted

Action: Follow the same actions described in the preceding error message, "Unable to obtain OCR integrity details from nodes:" on page 4-7.

OCR ID is inconsistent amongst the nodes.

Cause: One or more nodes list the OCR in a different location.

Action: Follow the same actions described in the preceding error message, "OCR version is inconsistent amongst the nodes." on page 4-8

Oracle Clusterware Background Processes

The following processes must be running in your environment after the Oracle Clusterware installation for Oracle Clusterware to function:

- `oprocd`: Process monitor for the cluster.
- `evmd`: Event manager daemon that starts the `racgevt` process to manage callouts.
- `ocssd`: Manages cluster node membership and runs as `oracle` user; failure of this process results in node restart.
- `crsd`: Performs high availability recovery and management operations such as maintaining the OCR. Also manages application resources and runs as `root` user and restarts automatically upon failure.

5

Installing Oracle Database 10*g* with Real Application Clusters

This chapter describes phase two of the installation procedures for installing Oracle Database 10*g* with Real Application Clusters (RAC). This chapter also describes some of the Oracle Universal Installer (OUI) features. This chapter contains the following topics:

- Verifying System Readiness for installing Oracle Database with CVU
- Selecting a Configuration Option
- Installation of Oracle Database 10g with RAC Using the Oracle Universal Installer
- De-Installing Real Application Clusters Software

Verifying System Readiness for installing Oracle Database with CVU

To help to verify that your system is prepared to install the Oracle Database with RAC successfully, enter a Cluster Verification Utility (CVU) command using the following command syntax:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh stage -pre dbinst -n node_list [-r
{10gR1|10gR2}] [-osdba osdba_group][-verbose]
```

In the preceding syntax example:

- The variable *mountpoint* is the mountpoint of the installation media
- The variable *node_list* is the list of nodes in your cluster, separated by commas
- The optional flag -r takes the options 10gR1 (test for an Oracle Database 10*g* Release 1 Oracle Cluster Ready Services installation) or 10gR2 (test for an Oracle Database 10g Release 2 Oracle Clusterware installation); if the -r flag is missing, by default, the command tests for Oracle Clusterware for Oracle Database 10*g* Release 2 (10.2) installation
- The variable *osdba_group* is the name of your OSDBA group, which by default is `dba`

For example, to perform a pre-installation check for an Oracle Database 10*g* Release 2 (10.2) with RAC installation on a two-node cluster with the mountpoint `/dev/dvdrom`, with nodes node1 and node2, and with the OSDBA group `dba`, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh stage -pre dbinst -n node1,node2
-verbose
```

Oracle recommends that you select the option `-verbose` to receive progress updates as the CVU performs its system checks. The -verbose option provides detailed test reporting, which you can use to identify the cause of any checks that fail.

If the cluster verification check fails, then review and correct the relevant system configuration steps, and run the test again. Use the system configuration checks described in "Troubleshooting Installation Setup for Linux" to assist you.

Troubleshooting Installation Setup for Linux

If you run the CVU and your system fails system configuration checks, then review the the CVU report, and use the output to resolve failed configuration checks.

User Equivalence Check Failed

Cause: Failure to establish user equivalency across all nodes.

Action: Check user equivalence on each node for the Oracle account.

Check for user equivalence for the oracle account by performing a remote login (rlogin) to each node in the cluster. For example:

```
# su - oracle
$ rlogin node_name
```

If you are prompted for a password, then you have not given the oracle account the same attributes on all nodes.

Check permissions on each node to ensure that the oracle user can write to the new mount points. For example:

```
# su - oracle
$ touch /u01/test
$ ls -l /u01/test
-rw-rw-r-- 1 oracle dba 0 Aug 15 09:36 /u01/test
```

As the `oracle` user, use SSH to connect from each node to itself, and from each node to each other node. For example, to check a two-node cluster with nodes named node1 and node2:

```
[oracle@node1 oracle]:$ ssh node1
[oracle@node1 oracle]:$ ssh node2
[oracle@node2 oracle]:$ ssh node2
[oracle@node2 oracle]:$ ssh node1
[oracle@node1 oracle]:$
```

If you are prompted for a password when logging in as the `oracle` user from one node to another in the cluster, then improper SSH configuration is a probable cause. Check to ensure that you have turned off X11 forwarding, and that the ~/.ssh/id_rsa and ~/.ssh/dsa.pub files for all nodes are listed in the ~/.ssh/authorized_keys file that you create during pre-installation.

> **See Also:** "Configuring SSH on All Cluster Nodes" in Chapter 2, "Pre-Installation Tasks" on page 2-11.

Node Reachability Check Failed

Cause: Failure of one or more nodes to be properly connected for communication.

Action: Possible causes of this message include the following:

- Improper network configuration

- The node running CVU is unable to connect to one or more nodes in the cluster

Use the following command to check your current configuration on each node:

```
ifconfig -a
```

> **See Also:** "Checking the Network Requirements" in Chapter 2, "Pre-Installation Tasks" on page 2-19.

Node Connectivity Check Failed

Cause: One or more of the cluster nodes is not able to be connected from all nodes in the cluster

Action: Check for firewalls preventing the nodes from communicating on their private network interfaces.

User Existence Check Failed or User-Group Relationship Check Failed

Cause: The administrative privileges for users and groups required for installation are missing or incorrect.

Action: Use the id command on each node to confirm that the oracle user is created with the correct group membership.

> **See Also:** "Creating Required Operating System Groups and User" in Chapter 2, "Pre-Installation Tasks" on page 2-5

Shared Storage Check Failed

Cause: Shared storage access from each node in the cluster is not confirmed.

Action: If you are using shared disk storage, then as root, use the command `fdisk -l` to confirm the reachability of shared storage.

System Requirements Check Failed

Cause: insufficient system resources, missing software packages, or other operating system or hardware problem.

Action: If you did not run the CVU command with the -verbose flag, then run the command again using -verbose, and review the report to determine which system requirement failed. Correct the problem.

> **See Also:** Chapter 2, "Pre-Installation Tasks" contains instructions for completing any system requirement configuration that CVU lists as incomplete.

Selecting a Configuration Option

This section describes OUI features that you should understand before beginning phase two of the RAC installation process.

> **See Also:** Appendix D, "Converting to Oracle Real Application Clusters from Single-Instance Oracle Databases" for information about how to convert single-instance databases to RAC

On the Select Configuration page, you are given the following options:

- **Create a database**: Provides you with the option to create a database using a preconfigured database template designed for particular system load demands, as

described in the section "Database Configuration Type Descriptions" on page 5-4. If you select the Automatic Storage Management option for storage, then an ASM instance is installed as part of the database installation.

- **Configure Automatic Storage Management**: Installs an ASM instance only. You are prompted for the ASM SYS password
- **Install database software only**: Installs Oracle software; you must complete database configuration later.

If you are installing Oracle Database, then Oracle recommends that you use one of the preconfigured database options, or select the Advanced option and configure a custom starter database.

If you want to configure your environment and create your database manually, then select the **Do not create a database** configuration option, and refer to the manual database creation procedures posted at the following Web site:

`http://otn.oracle.com`

Automatic Storage Management Configuration

If you want to use Automatic Storage Management, then select the **Configure Automatic Storage Management (ASM)** configuration option, and provide information as prompted. Click **Help** if you need assistance.

Database Configuration Type Descriptions

When you run OUI and select Oracle Database 10*g*, you can select the **General Purpose**, **Transaction Processing**, **Data Warehouse**, or **Advanced** database configuration type.

For the first three configuration types, you can complete additional procedures that are described later in this chapter. If you select Advanced configuration, then you can use Database Configuration Assistant (DBCA) to create the database as described in Chapter 6. Oracle recommends that you use DBCA to create your database.

General Purpose, Transaction Processing, and Data Warehouse Configuration Types

The General Purpose, Transaction Processing, and Data Warehouse configuration types use preconfigured database templates.

During installation, if you select one of the preconfigured database types, then OUI starts Oracle Network Configuration Assistant (NETCA) and DBCA, and installs the preconfigured database without further input. During database installation, OUI displays a progress indicator.

DBCA processing for these three configuration types creates a starter database, and configures the Oracle network services. If you choose raw devices on the Specify Database File Storage Option page, then DBCA verifies that you configured the raw devices for each tablespace.

If you select Advanced configuration, then you must enter specific information as described in the next section.

Using the Advanced Configuration Type

If you select the Advanced configuration type, then OUI runs DBCA, which displays General Purpose, Transaction Processing, Data Warehouse, and a fourth configuration type, Custom Database.

The first three templates are customizable versions of the preconfigured database types. The Custom Database type creates a database without using preconfigured options.

The following section provides more detail about OUI and DBCA processing when creating a RAC database.

Behavior of OUI, DBCA, and Other Assistants During Installation

After installation, OUI starts the NETCA. After the NETCA completes its processing, OUI runs DBCA to create your database using Optimal Flexible Architecture (OFA). This means that DBCA creates your database files, including the default server parameter file (SPFILE), using standard file naming and file placement practices. The primary phases of DBCA processing are:

- Verify that you correctly configured the shared disks for each tablespace if you are using raw storage
- Create the database
- Configure the Oracle network services
- Start the listeners and database instances

You can also use DBCA in standalone mode to create a database.

> **See Also:** The *Oracle Database Net Services Administrator's Guide* if you experience problems, for example, with the listener configuration, and for further information about LDAP support

You can use your Oracle9*i* database language and territory definition files with Oracle Database 10*g* Release 2 (10.2) that you are about to install. To enable this functionality, you must run OUI from a command line, described in Step 1 in "Installation of Oracle Database 10g with RAC Using the Oracle Universal Installer" on page 5-5, and use the following statement to set the `b_cr9idata` variable to `true`:

```
runInstaller oracle.rsf.nlsrtl_rsf:b_cr9idata=true
```

The remainder of this chapter explains how to use OUI to install Oracle Database 10*g* with RAC.

Installation of Oracle Database 10*g* with RAC Using the Oracle Universal Installer

Perform the following procedures to install the Oracle Database 10*g* software with RAC.

1. Start the `runInstaller` command from the DB directory of the Oracle Database 10*g* Release 2 (10.2) installation media. When OUI displays the Welcome page, click **Next**.
2. Provide information or run scripts as root when prompted by OUI. If you need assistance during installation, click **Help**. If you encounter problems during installation, then examine the OUI actions recorded in the installation log file. The

log file is located in the OraInventory directory (typically, `oinstall`), in the following location:

OraInventory`/logs/installActions`*date_time*`.log`

Note: The Oracle home name and path that you provide during database installation *must be different* from the home that you used during the Oracle Clusterware installation in phase one. You **must not** install Oracle Database 10*g* with RAC software into the same home in which you installed the Oracle Clusterware software.

The following is a list of additional information to note about installation:

- If you are not using the ASM library driver (ASMLIB), and you select Automatic Storage Management (ASM) during installation, then ASM default discovery finds all disks that ASMLIB marks as ASM disks.
- If you are not using ASMLIB, and you select ASM during installation, then ASM default discovery finds all disks marked `/dev/raw/*` for which the Oracle user has read/write permission. You can change the disk discovery string during the install if the disks you want to use for ASM are located elsewhere.

  If you are installing RAC from the Standard Edition, then you must use ASM for your database storage.

  On the Select Database Management Option page, if you have already completed the Grid Control Management Agent installation, then you can select either Grid or Local Database control. Otherwise, only Local Database control for database management is supported for RAC. When you use the local Database Control, you can choose the email option and enter the outgoing SMTP server name and email address.

  If you perform an installation that does not include Enterprise Manager, for example, a custom software install without Enterprise Manager, an installation with no Enterprise Manager configuration, or a database creation with your own scripts, then you can configure Enterprise Manager later with OUI, DBCA, or the Enterprise Manager Configuration Assistant (EMCA) utility.

  See Also: *Enterprise Manager Grid Control Installation and Basic Configuration* for details about installing Grid Control with OUI, and *Enterprise Manager Advanced Configuration Guide* for details about installing Database Control with DBCA and EMCA

When you have completed the second and final phase of the installation, proceed to Chapter 8, "Oracle Real Application Clusters Post-Installation Procedures" to perform the post-installation tasks.

Caution: After installation is completed and you have created the database, if you decide that you want to install additional Oracle Database 10g products in the 10g Release 2 (10.2) database, then you must stop *all* processes running in the Oracle home before you attempt to install the additional products. For the Oracle Universal Installer to relink certain executables and libraries, all database processes must be down. Refer to Appendix F, "How to Stop Processes in an Existing Oracle Real Application Clusters Database" for additional information.

De-Installing Real Application Clusters Software

If you need to de-install Real Application Cluster software, then you must run OUI to de-install the software on the same node from which you performed the installation, and you must de-install the Oracle database software first before de-installing the Oracle Clusterware software.

Perform the following procedures to de-install Oracle Database 10*g* RAC and Oracle Clusterware software, as described in the following sections:

- De-Installing Oracle Database 10g RAC Software and ASM
- De-Installing Oracle Clusterware

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about using RAC scalability features of adding and deleting nodes and instances from RAC databases, and for information about viewing OCR content

De-Installing Oracle Database 10*g* RAC Software and ASM

This section describes the procedure to de-install the Oracle Database 10*g* RAC software and Automatic Storage Management (ASM) software. Before you perform these steps, Oracle recommends that you make a backup of any databases that run from the Oracle home you are about to delete.

1. Examine the `oratab` file to identify the instance dependencies on this Oracle home.
2. Start DBCA, select **Oracle Real Application Clusters Database**, select Delete a database, and select the database that you want to delete. Repeat this step to delete all databases.
3. If you do not use ASM, then proceed to step 4.

   If the database is in the Oracle home from which the ASM database runs, then ensure that there are no other database dependencies on these group of ASM instances, and then remove the ASM configuration by logging on as the `oracle` user and completing the following steps:

   a. Connect to the ASM instance and run the following command to determine database instances using this ASM instance.:

   ```
   SQL> select INSTANCE_NAME from GV$ASM_CLIENT;
   ```

Note: This command only lists database instances that are running. It is possible that other instances are associated with the ASM instance, but they are not currently running. If you removed a database from this Oracle home but the output from the command shows that this ASM instance is supporting a database instance in another Oracle home, then do not remove the ASM instance or the Oracle home.

If your RAC database uses a different Oracle home from the Oracle home that your ASM environment uses, and you want to delete the RAC database, then perform step 5 in the RAC database home.

---

b. For each instance listed in the output of the statement you run in Step a, stop the respective databases.

c. Oracle recommends that you back up the database files for all of the databases that are currently using this ASM instance.

d. Using your connection to the ASM instance, run the following command:

```
SQL> select * from V$ASM_DISKGROUP;
```

e. For each diskgroup listed in the output of the statement you run in Step d, run the following command:

```
SQL> drop diskgroup diskgroup_name including contents;
```

where *diskgroup_name* is the name of the diskgroup.

f. Shut down ASM on all RAC nodes, and verify that all ASM instances are stopped.

g. If you are de-installing from a system with ASMLIB, then run the following procedure. First, run this command to list the disks that you need to delete:

```
oracleasm listdisks
```

Second, run the following command to delete every disk listed by the previous command:

```
oracleasm deletedisks
```

Third, re-run the `oracleasm listdisks` command to confirm that the disks are all deleted. Repeat this command on each of the RAC cluster nodes confirm that the disks have been deleted from all the nodes.

Fourth, as `root`, run the following commands on all nodes of your RAC cluster:

```
/etc/init.d/oracleasm stop
/etc/init.d/oracleasm disable
```

h. Deconfigure ASM, using either DBCA in silent mode, or performing deconfiguration manually.

To deconfigure ASM manually, complete steps i through l.

To deconfigure ASM using DBCA, run DBCA in silent mode to deconfigure ASM. To run DBCA in silent mode, navigate to the directory $ORACLE_HOME/bin and use the following command syntax:

```
dbca -silent -deleteASM -nodelist node1,node2,...
```

for the variables *node1*, *node2*, and so on, provide a list of all the nodes where ASM is configured. When DBCA completes de-installation, proceed to step 4.

i. To remove the ASM entry from the OCR, run the following command for all nodes on which this Oracle home exists:

```
srvctl remove asm -n nodename
```

where *nodename* is the name of a node from which you want to remove the ASM instance.

j. If you are using a shared cluster file system for your Oracle home, then run the following commands on the local node:

```
rm -f $ORACLE_HOME/dbs/*ASM*
rm -r $ORACLE_BASE/admin/+ASM
```

You may need to remove subordinate files or directories before these commands complete successfully.

k. If you are not using a shared cluster file system for your Oracle home, then run the commands from the previous step, Step j, on *each* node on which the Oracle home exists.

l. Remove `oratab` entries for the deleted Oracle home databases.

4. If there are no outstanding dependencies, then change directory to the `$ORACLE_HOME/bin` directory, and use the following NETCA command syntax to remove the listener and its Oracle Clusterware resources.

```
$ ./netca /deinst /nodeinfo node1,node2,...
```

In the preceding syntax example, replace the variables *node1*, *node2*, and so on, with all the member nodes of the RAC database.

Note: You cannot perform a RAC installation from the same OUI session in which you perform a RAC de-installation. In other words, if you de-install RAC with OUI and want to perform another RAC installation, then you must start a new OUI session.

---

5. Start OUI, and remove any existing Oracle Database 10g with RAC software by selecting **Deinstall Products**, and selecting the Oracle home that you want to remove. If any of these databases is an ASM database, then de-install the Oracle home with ASM where you want to de-install the Oracle Database software, and then de-install the Oracle Database. Removing ASM does not remove the ASM data or disk groups.

De-Installing Oracle Clusterware

De-install each Oracle Database 10*g* RAC home by running the procedure in the previous section, "De-Installing Oracle Database 10g RAC Software and ASM". Then complete the de-installation by removing the Oracle Clusterware software using the following procedures:

1. Run the command *CRS_home*`/install/rootdelete.sh` to disable the Oracle Clusterware applications that are running on the cluster node. The `rootdelete.sh` script requires three arguments. If you are running this

command on a remote node of the cluster, then use `remote` as the first argument, otherwise use `local` as the first argument. If the `ocr.loc` file is on a shared file system, then use `sharedvar`. Otherwise use `nosharedvar` as the second argument. If the Oracle Clusterware home is on a shared file system, then use `sharedhome`, otherwise use `nosharedhome` as the third argument. Repeat this step on each node of the cluster from which you want to de-install Oracle Clusterware.

Note: A node is a local node if you plan to run Steps 2 and 3 on that node. Removing shared OCR contents and the Oracle Clusterware home is done on the local node after you complete removal of other nodes in the cluster.

---

2. Run the script *`CRS_home`*`/install/rootdeinstall.sh` on a local node to remove the OCR. You only need to run this script once.
3. Start OUI. In the Welcome page, click **Deinstall Products** to display the list of installed products. Select the Oracle Clusterware home to de-install.

6

Creating Real Application Clusters Databases with Database Configuration Assistant

This chapter describes how to use Database Configuration Assistant (DBCA) in standalone mode to create and delete Real Application Clusters (RAC) databases. The topics in this chapter include:

- Using Database Configuration Assistant in Real Application Clusters
- Benefits of Using Database Configuration Assistant
- Oracle Real Application Clusters High Availability Services
- Automatic Listener Migration from Release 10.1 to 10.2
- Creating an Oracle Real Application Clusters Database with DBCA
- Deleting a Real Application Clusters Database with DBCA

> **See Also:** *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for procedures on using Database Configuration Assistant (DBCA) to add and delete instances

Using Database Configuration Assistant in Real Application Clusters

DBCA has the following primary functions:

- Create the database and its instances
- Set up network configuration for database, instances and database services
- Register the database in Enterprise Manager Grid Control or configure Database Control
- Configure Automatic Storage Management (ASM)
- Start up the database, its instances, services, and any other node applications

See Also:

- "Creating an Oracle Real Application Clusters Database with DBCA" on page 6-4 for more information about using DBCA in standalone mode
- *Oracle Database Net Services Administrator's Guide* to resolve problems—for example, with the listener configuration—and for further information about Lightweight Directory Access Protocol (LDAP)-compliant directory support

Benefits of Using Database Configuration Assistant

Oracle recommends that you use Database Configuration Assistant (DBCA) to create your RAC database, because DBCA's preconfigured databases optimize your environment for features such as Automatic Storage Management (ASM), the server parameter file, and automatic undo management. DBCA also provides pages to create new ASM disk groups if they are needed. If you use ASM or cluster file system storage, then DBCA also configures automated backup, which uses the flash recovery area.

With DBCA, you can create site-specific tablespaces as part of database creation. If you have data file requirements that differ from those offered by DBCA templates, then create your database with DBCA and modify the data files later. You can also run user-specified scripts as part of your database creation process.

DBCA also configures your RAC environment for various Oracle high availability features, such as services and cluster administration tools. It also starts any database instances required to support your defined configuration.

Oracle Real Application Clusters High Availability Services

When you configure high availability services with the DBCA Database Services page, you can also configure service instance preferences and Transparent Application Failover (TAF) policies.

Service Configuration and Instance Preferences

Use the Database Services page button in the column labeled Not Used, Preferred, or Available to configure service instance preferences as described in the following list:

- Preferred—The service runs primarily on the selected instance
- Available—The service may run on the instance if a preferred instance fails
- Not Used—The service never runs on the instance

Note: You can assign services to run on multiple preferred instances, and fail over to multiple available instances.

After you have created the database, you can configure service instance preferences through Oracle Enterprise Manager Database Control or Grid Control.

Transparent Application Failover Policies

Use the DBCA Database Services page to configure TAF failover policies. The DBCA Database Services page also has a TAF policy selector row under the instance

preference display. Select one of the following options in this row for your failover and reconnection policy preference:

- None—Do not use TAF
- Basic—Establish connections at failover time

Automatic Listener Migration from Release 10.1 to 10.2

If your system has an Oracle Database Release 10*g* Release 10. 1 installation, and you install an Oracle Database 10*g* Release 2 (10.2) either to coexist with or to upgrade the Oracle Database 10.1 installation, then most installation types automatically migrate the Oracle Database 10.1 listener to the 10*g* Release 2 (10.2) Oracle home. During migration, they configure and start a default Oracle Net listener using the same TCP/IP port as the existing listener, with the IPC key value EXTPROC. This process occurs through one of the following scenarios:

- During a coexisting installation, Database Configuration Assistant (DBCA) automatically migrates the listener and related files from the 10.1 Oracle home to the 10.2 Oracle home.
- During an upgrade, Oracle Database Upgrade Assistant (DBUA) automatically locates the Oracle 10*g* release 1 (10.1) listener, and migrates it to Oracle 10g release 2.

The listener migration process stops the listener in the existing Oracle home, and restarts the listener from the new Oracle home. During migration, client applications may not be able to connect to any databases that are registered to the listener that is being migrated.

Verifying Requirements for DBCA

To help to verify that your system is prepared to create the Oracle Database with RAC successfully, enter a Cluster Verification Utility command using the following command syntax:

```
/mountpoint/clusterware/cluvfy/runcluvfy.sh stage -pre dbcfg -n node_list -d
oracle_home [-verbose]
```

In the preceding syntax example, the variable *mountpoint* is the mountpoint of the installation media, the variable *node_list* is the list of nodes in your cluster, separated by commas, and the variable *oracle_home* is the path for the Oracle home directory where OUI creates or modifies the database.

For example, to perform a check to determine if your system is prepared for an Oracle Database with RAC on a two-node cluster with nodes node1 and node2, with the mountpoint `/dev/dvdrom/`, and with the Oracle home path /oracle/product/10.2.0, enter the following command:

```
/dev/dvdrom/clusterware/cluvfy/runcluvfy.sh stage -pre dbcfg -n node1,node2 -d
/oracle/product/10.2.0/
```

You can select the option `-verbose` to receive progress updates as the CVU performs its system checks, and detailed reporting of the test results.

If the Cluster Verification Utility summary indicates that the cluster verification check fails, then review and correct the relevant system configuration steps, and run the test again.

The command `runcluvfy.sh stage -pre dbcfg` verifies the following:

- Node Reachability: All the specified nodes are reachable from the local node
- User Equivalence: User equivalence exists on all the specified nodes
- Node Connectivity: Connectivity exists between all the specified nodes through the available public and private network interfaces
- Administrative Privileges: The `oracle` user has proper administrative privileges on the specified nodes for creating a RAC database
- Oracle Clusterware Integrity: All the components of the Oracle Clusterware stack are fully operational

Creating an Oracle Real Application Clusters Database with DBCA

To create a database with DBCA in standalone mode without ASM or a cluster file system, you must have configured each raw device as described in Appendix C. In addition, you must have run the Oracle Net Configuration Assistant to configure your Oracle Net `listener.ora` file.

If you select DBCA templates that use preconfigured data files and if you do not use ASM or a cluster file system, then during database creation, DBCA first verifies that you created the raw devices for each tablespace. If you have not configured the raw devices, then you must configure the raw devices and replace the default data file names that DBCA provides with raw device names on the DBCA Storage page to continue database creation.

To start DBCA, connect to one of your nodes with Oracle RAC installed as the Oracle owner for the $ORACLE_HOME, and enter the command `dbca` command from the `$ORACLE_HOME/bin` directory.

When you start DBCA, the first page it displays is the Welcome page for RAC, which includes the option to select an Oracle Real Application Clusters (RAC) database. DBCA displays this RAC-specific Welcome page only if the Oracle home from which it is started was cluster-installed.

If DBCA does not display the Welcome page for RAC, then DBCA was unable to detect whether the Oracle home is cluster-installed. In this case, check that the OUI inventory is correctly located in `/etc/oraInst.loc`, and that `oraInventory` is not corrupted. Also, perform clusterware diagnostics by running the CVU command */mountpoint*/clusterware/cluvfy/runcluvfy.sh stage -post crsinst -n *nodename*.

If the RAC Welcome page opens then provide information as prompted by DBCA. Click **Help** if you need assistance.

The following is a list of information you should be aware of when using DBCA:

- If nodes that are part of your cluster installation do not appear on the Node Selection page, then run the `olsnodes` command to perform inventory diagnostics and clusterware diagnostics.
- The global database name can be up to 30 characters in length, and must begin with an alphabetical character. The SID prefix must begin with an alphabetical character.
- The maximum number of characters you can use for the SID prefix is 8 characters. DBCA uses the SID prefix to generate a unique value for the variable `ORACLE_SID` for each instance.
- On the Management Options page, If you select the option **Enterprise Manager with the Grid Control**, and DBCA discovers agents. If you select the option **Database Control**, then you can set up E-mail notification and enable daily

backups. For E-mail notifications, you provide the outgoing mail server and E-mail address. For daily backups, you enter the backup time and operating system credentials for the user that performs backup.

- To use a flash recovery area, Oracle recommends that you create two separate ASM disk groups: one for the database area and one for the recovery area.

  See Also: *Oracle Database Concepts* for more information about using a flash recovery area

- On the ASM Disk Groups page, if you do not see the disks you wish to add, click **Change Disk Discovery Path** to alter the search path used by DBCA to find available disks. You can select disks with a status of `Candidate` or `Former` (never used in an ASM disk group or no longer in a group) by checking the select box. If you want to add disks that still have ASM disk headers but the disk group is no longer in use (a case that can occur if you are selecting disks after an aborted install, a de-install performed without dropping the disk group, or other configuration problems), then use the `Force` command.
- If DBCA displays the following message:

  ```
  The file oracle_home/bin/oracle does not exist on node node_name.
  Make sure that file exists on these nodes before proceeding.
  ```

  This message means that the Oracle home from which the first ASM instance in the cluster runs is not installed on these cluster nodes. You must extend the ASM Oracle home to these nodes by performing the procedure documented in "Step 4: Adding Nodes at the Oracle RAC Database Layer" in the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*. However, do not perform Step 5 in that section. OUI extends the ASM Oracle home to the selected nodes and performs any configuration required for running an ASM instance on these nodes.
- If DBCA displays the following message:

  ```
  Please run the DBCA from one of the nodes that has an existing ASM instance
  node_list.
  ```

  This message means that you are attempting to create a RAC database using ASM storage, but the ASM instance does not exist on the node from which you ran DBCA. However, ASM instances do exist on the remote nodes that appear in the message node list. In this case, DBCA cannot clone the existing ASM instance from the remote node to the local node. To correct this, start DBCA from one of the nodes shown in the node list to create your RAC database using ASM storage. This copies the local node's ASM instance and modifies its parameters and attributes to create ASM instances on the nodes in your cluster that do not have ASM instances.
- On the Recovery Configuration page, if you are using ASM or CFS storage, then you can also select the flash recovery area and size on the Recovery Configuration page. If you are using ASM, then the flash recovery area defaults to the ASM Disk Group. If you are using CFS, then the flash recovery area defaults to `$ORACLE_BASE/flash_recovery_area`.
- On the Initialization Parameters page, set the value of the `CLUSTER_DATABASE_INSTANCES` parameter to the number of instances you intend to use in the cluster if you are not including all the related nodes during the current execution of DBCA.

In addition, if your global database name is longer than eight characters, then the database name value (in the db_name parameter) is truncated to the first eight characters and the DB_UNIQUE_NAME parameter value is set to the global name.

After you respond to DBCA prompts, review the Summary dialog information and click **OK**, DBCA does the following:

- Creates an operative RAC database and its instances
- Creates the RAC data dictionary views
- Configures the network for the cluster database
- Starts the Oracle services if you are on a Windows-based platform
- Migrates Oracle Database 10g Release 1 (10.1) listener and related files to the 10.2 Oracle home
- Starts the listeners and database instances and then starts the high availability services
- Configures Enterprise Manager Database Control or Grid Control

Caution: After you have created the database, if you decide that you want to install additional Oracle Database 10*g* products in the 10*g* Release 2 (10.2) database you have created, then you must stop *all* processes running in the Oracle home before you attempt to install the additional products. For the Oracle Universal Installer to relink certain executables and libraries, all database processes must be down. Refer to Appendix F, "How to Stop Processes in an Existing Oracle Real Application Clusters Database" for additional information.

Deleting a Real Application Clusters Database with DBCA

This section explains how to delete a RAC database with DBCA. This process deletes a database and removes a database's initialization parameter files, instances, OFA structure, and Oracle network configuration. However, this process does not remove data files if you placed the files on raw devices or on raw partitions.

To delete a database with DBCA:

1. Start DBCA on one of the nodes
   - Run the `DBCA` command from the `$ORACLE_HOME/bin` directory

   The DBCA Welcome page appears.
2. Select Oracle Real Application Clusters and click **Next**.

   After you click **Next**, DBCA displays the Operations page.
3. Select Delete a database, click **Next**. DBCA displays the List of Cluster Databases page.
4. If your user ID and password are not operating-system authenticated, then the List of Cluster Databases page displays the user name and password fields. If these fields appear, then enter a user ID and password that has SYSDBA privileges.

   See Also: "Database Password and Role Management in Real Application Clusters" on page B-19

5. Select the database to delete and click **Finish**.

   After you click **Finish**, DBCA displays a dialog to confirm the database and instances that DBCA is going to delete.

6. Click **OK** to begin the deletion of the database and its associated files, services, and environment settings, or click **Cancel** to stop the operation.

When you click **OK**, DBCA continues the operation and deletes all of the associated instances for this database. DBCA also removes the parameter files, password files, and oratab entries.

At this point, you have accomplished the following:

- Deleted the selected database from the cluster
- Deleted the selected database's Oracle services for Windows-based platforms
- Deleted high availability services that were assigned to the database
- Deleted the Oracle Net configuration for the database
- Deconfigures Database Control
- Deleted the OFA directory structure from the cluster
- Deleted the data files if the data files were not on raw devices

7

Configuring the Server Parameter File in Oracle Real Application Clusters Environments

This chapter describes server parameter file (SPFILE) placement and configuration in Real Application Clusters (RAC) environments. The topics in this chapter are:

- Parameter Files and Real Application Clusters
- Using Server Parameter Files in Real Application Clusters
- Parameter File Search Order in Real Application Clusters
- Migrating to the Server Parameter File in Real Application Clusters Environments
- Server Parameter File Errors in Real Application Clusters

> **See Also:** The *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about parameters and the *Oracle Real Application Clusters Deployment and Performance Guide* for a discussion of parallel execution-related parameters in RAC data warehouse environments

Parameter Files and Real Application Clusters

Oracle uses parameter settings in parameter files to determine how to control various database resources. You can use two types of files for parameter administration: the server parameter file (SPFILE) or one or more traditional client-side parameter files.

Oracle recommends that you manage parameters using an SPFILE. If you use client-side parameter files, then Oracle does not preserve parameter changes made for self-tuning after shutdown.

> **See Also:** Oracle Database 10*g* Real Application Clusters documentation for more information about using client-side parameter files

Using Server Parameter Files in Real Application Clusters

By default, Oracle creates the server parameter file based on one `SPFILE`. You can change parameter settings in the server parameter file only by using Oracle Enterprise Manager or `ALTER SYSTEM SET` SQL statements; the server parameter file is a binary file that you should not edit.

Note: Oracle recommends that you avoid modifying the values for self-tuning parameters; overriding these settings can adversely affect performance.

If you are upgrading from a previous Oracle release, then create and configure the server parameter file for RAC using the procedures described in the following section.

Location of The Server Parameter File

The default location of the server parameter file (SPFILE) when the database creates it from a `PFILE` is platform-dependent.

The default location of the server parameter file is:

`$ORACLE_HOME/dbs/spfile$`*`ORACLE_sid`*`.ora`

The default location of the server parameter file is inappropriate for RAC databases and ASM instances if you use raw devices, because all instances must use the same server parameter file.

Instead, Oracle recommends that you use a `PFILE` in this directory:

`$ORACLE_HOME/dbs/init`*`$ORACLE_sid`*`.ora`

This path is valid for each instance and it refers to a single, shared initialization parameter file. If you use raw storage, then the file must contain the following entry:

`SPFILE='/dev/vx/rdsk/oracle_dg/`*`db`*`spfile'`

However, if you use a cluster file system, then use one of the following file location:

```
SPFILE='$ORACLE_HOME/dbs/spfile.ora'
```

If you use ASM, then the SPFILE value will be:

```
SPFILE='+disk_group_name/dbunique_name/spfiledbname.ora'
```

where *`dbunique_name`* is the unique database name and *`dbname`* is the database name.

You must use the same value of `SPFILE` so that all instances use the same server parameter file at startup.

To use DBCA to create your database and to use the server parameter file, on the Initialization Parameters page select Create server parameter file (SPFILE) under the File Locations tab which is visible only if you are using raw storage. Then enter either a shared file system filename or the raw device path name in the `Server Parameters Filename` field.

Note: When you use DBCA to create the server parameter file, the default `PFILE` file name is `$ORACLE_HOME/dbs/init$ORACLE_`*`sid`*`.ora`.

Parameter File Search Order in Real Application Clusters

Oracle searches for your parameter file in the following order:

1. `$ORACLE_HOME/dbs/spfile`*`sid`*`.ora`

2. `$ORACLE_HOME/dbs/spfile.ora`
3. `$ORACLE_HOME/dbs/init*sid*.ora`

Migrating to the Server Parameter File in Real Application Clusters Environments

Migrate to the server parameter file by creating and editing the server parameter file using the procedures described in this section.

Server Parameter File Placement in Real Application Clusters

For single-node cluster-enabled configurations, or if you are using ASM disk groups or a cluster file system, place the server parameter file in an ASM disk group or on a file system. Otherwise, place the server parameter file on a shared raw device that is at least 5MB in size.

Procedures for Migrating to the Server Parameter File

Migrate to the server parameter file (SPFILE) file by completing the following procedures:

1. Combine the initialization parameter files for all instances into one `init*dbname*.ora` file by copying all shared IFILE contents *as is*. All parameters defined in your IFILE parameter files are global. Therefore, create them as "parameter=value" without system identifier (sid) prefixes.
2. Copy all instance-specific parameter definitions from `init*sid*.ora` files using the following syntax, where the variable *sid* is the system identifier (SID) of the instance:

   ```
   sid.parameter=value
   ```

3. If you are using a cluster file system, then create the server parameter file using the `CREATE SPFILE` statement as in the following example:

   ```
   CREATE SPFILE='?/dbs/spfile_dbname.ora'
   FROM PFILE='?/dbs/initdbname.ora'
   ```

   If you use ASM, then use the following syntax to create the server parameter file:

   ```
   CREATE SPFILE='+disk_group_name/db_uniquename/spfiledbname.ora'
   FROM PFILE='?/dbs/initdbname.ora'
   ```

   If you use raw storage, then use the following syntax to create the server parameter file on a raw device:

   ```
   CREATE SPFILE='/dev/vx/rdsk/oracle_dg/dbspfile'
   FROM PFILE='?/dbs/initdbname.ora'
   ```

   These statements read your combined `init*dbname*.ora` file that you created by merging your IFILEs. Then it transfers the settings for the parameters from the merged file into your server parameter file.

4. Oracle recommends that you use the SPFILE by using the PFILE as a pointer to the SPFILE. Do this by running the `STARTUP` command as in the following example, where the SID variable *sid* in the file `init*sid*.ora` is the SID that you use in the SPFILE entry from Step 3. For example:

   ```
   STARTUP PFILE=$ORACLE_HOME/dbs/initsid.ora
   ```

If you use this `STARTUP` command syntax, then Oracle uses the server parameter file entry specified in `init`*`sid`*`.ora`.

Note: the SPFILE for ASM instance cannot be stored in ASM, as it is used to start the ASM instance. You can place an ASM instance SPFILE on a shared raw device or in a cluster file system for ASM, or you can have a traditional client-side parameter file for each ASM instance.

Server Parameter File Errors in Real Application Clusters

Oracle reports errors that occur during server parameter file creation or while reading the file during startup. If an error occurs during a parameter update, then Oracle records the error in your `ALERT.LOG` file and ignores subsequent parameter updates to the file. If this happens, then do either of the following:

- Shut down the instance, recover the server parameter file, and restart the instance.
- Enable the instance to continue running without regard for subsequent parameter updates.

Oracle displays errors for parameter changes that you attempt when you incorrectly use the `ALTER SYSTEM SET` statement. Oracle does this when an error occurs while reading from or writing to the server parameter file.

> **See Also:** The *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about backing up the SPFILE

8

Oracle Real Application Clusters Post-Installation Procedures

This chapter describes how to complete the post-installation tasks after you have installed the Oracle Database 10*g* with Real Application Clusters (RAC) software. It contains the following sections:

- Required Post-Installation Tasks
- Recommended Post-Installation Tasks

> **Note:** This chapter only describes basic configurations. Refer to the *Oracle Database 10g Administrator's Guide for UNIX Systems*, *Oracle Database Platform Guide for Microsoft Windows (32-Bit)*, and the product administration and tuning guides for more detailed configuration and tuning information.

Required Post-Installation Tasks

You must perform the following tasks after completing your installation:

- Back Up the Voting Disk After Installation
- Download and Install Patches
- Configure Oracle Products
- Oracle Real Application Clusters 10g Installed on an OCFS

Back Up the Voting Disk After Installation

After your Oracle Database 10*g* with RAC installation is complete and after you are sure that your system is functioning properly, make a backup of the contents of the voting disk. use the `dd.exe` utility.

Also make a backup of the voting disk contents after you complete any node additions or node deletions and after running any deinstallation procedures.

Download and Install Patches

Refer to the Oracle*MetaLink* Web site for required patches for your installation. To download required patches:

1. Use a Web browser to view the Oracle*MetaLink* Web site:

   `http://metalink.oracle.com`

2. Log in to Oracle*MetaLink*.

   Note: If you are not an Oracle*MetaLink* registered user, then click **Register for MetaLink** and register.

3. On the main Oracle*MetaLink* page click **Patches**.
4. On the Select a Patch Search Area page click **New MetaLink Patch Search**.
5. On the Simple Search page click **Advanced**.
6. On the Advanced Search page click the search icon next to the Product or Product Family field.
7. In the Search and Select: Product Family field, enter RDBMS Server in the For field and click **Go**.
8. Select RDBMS Server under the Results heading and click **Select**.

   RDBMS Server appears in the Product or Product Family field and the current release appears in the Release field.
9. Select your platform from the list in the Platform field and click **Go**.
10. Any available patches appear under the Results heading.
11. Click the number of the patch that you want to download.
12. On the Patch Set page, click **View README** and read the page that appears. The README page contains information about the patch set and how to apply the patches to your installation.
13. Return to the Patch Set page, click **Download**, and save the file on your system.
14. Use the unzip utility provided with Oracle Database 10*g* to uncompress the Oracle patches that you downloaded from Oracle*MetaLink*. the unzip utility is located in the `$ORACLE_HOME/bin` directory.

Configure Oracle Products

Many Oracle products and options must be configured before you use them for the first time. Before using individual Oracle Database 10*g* database products or options, refer to the manual in the product documentation library which is available in the DOC directory on the 10*g* Release 2 (10.2) installation media, or on the OTN Web site.

Oracle Real Application Clusters 10*g* Installed on an OCFS

If you installed RAC on an OCFS, then perform the following steps on each node of the cluster after installation:

1. Stop the Oracle instance.
2. Move the `$ORACLE_HOME/dbs/hc_*.dat` files to a directory on a local file system.
3. Create symbolic links from the `$ORACLE_HOME/dbs` directory to the `hc_*.dat` files on the local file system.
4. Restart the Oracle instance.

Verify Required Post-Installation is Completed

Use the following commands to verify that all required post-installation steps are completed

Recommended Post-Installation Tasks

This section explains the tasks that Oracle recommends you perform after completing an installation. There is one task recommended on all systems and additional tasks that apply only to UNIX-based or Windows-based systems (but not both), as follows:

- Verifying Enterprise Manager Operations
- Recommended Post-Installation Tasks

Verifying Enterprise Manager Operations

Run the following command to verify the Enterprise Manager configuration in your newly installed Real Application Clusters environment:

```
srvctl config database -d db_name
```

SRVCTL displays the name of the node and the instance for the node. The following example shows a node named `db1-server` running an instance named `db1`. Run the following command:

```
srvctl config database -d db
```

The output should be similar to:

```
db1-server db1 /private/system/db
db2-server db2 /private/system/db
```

Continue with the following section, "Recommended Post-Installation Tasks". When you have completed these asks, you should proceed with the initial configuration tasks described in Part IV.

Recommended Post-Installation Tasks

Oracle recommends that you complete the following tasks after installing Oracle RAC:

- Back Up the root.sh Script
- Set Up Users Accounts
- Set the ORACLE_HOME and ORACLE_SID Environments

Back Up the root.sh Script

Oracle recommends that you back up the `root.sh` script after you complete an installation. If you install other products in the same Oracle home directory, then the Oracle Universal Installer (OUI) updates the contents of the existing `root.sh` script during the installation. If you require information contained in the original `root.sh` script, then you can recover it from the `root.sh` file copy.

Set Up Users Accounts

For information about setting up additional optional user accounts, refer to the *Oracle Database 10g Administrator's Guide for UNIX Systems*.

When you complete these procedures, you are ready to perform the initial configuration tasks described in Part IV.

Set the ORACLE_HOME and ORACLE_SID Environments

To ensure that you can use Oracle tools such as SQL*Plus to connect to the database, Oracle recommends that you set the ORACLE_HOME and ORACLE_SID environment variables to the correct Oracle home and Oracle SID that you configured during installation.

Part IV

Oracle Real Application Clusters Environment Configuration

Part IV describes how to use the server parameter file (SPFILE) in Oracle Database 10*g* Real Application Clusters (RAC) and it describes the installed configuration. The chapters in Part IV are:

- Chapter 7, "Configuring the Server Parameter File in Oracle Real Application Clusters Environments"
- Chapter 9, "Understanding the Oracle Real Application Clusters Installed Configuration"

9

Understanding the Oracle Real Application Clusters Installed Configuration

This chapter describes the Oracle Real Application Clusters (RAC) installed configuration. The topics in this chapter include:

- Understanding the Configured Environment in Real Application Clusters
- Oracle Cluster Registry in Oracle Real Application Clusters
- Oratab Configurations for Oracle Real Application Clusters
- Database Components Created Using Database Configuration Assistant
- Managing Undo Tablespaces in Real Application Clusters
- Initialization Parameter Files
- Configuring Service Registration-Related Parameters in Real Application Clusters
- Configuring the Listener File (listener.ora)
- Directory Server Access (ldap.ora File)
- Net Service Names (tnsnames.ora File)
- Net Services Profile (sqlnet.ora File)

Understanding the Configured Environment in Real Application Clusters

The Oracle Net Configuration Assistant (NetCA) and Database Configuration Assistant (DBCA) configure your environment to meet the requirements for database creation and Enterprise Manager discovery of Real Application Cluster databases.

Note: Configuration files are created on each node in your cluster database.

Oracle Cluster Registry in Oracle Real Application Clusters

DBCA uses Oracle Cluster Registry (OCR) for storing the configurations for the cluster databases that it creates. The OCR is a shared file in a cluster file system environment. If you do not use a cluster file system, then you must make this file a shared raw device. The Oracle Universal Installer (OUI) automatically initializes the OCR during the Oracle Clusterware installation.

Oratab Configurations for Oracle Real Application Clusters

Oracle creates an entry for each RAC database in the `oratab` configuration file. Oracle Enterprise Manager uses this file during service discovery to determine the name of the RAC database, and to determine whether the database should be automatically started when the system is restarted. The database entry has the following syntax:

```
db_unique_name:$ORACLE_HOME:N
```

where *db_unique_name* is the database name for your RAC database, `$ORACLE_HOME` is the directory path to the database, and `N` indicates that the database should not be started at restart time. A sample entry for a database named `db` is:

```
db:/private/system/db:N
```

Note: Where the notation *db_name* appears in the previous example and throughout this chapter, it refers to the database name you entered when prompted by DBCA, or it refers to the entry you made for the `DATABASE` keyword of the `CREATE DATABASE` statement.

Database Components Created Using Database Configuration Assistant

This section describes the database components that DBCA creates, which include:

- Tablespaces and Data files
- Control Files
- Redo Log Files

Tablespaces and Data files

For both single-instance and cluster database environments, an Oracle Database is divided into smaller logical areas of space known as tablespaces. Each tablespace corresponds to one or more data files stored on a disk. Table 9–1 shows the tablespace names used by a RAC database and the types of data they contain:

Table 9–1 Tablespace Names that Real Application Clusters Databases Use

Tablespace Name	Contents
`SYSTEM`	Consists of the data dictionary, including definitions of tables, views, and stored procedures needed by the database. Oracle automatically maintains information in this tablespace.
`SYSAUX`	An auxiliary system tablespace that contains the `DRSYS` (contains data for OracleText), `CWMLITE` (contains the OLAP schemas), `XDB` (for XML features), `ODM` (for Oracle Data Mining), `TOOLS` (contains Enterprise Manager tables), `INDEX`, `EXAMPLE`, and `OEM-REPO` tablespaces.
`USERS`	Consists of application data. As you create and enter data into tables, Oracle fills this space with your data.

Table 9–1 (Cont.) Tablespace Names that Real Application Clusters Databases Use

Tablespace Name	Contents
TEMP	Contains temporary tables and indexes created during SQL statement processing. You may need to expand this tablespace if you are running a SQL statement that involves significant sorting, such as ANALYZE COMPUTE STATISTICS on a very large table, or the constructs GROUP BY, ORDER BY, or DISTINCT.
UNDOTBS*n*	These are the undo tablespaces for each instance that DBCA creates for automatic undo management.
RBS	If you do not use automatic undo management, then Oracle uses the RBS tablespace for the rollback segments.

You cannot alter these tablespace names when using the preconfigured database configuration options from the Oracle Universal Installer. However, you can change the names of the tablespaces if you use the advanced database creation method.

As mentioned, each tablespace has one or more data files. The data file names created by the preconfigured database configuration options vary by storage type such as ASM, OFS, raw devices, and so on.

You can specify different symbolic names with the Advanced database configuration option.

Control Files

The database is configured with two control files that are stored on shared storage.

Redo Log Files

Each instance is configured with at least two redo log files that are stored in the shared storage. If you chose cluster file system, then these files are shared file system files. If you do not have a cluster file system, then these files are raw devices. If you use ASM, then these files are stored on the ASM disk group.

The file names of the redo log files that are created with the preconfigured database configuration options vary by storage type. You must enter the raw device names unless you are using a cluster file system.

When using raw devices, to use the advanced database creation method, locate the redo log files in the Database Storage page and replace their default filenames with the correct raw device names or symbolic link names.

Managing Undo Tablespaces in Real Application Clusters

Oracle stores rollback or undo information in undo tablespaces. To manage undo tablespaces, Oracle recommends that you use automatic undo management. Automatic undo management is an automated undo tablespace management mode that is easier to administer than manual undo management.

> **See Also:** *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about managing undo tablespaces

Initialization Parameter Files

Oracle recommends using the server parameter file (SPFILE). This file resides on the server on the shared disk; all instances in a cluster database can access this parameter file.

> **See Also:** Chapter 7, "Configuring the Server Parameter File in Oracle Real Application Clusters Environments" for more information about the creation and use of parameter files

Configuring Service Registration-Related Parameters in Real Application Clusters

Two key benefits of RAC are connection load balancing and failover. RAC extends the ability of single-instance Oracle database load balancing, where connections are distributed among local dispatchers, to the balancing of connections among all instances in a cluster database. In addition, RAC provides failover by configuring multiple listeners on multiple nodes to manage client connection requests for the same database service. Connection load balancing and failover increase availability by taking advantage of the redundant resources within a cluster database. These features, however, require cross instance registration.

Cross instance registration in RAC occurs when an instance's PMON process registers with the local listener and with all other listeners. Thus, all instances in the cluster database register with all listeners that run on nodes that run instances of the cluster database. This enables all listeners to manage connections across all instances for both load balancing and failover.

Cross instance registration requires configuring the `LOCAL_LISTENER` and `REMOTE_LISTENER` initialization parameters. The `LOCAL_LISTENER` parameter identifies the local listener and the `REMOTE_LISTENER` parameter identifies the global list of listeners. The `REMOTE_LISTENER` parameter is dynamic. Oracle changes the setting for `REMOTE_LISTENER` dynamically when you reconfigure your cluster database, for example, when you add or delete instances.

By default, DBCA configures your environment with dedicated servers. However, if you select the Shared server option on DBCA, then Oracle configures the shared server. In this case, Oracle uses both dedicated and shared server processing. When shared servers are configured, the `DISPATCHERS` parameter is specified as in the following example:

```
DISPATCHERS="(protocol=tcp)"
```

If the `DISPATCHERS` initialization parameter does not specify the `LISTENER` attribute as in the previous example, then the PMON process registers information for all dispatchers with the listeners specified by the `LOCAL_LISTENER` and `REMOTE_LISTENER` parameters.

However, when the `LISTENER` attribute is specified, the PMON process registers dispatcher information with the listeners specified by the `LISTENER` attribute. In this case, setting the `LISTENER` attribute overrides `REMOTE_LISTENER` settings for the specified dispatchers as in the following example:

```
DISPATCHERS="(protocol=tcp)(listener=listeners_db_name)"
```

See Also: *Oracle Database Net Services Administrator's Guide* for further information about cross instance registration, shared and dedicated server configurations, and connection load balancing

Configuring the Listener File (listener.ora)

You can configure two types of listeners in the `listener.ora` file as described under the following headings:

- Local Listeners
- Multiple Listeners
- How Oracle Database Uses the Listener (listener.ora File)

Local Listeners

If you configured dedicated server mode using the DBCA Connection Mode tab on the Initialization Parameters page, then DBCA automatically configures the `LOCAL_LISTENER` parameter when the listener uses a nondefault address port.

If you configured the dedicated server mode by setting the `REMOTE_LISTENER` initialization parameter, then you must also configure the instance-specific `LOCAL_LISTENER` initialization parameter.

For example, to configure the `LOCAL_LISTENER` parameter, add the following entry to the initialization parameter file, where `listener_`*`sid`* is resolved to a listener address through either a `tnsnames.ora` file or through the Oracle Names Server:

```
sid.local_listener=listener_sid
```

The following entry should be in your `tnsnames.ora` file:

```
listener_sid=(address=(protocol=tcp)(host=node1-vip)(port=1522))
```

Multiple Listeners

If DBCA detects more than one listener on the node, it displays a list of the listeners. You can select one or all of these listeners with which to register your database.

How Oracle Database Uses the Listener (listener.ora File)

Services coordinate their sessions using listener file entries by running a process on the server that receives connection requests on behalf of a client application. Listeners are configured to respond to connection requests sent to protocol addresses for a database service or non-database service.

Protocol addresses are configured in the listener configuration file, `listener.ora`, for a database service or a non-database service. Clients configured with the same addresses can connect to a service through the listener.

During a preconfigured database configuration installation, the Oracle Net Configuration Assistant creates and starts a default listener called `LISTENER_`*`NODENAME`*. The listener is configured with a default protocol listening addresses for the database and external procedures. The advanced installation process prompts you to create at least one listener with the Oracle Net Configuration Assistant. The listener is configured to respond to connection requests that are directed at one protocol address you specify, as well as an address for external procedures.

Both installation modes configure service information about the RAC database and external procedures. An Oracle Database 10*g* Release 2 (10.2) database service automatically registers its information with the listener, such as its service name, instance names, and load information.

This feature, called service registration, does not require configuration in the `listener.ora` file. After listener creation, the Oracle Net Configuration Assistant starts the listener. A sample `listener.ora` file with an entry for an instance named `node1` is:

```
listener_node1=
  (description=
    (address=(protocol=ipc)(key=extproc))
    (address=(protocol=tcp)(host=node1-vip)(port=1521))
    (address=(protocol=tcp)(host=node1-ip)(port=1521)))
sid_list_listener_node1=
  (sid_list=
    (sid_desc=
      (sid_name=plsextproc)
      (oracle_home=/private/system/db)
      (program=extproc)
```

Listener Registration and PMON Discovery

When a listener starts after the Oracle instance starts, and the listener is listed for service registration, registration does not occur until the next time the PMON discovery routine starts. By default, PMON discovery occurs every 60 seconds.

To override the 60-second delay, use the SQL ALTER SYSTEM REGISTER statement. This statement forces PMON to register the service immediately.

Oracle recommends that you create a script to run this statement immediately after starting the listener. If you run this statement while the listener is up and the instance is already registered, or while the listener is down, then the statement has no effect.

> **See Also:** *Oracle Database Net Services Administrator's Guide* for further information about the listener and the `listener.ora` file

Directory Server Access (ldap.ora File)

If you configure access to a Lightweight Directory Access Protocol (LDAP)-compliant directory server with the Oracle Net Configuration Assistant during a Custom Install or Advanced database configuration, an `ldap.ora` file is created. The `ldap.ora` file contains the following types of information:

- Type of directory
- Location of the directory
- Administrative context from which this server can look up, create, and modify a net service name, and the database service entries

> **See Also:** *Oracle Database Net Services Administrator's Guide* for further information about directory naming configuration and directory server access configuration

Net Service Names (tnsnames.ora File)

A `tnsnames.ora` file is created on each node with net service names. A connect identifier is an identifier that maps to a connect descriptor. A connect descriptor contains the following information:

- The network route to the service, including the location of the listener through a protocol address
- The `SERVICE_NAME` for an Oracle release 8.1 or later, or `sid` for pre-8.1 Oracle releases

> **Note:** The `SERVICE_NAME` parameter you use in `tnsnames.ora` is singular because you can only specify one service name.

DBCA creates net service names for connections as shown in Table 9–2:

Table 9–2 Connections for Net Service Names

Net Service Name Type	Description

Database connections	Clients that connect to any instance of the database use the net service name entry for the database. This entry also enables Oracle Enterprise Manager to discover a RAC database.
	
	A listener address is configured for each node that runs an instance of the database. The `LOAD_BALANCE` option causes Oracle to choose the address randomly. If the chosen address fails, then the `FAILOVER` option causes the connection request to fail over to the next address. Thus, if an instance fails, then clients can still connect using another instance.
	
	In the following example, `db.us.oracle.com` is used by the client to connect to the target database, `db.us.oracle.com`.
	
	`db.us.acme.com=`
	`(description=`
	`(load_balance=on)`
	`(address=(protocol=tcp)(host=node1-vip)(port=1521)`
	`(address=(protocol=tcp)(host=node2-vip)(port=1521)`
	`(connect_data=`
	`(service_name=db.us.acme.com)))`
	
	Note: `FAILOVER=ON` is set by default for a list of addresses. Thus, you do not need to explicitly specify the `FAILOVER=ON` parameter.
	
	When you set `DB_UNIQUE_NAME` by entering a global database name that is longer than eight characters, excluding `DB_DOMAIN`, then a net service entry similar to the following is created:
	
	`mydatabase.us.acme.com=`
	`(description =`
	`(address = (protocol = tcp)(host = node1-vip)(port = 1521))`
	`(address = (protocol = tcp)(host = node2-vip)(port = 1521))`
	`(load_balance = yes)`
	`(connect_data =`
	`(server = dedicated)`
	`(service_name = mydatabase.us.acme.com)`
	`)`
	`)`

Table 9–2 (Cont.) Connections for Net Service Names

Net Service Name Type	Description

Instance connections	Clients that connect to a particular instance of the database use the net service name entry for the instance. This entry, for example, enables Oracle Enterprise Manager to discover the instances in the cluster. These entries are also used to start and stop instances.
	
	In the following example, `db1.us.acme.com`, is used by Oracle Enterprise Manager to connect to an instance named `db1` on `db1-server`:
	
	`db1.us.acme.com=`
	`(description=`
	`(address=(protocol=tcp)(host=node1-vip)(port=1521))`
	`(connect_data=`
	`(service_name=db.us.acme.com)`
	`(instance_name=db1)))`

Remote listeners	As discussed in "Configuring Service Registration-Related Parameters in Real Application Clusters" on page 9-4, the `REMOTE_LISTENER` parameter identifies the global list of listeners and it is dynamic. Oracle changes the setting for `REMOTE_LISTENER` when you reconfigure your cluster database.
	
	Whether using shared servers or dedicated servers, the list of remote listeners is supplied using the `REMOTE_LISTENERS` parameter, for example:
	
	`REMOTE_LISTENERS=listeners_`*`db_unique_name`*
	
	This enables the instance to register with remote listeners on the other nodes; `listeners_`*`db_unique_name`* is resolved through a naming method such as a `tnsnames.ora` file.
	
	In the following example, `listeners_db.us.acme.com` is resolved to a list of listeners available on the nodes on which the cluster database has instances:
	
	`listeners_db.us.acme.com=`
	`(address_list=`
	`(address=(protocol=tcp)(host=node1-vip)(port=1521))`
	`(address=(protocol=tcp)(host=node2-vip)(port=1521)))`
	
	The instance uses this list to determine the addresses of the remote listeners with which to register its information.

Nondefault listeners	As discussed in "Local Listeners" on page 9-5 and "Multiple Listeners" on page 9-5, the `LOCAL_LISTENER` parameter is set in the `init`*`sid`*`.ora` file if a nondefault listener is configured, for example:
	
	`sid``.local_listener=listener_`*`sid`*
	
	Where `listener_`*`sid`* is resolved to a listener address through a naming method such as a `tnsnames.ora` file.
	
	In the following sample, `listener_db1.us.acme.com` is resolved to the nondefault listener address:
	
	`listener_db1.us.acme.com=`
	`(address=(protocol=tcp)(host=node1-vip)(port=1522))`

Table 9–2 (Cont.) Connections for Net Service Names

Net Service Name Type	Description
Services Entries	When you configure high availability services using the DBCA Services page, then DBCA creates net service entries similar to the following. The three services in the following examples, db_svc1, db_svc2, and db_svc3, have TAF policies of NONE, BASIC and PRECONNECT respectively.

```
db_svc1.us.acme.com=
  (description =
    (address=(protocol=tcp)(host=node1-vip)(port=1521))
    (address=(protocol=tcp)(host=node2-vip)(port=1521))
    (load_balance=yes)
    (connect_data=
        (server = dedicated)
        (service_name = db_svc1.us.acme.com)
  )
  )

db_svc2.us.acme.com=
  (description=
    (address=(protocol=tcp)(host=node1-vip)(port=1521))
    (address=(protocol=tcp)(host=node2-vip)(port=1521))
    (load_balance=yes)
    (connect_data =
        (server = dedicated)
        (service_name=db_svc2.us.acme.com)
        (failover_mode =
        (type=select)
        (method=basic)
           (retries=180)
           (delay=5)
     )
     )
     )

db_svc3.us.acme.com=
    (description=
      (address=(protocol=tcp)(host=node1-vip)(port=1521))
      (address=(protocol=tcp)(host=node2-vip)(port=1521))
      (load_balance=yes)
      (connect_data=
        (server=dedicated)
        (service_name=db_svc3.us.acme.com)
        (failover_mode=
        (backup=db_svc3_preconnect.us.acme.com)
        (type=select)
        (method=preconnect)
        (retries=180)
        (delay=5)
   )
   )
   )
```

Table 9–2 (Cont.) Connections for Net Service Names

Net Service Name Type	Description

Services Entries (continued)	When a service has a TAF policy of `PRECONNECT`, then a *`service_name`*`_preconnect net service` entry is also created as in the following example:
	`db_svc3_preconnect.us.acme.com =`
	`  (description =`
	`  (address = (protocol = tcp)(host = node1-vip)(port = 1521))`
	`  (address = (protocol = tcp)(host = node2-vip)(port = 1521))`
	`  (load_balance = yes)`
	`  (connect_data =`
	`  (server = dedicated)`
	`  (service_name = db_svc3_preconnect.us.amce.com)`
	`  (failover_mode =`
	`  (backup = db_svc3.us.acme.com)`
	`  (type = select)`
	`  (method = basic)`
	`  (retries = 180)`
	`  (delay = 5)`
	`  )`
	`  )`
	`  )`

External procedures	An entry for connections to external procedures. This enables an Oracle Database 10*g* database to connect to external procedures.
	`extproc_connection_data.us.acme.com=`
	`  (description=`
	`  (address_list=`
	`  (address=(protocol=ipc)(key=extproc0))`
	`  (connect_data=`
	`  (sid=plsextproc)))`

Example 9–1 Example tnsnames.ora File

The following is a sample `tnsnames.ora` file that is created during a preconfigured database configuration installation:

```
db.us.acme.com=
 (description=
  (load_balance=on)
   (address=(protocol=tcp)(host=node1-vip)(port=1521))
   (address=(protocol=tcp)(host=node2-vip)(port=1521))
  (connect_data=
     (service_name=db.us.acme.com)))

db1.us.acme.com=
 (description=
  (address=(protocol=tcp)(host=node1-vip)(port=1521))
  (connect_data=
    (service_name=db.us.acme.com)
    (instance_name=db1)))

db2.us.acme.com=
 (description=
  (address=(protocol=tcp)(host=node2-vip)(port=1521))
  (connect_data=
    (service_name=db.us.acme.com)
    (instance_name=db2)))

listeners_db.us.acme.com=
(address_list=
```

```
   (address=(protocol=tcp)(host=node1-vip)(port=1521))
   (address=(protocol=tcp)(host=node2-vip)(port=1521)))

extproc_connection_data.us.acme.com=
 (description=
  (address_list=
    (address=(protocol=ipc)(key=extproc)))
  (connect_data=
    (sid=plsextproc)
    (presentation=RO)))
```

> **See Also:** *Oracle Database Net Services Administrator's Guide* for further information about the `tnsnames.ora` file

Net Services Profile (sqlnet.ora File)

Oracle Universal Installer starts Oracle Net Configuration Assistant after the database is installed, and creates the Net Services profile, or `sqlnet.ora` file.

By default, the `sqlnet.ora` file is located in the following directory:

```
$ORACLE_HOME/network/admin
```

In this directory, in addition to the default `sqlnet.ora` file generated during installation, you also can find a sample sqlnet.ora file in the directory `sample`.

During installation, Oracle Net Configuration Assistant creates the following entries in the `sqlnet.ora` file:

```
NAMES.DIRECTORY_PATH=(TNSNAMES, EZCONNECT)
```

The parameter NAMES.DIRECTORY_PATH specifies the priority order of the naming methods to use to resolve connect identifiers to connect descriptors.

The order of naming methods is as follows: directory naming (for Custom Install or Advanced database configuration options only), `tnsnames.ora` file, Easy Connect, and host naming.

The Easy Connect naming method eliminates the need for service name lookup in the tnsnames.ora files for TCP/IP environments. With Easy Connect, clients use a connect string for a simple TCP/IP address, consisting of a host name and optional port and service name. If you use this method, then no naming or directory system is required.

> **See Also:** *Oracle Database Net Services Administrator's Guide* for further information about the `sqlnet.ora` file

Part V

Oracle Clusterware and Oracle Real Application Clusters Installation and Configuration Reference Information

Part V provides Oracle Clusterware and Oracle Real Application Clusters (RAC) installation and configuration reference information. The contents of Part V are:

- Appendix A, "Troubleshooting the Oracle Clusterware and Oracle Real Application Clusters Installation Process"
- Appendix B, "Using Scripts to Create Oracle Real Application Clusters Databases"
- Appendix C, "Configuring Raw Devices for Oracle Real Application Clusters"
- Appendix D, "Converting to Oracle Real Application Clusters from Single-Instance Oracle Databases"
- Appendix E, "Directory Structure for Oracle Database 10g Oracle Real Application Clusters Environments"

A

Troubleshooting the Oracle Clusterware and Oracle Real Application Clusters Installation Process

This appendix provides troubleshooting information for installing Oracle Clusterware and Oracle Database 10*g* Real Application Clusters (RAC). The topic in this appendix is:

- Troubleshooting the Oracle Real Application Clusters Installation

> **See Also:** The Oracle Database 10*g* Oracle Real Application Clusters documentation set included with the installation media in the Documentation directory:
>
> - *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*

Troubleshooting the Oracle Real Application Clusters Installation

This section contains these topics:

- General Installation Issues
- Real Application Clusters Installation Error Messages
- Performing Cluster Diagnostics During Real Application Clusters Installations

General Installation Issues

The following is a list of examples of types of errors that can occur during installation

An error occurred while trying to get the disks

Cause: There is an entry in /etc/oratab pointing to a non-existent Oracle home. The OUI error file should show the following error: "java.io.IOException: /home/oracle/OraHome//bin/kfod: not found" (Oracle*Metalink* bulletin 276454.1)

Action: Remove the entry in `/etc/oratab` pointing to a non-existing Oracle home.

Nodes unavailable for selection from the OUI Node Selection screen

Cause: Oracle Clusterware is either not installed, or the Oracle Clusterware services are not up and running.

Action: Install Oracle Clusterware, or review the status of your Oracle Clusterware. Consider restarting the nodes, as doing so may resolve the problem.

Node *nodename* is unreachable

Cause: Unavailable IP host

Action: Attempt the following:

1. Run the shell command `ifconfig -a`. Compare the output of this command with the contents of the /etc/hosts file to ensure that the node IP is listed.
2. Run the shell command `nslookup` to see if the host is reachable.
3. As Oracle, attempt to connect to the node with ssh or rsh. If you are prompted for a password, then user equivalence is not set up properly. Review the section "Configuring SSH on All Cluster Nodes" on page 2-11.

Time stamp is in the future

Cause: One or more nodes has a different clock time than the local node. If this is the case, then you may see output similar to the following:

```
time stamp 2005-04-04 14:49:49 is 106 s in the future
```

Action: Ensure that all member nodes of the cluster have the same clock time.

YPBINDPROC_DOMAIN: Domain not bound

Cause: This error has been seen during post-installation testing when a node public network interconnect is pulled out, and the VIP does not fail over. Instead, the node hangs, and users are unable to log in to the system. This error occurs when the Oracle home, listener.ora, Oracle log files, or any action scripts are located on an NAS device or NFS mount, and the name service cache daemon `nscd` has not been activated.

Action: Enter the following command on all nodes in the cluster to start the nscd service:

```
/sbin/service  nscd start
```

Real Application Clusters Installation Error Messages

Oracle Real Application Clusters Management Tools Error Messages are in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.

Performing Cluster Diagnostics During Real Application Clusters Installations

If the Oracle Universal Installer (OUI) does not display the Node Selection page, then perform clusterware diagnostics by running the `olsnodes -v` command from the binary directory in your Oracle Clusterware home (*`CRS_home`*`/bin`) and analyzing its output. Refer to your clusterware documentation if the detailed output indicates that your clusterware is not running.

In addition, use the following command syntax to check the integrity of the Cluster Manager:

```
cluvfy comp clumgr -n node_list -verbose
```

In the preceding syntax example, the variable *`node_list`* is the list of nodes in your cluster, separated by commas.

B

Using Scripts to Create Oracle Real Application Clusters Databases

This chapter describes the steps required to create an Oracle Real Application Clusters (RAC) database using scripts to perform noninteractive installations. The topic in this appendix is:

- Creating a Database Using Scripts

Note: The scripts generated by DBCA are for reference purposes only. Oracle strongly recommends that you use DBCA to create a database.

See Also: *Oracle Universal Installer and OPatch User's Guide* for detailed information about generating scripts

Creating a Database Using Scripts

To generate scripts to create a Real Application Clusters database, create a database using the scripts, and prepare the database for use, complete the following steps:

1. Start Database Configuration Assistant (DBCA) and select your preferred options to build the RAC database. Note: you must select the Custom Database template on the Database Templates page for DBCA to provide the script generation option.

   On the Creation Options page of your DBCA session, deselect Create Database and select Generate Database Creation Scripts before you click **Finish**. You can accept the default destination directory for the scripts, or browse for a different location. In either case, you should note the path name for use in the next step.

   See Also: Chapter 6, "Creating Real Application Clusters Databases with Database Configuration Assistant" for additional information about running a DBCA session.

2. Navigate to the directory, which you noted in Step 1, where DBCA created the scripts, and review the SQL scripts to ensure that they contain the statements to build a database with the characteristics you require. If they do not, then Oracle recommends that you rerun DBCA to create scripts with the desired configuration rather than editing the scripts yourself.
3. On each cluster node you identified during your DBCA session, run the script sid`.sh`, where *`sid`* is the SID prefix that you entered on the DBCA Database Name page.

4. Set the initialization parameter, `cluster_database`, to the value `TRUE` in your SPFILE by issuing an `ALTER SYSTEM` command, or by uncommenting it in your PFILE for each instance.
5. Configure Net Services to support your new database and instances as described in Chapter 9, "Understanding the Oracle Real Application Clusters Installed Configuration".
6. Set the local_listener and remote_listener parameters in your SPFILE by issuing an `ALTER SYSTEM` command, or by uncommenting it in your PFILE for each instance.
7. Run SRVCTL to configure and start database and instance applications as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.

See Also: *Oracle Universal Installer and OPatch User's Guide* for additional information about creating and using scripts to install Oracle software

C

Configuring Raw Devices for Oracle Real Application Clusters

This appendix provides additional information about configuring raw devices to deploy Real Application Clusters (RAC) using Oracle Database Configuration Assistant (DBCA). You must configure raw devices if you do not use Automatic Storage Management (ASM), or Oracle Cluster File System. The topic in this appendix is:

- Raw Devices Required by DBCA for Non-OCFS Environments

Raw Devices Required by DBCA for Non-OCFS Environments

If you want to use DBCA to create a database on raw storage, then configure the raw devices, as described in this section. These devices are in addition to the Oracle Cluster Registry (OCR) and voting disk required to install Oracle Clusterware. Create these devices before running Oracle Universal Installer (OUI) to install the Oracle Database 10*g* software. DBCA cannot create a RAC database unless you have properly configured the following devices:

- Four raw devices for four tablespace data files
- At least two raw devices for control files
- One raw device for each instance for its own tablespace for automatic undo management
- At least two raw devices for redo log files for each instance
- One raw device for the server parameter file

Note: Each instance has its own redo log files, but all instances in a cluster share the control files and data files. In addition, each instance's online redo log files must be readable by all other instances to enable recovery.

Planning Your Raw Device Creation Strategy

Before installing the Oracle Database 10*g* software with Oracle Real Application Clusters, create enough partitions of specific sizes to support your database, and also leave a few spare partitions of the same size for future expansion. For example, if you have space on your shared disk array, then select a limited set of standard partition sizes for your entire database. Partition sizes of 50MB, 100MB, 500MB, and 1GB are suitable for most databases. Also create a few very small and a few very large spare partitions that are, for example, 1MB and perhaps 5GB or greater in size. Based on

your plans for using each partition, determine the placement of these spare partitions by combining different sizes on one disk, or by segmenting each disk into same-sized partitions.

Note: Ensure that there are spare partitions enables you to perform emergency file relocations or additions if a tablespace data file becomes full.

D

Converting to Oracle Real Application Clusters from Single-Instance Oracle Databases

This chapter describes the procedures for converting from Oracle Database 10*g* single-instance databases to Real Application Clusters (RAC) databases. The topics in this appendix are:

- Single-Instance to Cluster-Enabled Conversion Administrative Issues
- Converting with DBCA
- Converting with rconfig
- Post-Conversion Steps

If you are upgrading from Oracle Parallel Server to RAC or from an earlier version of RAC, then use the Database Upgrade Assistant (DBUA). In other words, the procedures in this chapter assume that your original single-instance database and the target RAC database are the same version of Oracle 10*g*, and running on the same platform.

Note: You must use clustered ASM instances for RAC databases.

See Also: You must comply with the restrictions of the license you have purchased. To understand the restrictions of your license, please refer to the *Database Licensing Information* manual.

Single-Instance to Cluster-Enabled Conversion Administrative Issues

Note the following administrative considerations before conversion:

- Backup procedures should be available before converting from a single-instance Oracle database to RAC.
- Additional archiving considerations apply in RAC environments. In particular, the archive file format requires a thread number. In addition, the archived logs from all instances of a RAC database are required for media recovery. If you archive to a file and you do not use a cluster file system, then a method of accessing the archive logs from all nodes on which the cluster database has instances is required where file systems are not shared.

Converting with DBCA

You can use Database Configuration Assistant to convert from single-instance Oracle databases to RAC. DBCA automates the configuration of the control file attributes, creates the undo tablespaces and the redo logs, and makes the initialization parameter file entries for cluster-enabled environments. It also configures the Oracle Net Services, Oracle Clusterware resources, and the configuration for RAC database management for use by Oracle Enterprise Manager or the SRVCTL utility.

Before you use DBCA to convert a single-instance database to a RAC database, ensure that the following conditions are met:

- A supported hardware and operating system software configuration
- Shared storage, either Oracle Cluster File System or ASM, is available and accessible from all nodes
- Your application has no design characteristics that preclude its use with cluster database processing

If your platform supports a cluster file system, then you can use it for RAC. You can also convert to RAC and use a non-shared file system. In either case, Oracle strongly recommends that you use Oracle Universal Installer (OUI) to perform an Oracle Database 10*g* installation that sets up the Oracle home and inventory in an identical location on each of the selected nodes in your cluster.

This section describes the following scenarios:

- Single Instance on a Non-Cluster Computer to Oracle Database 10g with RAC
- Single Instance on a Cluster to Oracle Database 10g RAC

Single Instance on a Non-Cluster Computer to Oracle Database 10*g* with RAC

To convert from a single-instance Oracle database that is on a non-cluster computer to RAC, perform the procedures described under the following headings in the order shown:

- Back Up the Original Single-Instance Database
- Perform the Pre-Installation Steps
- Set Up the Cluster
- Validate the Cluster
- Copy the Preconfigured Database Image
- Install Oracle Database 10g Software with Real Application Clusters

Back Up the Original Single-Instance Database

Use DBCA to create a preconfigured image of your single-instance database by using the following procedure:

1. Navigate to the bin directory in ORACLE_HOME, and start DBCA.
2. At the Welcome page, click **Next**.
3. On the Operations page, select **Manage Templates**, and click **Next**.
4. On the Template Management page, select **Create a database template** and **From an existing database (structure as well as data)**, and click **Next**.

5. On the Source Database page, enter the database name in the **Database instance** field, and click **Next**.
6. On the Template Properties page, enter a name for your template in the **Name** field. Oracle recommends that you use the database name.

   By default, the template files are generated in the directory `ORACLE_HOME/assistants/dbca/templates` If you choose to do so, you can enter a description of the file in the **Description** field, and change the template file location in the **Template data file** field.

   When you have completed entries, click **Next**.
7. On the Location of Database Related Files page, select **Maintain the file locations**, so that you can restore the database to the current directory structure, and click **Finish**.

DBCA generates two files: a database structure file (*`template_name`*`.dbc`), and a database preconfigured image file (*`template_name`*`.dfb`).

Perform the Pre-Installation Steps

Perform the pre-installation steps as documented in Part II of this book. Then set up shared storage by referring to the "Configure Disk Storage for Oracle Database and Recovery Files" sections in the pre-installation chapters in Part II.

> **See Also:** Storage vendor-specific documentation for setting up the shared disk subsystem and for information about how to mirror and stripe disks

Set Up the Cluster

Form a cluster with the required number of nodes according to your hardware vendor's documentation. When you have configured all of the nodes in your cluster, then install Oracle Clusterware and validate cluster components by referring to the procedures in Chapter 4, "Installing Oracle Clusterware".

Validate the Cluster

Validate the cluster configuration using the Cluster Verification Utility, as described in Chapter 5, "Installing Oracle Database 10g with Real Application Clusters".

Copy the Preconfigured Database Image

Copy the preconfigured database image. This includes copying the database structure `*.dbc` file and the database preconfigured image `*.dfb` file that DBCA created in the previous procedure "Back Up the Original Single-Instance Database" on page D-2 to a temporary location on the node in the cluster from which you plan to run DBCA.

Install Oracle Database 10*g* Software with Real Application Clusters

1. Run the Oracle Universal Installer (OUI) to perform an Oracle installation with the Oracle 10*g* Database with RAC.
2. Select Cluster Installation Mode on the Specify Hardware Cluster Installation page of the Oracle Universal Installer (OUI) and select the nodes to include in your RAC database.
3. On the OUI Database Configuration Types page, select the Advanced install type.

   After installing the Oracle software, OUI runs post-installation configuration tools, such as the Network Configuration Assistant (NetCA), DBCA, and so on.

4. On the DBCA Template Selection page, use the template that you copied to a temporary location in the "Copy the Preconfigured Database Image" procedure. Use the browse option to select the template location.
5. If you selected raw storage on the OUI Storage Options page, then on the DBCA File Locations Tab on the Initialization Parameters page, replace the data files, control files, and log files, and so on, with the corresponding raw device files if you have not setup the `DBCA_RAW_CONFIG` environment variable. You must also replace default database files with raw devices on the Storage page.

   See Also: Chapter 6 for more details about DBCA

6. After creating the RAC database, DBCA displays the Password Management page on which you must change the passwords for database privileged users who have `SYSDBA` and `SYSOPER` roles. When DBCA exits, the conversion process is complete.

Single Instance on a Cluster to Oracle Database 10*g* RAC

There are three scenarios in which a single-instance database can exist on a cluster computer:

- Scenario 1: The Oracle home from which the single-instance database is running is cluster-installed. The steps to perform a conversion to RAC in this scenario are described in "Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home" on page D-4.
- Scenario 2: The Oracle home from which the single-instance database is running is cluster-installed, but the RAC feature is disabled. Refer to "Single Instance on a Cluster Running from a RAC-Disabled Oracle Home" on page D-7 for instructions on how to convert the database to RAC in this environment.
- Scenario 3: The Oracle home from which the single-instance database is running is not cluster-installed. The procedure to convert the single-instance database to RAC in this scenario is described in "Single Instance on a Cluster Running from a Single-Instance Installed Oracle Home" on page D-7.

Use the following procedures to convert your single-instance database on a cluster computer to RAC for all of these scenarios.

Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home

Perform the following procedures to convert a single-instance database on a cluster running from a cluster-installed (Oracle Database 10*g* with RAC) Oracle home.

1. Use DBCA to create a preconfigured image of your single-instance database as described under the heading "Back Up the Original Single-Instance Database" on page D-2. To perform the conversion manually, shut down the single-instance database.
2. To add nodes to your cluster, add and connect these nodes to the cluster as described under the heading "Perform the Pre-Installation Steps" on page D-3. Ensure that all of these nodes can access the shared storage. Also extend the Oracle Clusterware home to the new nodes using the procedures for "Extending Clusterware and Oracle Software to New Nodes" as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.
3. From the existing Oracle home, extend this home to the new nodes using the procedure "Adding Nodes at the Oracle RAC Database Layer" as described in

Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide.

4. From one of the newly added nodes, configure the listeners on the additional nodes using the NetCA. Choose the same port number and protocol that you used on the existing node. If the NetCA displays the existing node in the node list page, then do not select this node, because the listener is already configured on it.
5. Convert the database using one of the following procedures:
   - Automated Conversion Procedure
   - Manual Conversion Procedure

Automated Conversion Procedure 1.If you created the preconfigured image of the single instance database as described under the heading "Back Up the Original Single-Instance Database" on page D-2, then use DBCA to complete the conversion to a RAC database.

2. Start DBCA from the initial node. Select the names of the nodes that you want to include as part of your cluster database. On the Template Selection page, select the preconfigured template that you created in Step 1 on page D-4. Enter the database name and respond to the remaining DBCA prompts.
3. To use raw devices for the cluster database files: When the Initialization Parameters page appears, enter the raw device name for the SPFILE on the File Locations tab. On the Storage page, replace the default database file names with the raw devices for the control files, redo logs, and data files to create the cluster database. Click **Finish,** and create the database.

After creating the RAC database, DBCA displays the Password Management page on which you must change the passwords for the database privileged users who have `SYSDBA` and `SYSOPER` roles. When DBCA exits, the conversion process is complete.

Manual Conversion Procedure Because you did not use DBCA to create a preconfigured image of your single-instance database in Step 1 on page D-4, perform the following steps to complete the conversion:

1. Create the OFA directory structure on each of the nodes that you have added.

   See Also: "Directory Structures for Real Application Clusters" on page E-1 for more information about OFA.

2. If you are converting single-instance database files on a file system to raw devices, then copy the database data files, control files, redo logs, and server parameter file to their corresponding raw devices using the `dd` command. Otherwise, continue to the next step.
3. Re-create the control files by running the `CREATE CONTROLFILE` SQL statement with the `REUSE` keyword and specify `MAXINSTANCES` and `MAXLOGFILES`, and so on, as needed for your RAC configuration. The `MAXINSTANCES` recommended default is 32.
4. Shut down the database instance.
5. If your single-instance database was using an SPFILE parameter file, then create a temporary PFILE from the SPFILE using the following SQL statement:

```
CREATE PFILE='pfile_name' from spfile='spfile_name'
```

6. Set the CLUSTER_DATABASE parameter to TRUE, set the INSTANCE_NUMBER parameter to a unique value for each instance, using a *sid.parameter=value* syntax.

   If you optimized memory usage on your single-instance database, then adjust the size of the SGA to avoid swapping and paging when you convert to RAC. you should make this adjustment because RAC requires about 350 bytes for each buffer to accommodate the Global Cache Service (GCS). For example, if you have 10,000 buffers, then RAC requires about 350*10,000 bytes more memory. Therefore, adjust the size of the SGA by changing the DB_CACHE_SIZE and DB_nK_CACHE_SIZE parameters accordingly.

7. Start up the database instance using the PFILE created in step 5.

8. If your single-instance database was using automatic undo management, then create an undo tablespace for each additional instance using the CREATE UNDO TABLESPACE SQL statement. If you are using raw devices, then ensure that the data file for the undo tablespace is on the raw device.

9. Create redo threads that have at least two redo logs for each additional instance. If you are using raw devices, then ensure that the redo log files are on raw devices. Enable the new redo threads by using an ALTER DATABASE SQL statement. Then shutdown the database instance.

10. Copy the Oracle password file from the initial node, or from the node from which you are working, to the corresponding location on the additional nodes on which the cluster database will have an instance. Make sure that you replace the ORACLE_SID name in each password file appropriately for each additional instance.

11. Add REMOTE_LISTENER=LISTENERS_*DB_NAME* and *sid*.LOCAL_LISTENER=LISTENER_*SID* parameters to the PFILE.

12. Configure the net service entries for the database and instances and address entries for the LOCAL_LISTENER for each instance and REMOTE_LISTENER in the tnsnames.ora file and copy it to all nodes.

13. Create the SPFILE from the PFILE using the procedures under the heading "Procedures for Migrating to the Server Parameter File" on page 7-3. If you are not using a cluster file system, then ensure that the SPFILE is on a raw device.

14. Create the $ORACLE_HOME/dbs/init*sid*.ora file that contains the following entry:

    ```
    spfile='spfile_path_name'
    ```

    where *spfile_path_name* is the complete path name of the SPFILE.

15. Add the configuration for the RAC database and its instance-to-node mapping using SRVCTL.

16. Start the RAC database using SRVCTL.

After starting the database with SRVCTL, your conversion process is complete and, for example, you can run the following SQL statement to see the statuses of all the instances in your RAC database:

```
select * from v$active_instances
```

Single Instance on a Cluster Running from a RAC-Disabled Oracle Home

You can install a single instance on a cluster running from a RAC-disabled Oracle home if you performed a one-node cluster (with RAC) installation, but later disabled the RAC feature by unlinking it from the `oracle` binary before creating the single instance database. (However, you can also select the "local", "non-cluster" selection on the Node Selection Page to create a non-RAC-enabled single-instance home on a cluster.) Perform the following procedures to convert this type of single-instance database to a RAC database:

1. Use DBCA to create a preconfigured image of your single-instance database as described under the heading "Back Up the Original Single-Instance Database" on page D-2. To perform the conversion manually, shut down the single-instance database.
2. Change the directory to the `lib` subdirectory in the `rdbms` directory under the Oracle home.
3. Relink the `oracle` binary by running the following commands:

```
make -f ins_rdbms.mk rac_on
make -f ins_rdbms.mk ioracle
```

4. Continue with Step 2 on page D-4.

Single Instance on a Cluster Running from a Single-Instance Installed Oracle Home

You can only install a single instance on a cluster running from a single-instance installed Oracle home if you selected the local installation option on the OUI Specify Hardware Cluster Installation page during the single-instance Oracle Database 10*g* installation.

To convert this type of database to a RAC database, perform the procedures described under the following headings:

1. "Back Up the Original Single-Instance Database" on page D-2.
2. "Perform the Pre-Installation Steps" on page D-3.
3. "Set Up the Cluster" on page D-3.
4. "Install Oracle Database 10g Software with Real Application Clusters". In this step, make sure that you select a new Oracle home other than the one from which the single-instance database was running.

Converting with rconfig

You can use `rconfig` to assist you with converting a single-instance database installation to a RAC database. The `rconfig` option is a command line utility. The following sections describe how to use these conversion tools:

- Prerequisites for Converting to RAC Databases
- Single-Instance to RAC Conversion Scenarios
- Converting Single-Instance Databases to RAC Using rconfig
- Example of an rconfig XML Input File
- Converting Single-Instance ASM to Cluster ASM

Note: Before you start the conversion, as is always true before you start any major change to your database, always back up your existing database.

Prerequisites for Converting to RAC Databases

Before you convert a single-instance database to a RAC database, ensure that the following conditions are met for each cluster node that you intend to make a RAC database node:

- Oracle Clusterware 10*g* Release 2 (10.2) is installed, configured, and running
- Oracle Real Application Clusters 10*g* Release 2 (10.2) software is installed
- The Oracle binary is enabled for RAC
- Shared storage, either Oracle Cluster File System or ASM, is available and accessible from all nodes
- User equivalence exists for the oracle user

Note: You must use clustered ASM instances for RAC databases. If your single-instance database uses ASM Storage, then before you use `rconfig` to convert the Oracle Database with RAC, convert your ASM installation to a cluster-enabled ASM as described in "Converting Single-Instance ASM to Cluster ASM" on page D-11.

Single-Instance to RAC Conversion Scenarios

The following is a list of scenarios for converting a single-instance Oracle database to an Oracle Real Application Clusters (RAC) database:

- Converting a single-instance Oracle Database 10*g* Release 2 (10.2) to an Oracle 10*g* Release 2 (10.2) RAC database, running out of the *same* Oracle home and using the same data files as the single-instance database

  In this scenario, either use the `rconfig` utility, or use the **Convert to RAC** option on the single-instance database target of Oracle Enterprise Manager Grid Control.

- Converting a single-instance Oracle Database version previous to Oracle Database 10*g* Release 2 (10.2), running out of the same Oracle home and using the same data files as the single-instance database

  In this scenario, use OUI and Database Upgrade Assistant (DBUA) to update the single-instance database to Oracle Database 10*g* Release 2 (10.2). Then use `rconfig` or the Enterprise Manager **Convert to RAC** option, as described in the preceding scenario.

- Converting a single-instance Oracle Database 10*g* Release 2 (10.2) to an Oracle 10*g* Release 2 (10.2) RAC database, running out of a *different* Oracle home and using the same data files as the single-instance database.

  In this scenario, either use the `rconfig` utility in the source database home (the database that you are converting), or use the **Convert to RAC** option in the single-instance database target of Oracle Enterprise Manager Grid Control. Provide the file storage location when prompted.

- Converting a single-instance Oracle Database 10*g* Release 2 (10.2) to an 10*g* Release 2 (10.2) RAC database, running out of a *different* Oracle home, and where the host

where the single-instance database is running is *not* one of the nodes of the RAC database

In this scenario, create a clone image of the single-instance database, and move the clone image to a host that is one of the nodes of the cluster. Then use `rconfig` or the Enterprise Manager **Convert to RAC** option, as described in the preceding scenario.

Converting Single-Instance Databases to RAC Using rconfig

You can use the command line utility `rconfig` to convert a single-instance database to RAC. To use this feature, complete the following steps:

> **See Also:** *Oracle Database Upgrade Guide* for information about upgrading existing Oracle Database installations

1. As the Oracle user, navigate to the directory `$ORACLE_HOME/assistants/rconfig/sampleXMLs`, and open the file `ConvertToRAC.xml` using a text editor, such as vi.
2. Review the `ConvertToRAC.xml` file, and modify the parameters as required for your system. The XML sample file contains comment lines that provide instructions for how to configure the file.

   When you have completed making changes, save the file with the syntax *filename*.xml. Make a note of the name you select.
3. Navigate to the directory `$ORACLE_HOME/bin`, and use the following syntax to run the command `rconfig`:

   rconfig *input.xml*

   where *input.xml* is the name of the XML input file you configured in step 2.

   For example if you create an input XML file called `convert.xml`, then enter the following command

   ```
   $ ./rconfig convert.xml
   ```

Note: The `Convert verify` option in the `ConvertToRAC.xml` file has three options:

- `Convert verify="YES"`: rconfig performs checks to ensure that the prerequisites for single-instance to RAC conversion have been met before it starts conversion
- `Convert verify="NO"`: rconfig does not perform prerequisite checks, and starts conversion
- `Convert verify="ONLY"` rconfig only performs prerequisite checks; it does not start conversion after completing prerequisite checks

---

Example of an rconfig XML Input File

The following is an example of an XML input file for the rconfig utility. In the example, the XML input file is for converting a single-instance database with ASM to a RAC database with ASM in the same Oracle home.

```
<n:RConfig xsi:schemaLocation="http://www.oracle.com/rconfig">
```

```
-
<n:ConvertToRAC>
-
<!--
 Verify does a precheck to ensure all pre-requisites are met, before the
conversion is attempted. Allowable values are: YES|NO|ONLY
-->
-
<n:Convert verify="YES">
-
<!--
Specify current OracleHome of non-rac database for SourceDBHome
-->
<n:SourceDBHome>/oracle/product/10.2.0/db_1</n:SourceDBHome>
-
<!--
Specify OracleHome where the rac database should be configured. It can be same as
SourceDBHome
-->
<n:TargetDBHome>/oracle/product/10.2.0/db_1</n:TargetDBHome>
-
<!--
Specify SID of non-rac database and credential. User with sysdba role is required
to perform conversion
-->
-
<n:SourceDBInfo SID="sales">
-
<n:Credentials>
<n:User>sys</n:User>
<n:Password>oracle</n:Password>
<n:Role>sysdba</n:Role>
</n:Credentials>
</n:SourceDBInfo>
-
<!--
ASMInfo element is required only if the current non-rac database uses ASM Storage
-->
-
<n:ASMInfo SID="+ASM1">
-
<n:Credentials>
<n:User>sys</n:User>
<n:Password>welcome</n:Password>
<n:Role>sysdba</n:Role>
</n:Credentials>
</n:ASMInfo>
-
<!--
Specify the list of nodes that should have rac instances running. LocalNode should
be the first node in this nodelist.
-->
-
<n:NodeList>
<n:Node name="node1"/>
<n:Node name="node2"/>
</n:NodeList>
-
<!--
Specify prefix for rac instances. It can be same as the instance name for non-rac
```

```
database or different. The instance number will be attached to this prefix.
-->
<n:InstancePrefix>sales</n:InstancePrefix>
-
<!--
Specify port for the listener to be configured for rac database.If port="",
alistener existing on localhost will be used for rac database.The listener will be
extended to all nodes in the nodelist
-->
<n:Listener port="1551"/>
-
<!--
Specify the type of storage to be used by rac database. Allowable values are
CFS|ASM. The non-rac database should have same storage type.
-->
-
<n:SharedStorage type="ASM">
-
<!--
Specify Database Area Location to be configured for rac database.If this field is
left empty, current storage will be used for rac database. For CFS, this field
will have directory path.
-->
<n:TargetDatabaseArea>+ASMDG</n:TargetDatabaseArea>
-
<!--
Specify Flash Recovery Area to be configured for rac database. If this field is
left empty, current recovery area of non-rac database will be configured for rac
database. If current database is not using recovery Area, the resulting rac
database will not have a recovery area.
-->
<n:TargetFlashRecoveryArea>+ASMDG</n:TargetFlashRecoveryArea>
</n:SharedStorage>
</n:Convert>
</n:ConvertToRAC>
</n:RConfig>
```

Converting Single-Instance ASM to Cluster ASM

To convert a single-instance node using ASM to a RAC node using ASM:

1. Shut down the database instance, the ASM instance, and the listener.

   Remove the CSS auto-start line from the `/etc/inittab` to enable you to shut down and uninstall CSS.

2. Follow the instructions in Chapter 2 and Chapter 3 to configure each node that you want to have as a cluster member.

3. If any cluster member node has had previous versions of Oracle clusterware (Cluster Ready Services or Oracle Clusterware) installed, then ensure that you remove the file `ocr.loc` on any node that has been labeled previously as a cluster node. The file `ocr.loc` is in `/etc/oracle` on Linux.

4. log in as the Oracle user, and run DBCA in silent mode to deconfigure ASM. To run DBCA in silent mode, navigate to the directory $ORACLE_HOME/bin and use the following command syntax on the local node:

   ```
   dbca -silent -deleteASM
   ```

   You can de-install the single-instance ASM without losing data.

5. Install Oracle Clusterware on all nodes you intend to have as cluster members, following the directions in Chapter 4.
6. Install Cluster ASM using DBCA. At the Node Selection page, select all the cluster member nodes for which you intend to use ASM to manage storage. When the ASM Disk Groups page prompts you to select disk groups, select the existing disk groups you used with the single-instance ASM instance that you deleted in step 4.
7. Start up the single-instance Oracle Database.

Post-Conversion Steps

After completing the conversion, note the following recommendations for RAC environments, as described in the RAC documentation:

- Follow the recommendations for using load balancing and transparent application failover, as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*
- Use locally managed tablespaces instead of dictionary managed tablespaces to reduce contention and manage sequences in RAC as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*
- Follow the guidelines for configuring an interconnect, for using automatic segment space management and for using SRVCTL to administer multiple instances, as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*

The buffer cache and shared pool capacity requirements in RAC are slightly greater than the requirements for single-instance Oracle databases. Therefore, you should increase the size of the buffer cache by about 10%, and the size of the shared pool by about 15%.

E

Directory Structure for Oracle Database 10*g* Oracle Real Application Clusters Environments

This appendix describes the directory structures for Real Application Clusters (RAC) software environments. The topics in this appendix are:

- Understanding the Real Application Clusters Directory Structure
- Directory Structures for Real Application Clusters

Understanding the Real Application Clusters Directory Structure

When you install Oracle Database 10*g* with RAC, all subdirectories are under a top-level `ORACLE_BASE` directory. The `ORACLE_HOME` and `admin` directories are also located under `ORACLE_BASE`.

> **See Also:** *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for further information about the $ORACLE_HOME and /admin directories

Directory Structures for Real Application Clusters

Table E–1 shows the hierarchical directory tree of a sample OFA-compliant database for RAC:

Table E–1 Directory Structure for A Sample OFA-Compliant Environment

Root	Second-Level	Third-Level	Fourth-Level	Description
`$ORACLE_BASE`				`/u01/app/oracle` The default ORACLE_BASE directory.
	`ORACLE_HOME`			`/product/10.2.0` The default name of the Oracle home.
	`/admin`			Administrative directory.
		`/db_unique_name`		Database unique name, the same as `dbname` when database name is eight or fewer characters in length.

Table E–1 (Cont.) Directory Structure for A Sample OFA-Compliant Environment

Root	Second-Level	Third-Level	Fourth-Level	Description
			/bdump /cdump /hdump /pfile /udump	Dump destinations for database server.
$CRS_home				/u01/crs/oracle/product/10.2.0/app The default name of the Oracle Clusterware home. After you install the software, you must change the permissions of all of the parent directories of the Oracle Clusterware home directory to grant write access only to the root user. Because of this, the Oracle Clusterware home directory must not be a subdirectory of the 10*g* Release 2 (10.2) Oracle base directory.
		/bin		Subtree for Oracle binaries.
		/network		Subtree for Oracle Net.

See Also: *Oracle Database Administrator's Reference 10g* Release 2 (10.2) *for UNIX Systems: AIX-Based Systems, HP-UX, HP Tru64 UNIX, Linux, and Solaris Operating System (SPARC)* for further information about the $ORACLE_HOME and /admin directories

F

How to Stop Processes in an Existing Oracle Real Application Clusters Database

This appendix describes how to stop all processes in an Oracle Real Application Clusters (RAC) database, in preparation for adding additional products to an existing database, or in preparation for patch updates. The topics in this appendix are:

- Back Up the Oracle Real Application Clusters Database
- Shut Down Oracle Real Application Clusters Databases
- Stop All Oracle Processes

Back Up the Oracle Real Application Clusters Database

Before you make any changes to the Oracle software, Oracle recommends that you create a backup of the Oracle Database 10g installation.

Shut Down Oracle Real Application Clusters Databases

Shut down any existing Oracle Database instances on each node, with normal or immediate priority.

If Automatic Storage Management (ASM) is running, shut down all databases that use ASM, and then shut down the ASM instance on each node of the cluster.

Note: If you are using Oracle Clusterware or Cluster Ready Services, then you must shut down all Oracle Database instances on all cluster nodes before modifying the Oracle software. If you are performing a patch update, review the instructions in the Patch Set Notes for detailed instructions.

Stop All Oracle Processes

Stop all listener and other processes running in the Oracle home directories where you want to modify the database software.

Note: If you shut down ASM instances, then you must first shut down all database instances that use ASM, even if these databases run from different Oracle homes.

Stopping Oracle Clusterware or Cluster Ready Services Processes

If you are modifying an Oracle Clusterware or Oracle Cluster Ready Services (CRS) installation, shut down the following Oracle Database 10g services.

Note: You must perform these steps in the order listed.

1. Shut down any processes in the Oracle home on each node that might be accessing a database; for example, Oracle Enterprise Manager Database Control.

   Note: Before you shut down any processes that are monitored by Enterprise Manager Grid Control, set a blackout in Grid Control for the processes that you intend to shut down. This is necessary so that the availability records for these processes indicate that the shutdown was planned downtime, rather than an unplanned system outage.

2. Shut down all RAC instances on all nodes. To shut down all RAC instances for a database, enter the following command, where *db_name* is the name of the database:

   ```
   $ oracle_home/bin/srvctl stop database -d db_name
   ```

3. Shut down all ASM instances on all nodes. To shut down an ASM instance, enter the following command, where *node* is the name of the node where the ASM instance is running:

   ```
   $ oracle_home/bin/srvctl stop asm -n node
   ```

4. Stop all node applications on all nodes. To stop node applications running on a node, enter the following command, where *node* is the name of the node where the applications are running

   ```
   $ oracle_home/bin/srvctl stop nodeapps -n node
   ```

5. Shut down the Oracle Clusterware or CRS process by entering the following command on all nodes as the root user

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

   Note: If the Oracle Clusterware or CRS installation is not on a shared Oracle home, then you can upgrade the Oracle Clusterware or CRS installation one node at a time. To do this, perform the preceding steps only on the first node that you are upgrading, and then follow the instructions on the Installer screen.

Stopping Oracle Database 10g Processes Before Adding Products or Upgrading

This section provides an overview of what needs to be done before adding additional products to Oracle Database 10*g* Release 2 (10.2). If you are performing a patch upgrade, then refer to the Database Patch Set Notes for the patch for additional instructions.

If node applications are running in the RAC Oracle home, then shut down the following Oracle Database 10g services before modifying the Oracle Database software:

Note: You must perform these steps in the order listed.

1. Shut down any processes in the Oracle home on each node that can access a database; for example, Oracle Enterprise Manager Database Control.

   Note: Before you shut down any processes that are monitored by Enterprise Manager Grid Control, set a blackout in Grid Control for the processes that you intend to shut down. This is necessary so that the availability records for these processes indicate that the shutdown was planned downtime, rather than an unplanned system outage.

2. Shut down all RAC instances on all nodes. To shut down all RAC instances for a database, enter the following command, where *db_name* is the name of the database:

   ```
   $ oracle_home/bin/srvctl stop database -d db_name
   ```

3. Shut down all ASM instances on all nodes. To shut down an ASM instance, enter the following command, where *node* is the name of the node where the ASM instance is running:

   ```
   $ oracle_home/bin/srvctl stop asm -n node
   ```

4. Stop all node applications on all nodes. To stop node applications running on a node, enter the following command, where *node* is the name of the node where the applications are running:

   ```
   $ oracle_home/bin/srvctl stop nodeapps -n node
   ```

Index

A

Additional Real Application Clusters
documentation, 1-1
Advanced
configuration type, 5-4, 5-5
API libraries
for Oracle Clusterware, xxii
architecture
checking system architecture, 2-19
Optimal Flexible Architecture (OFA), 5-5
archive logs
destinations, converting to cluster database, D-1
ASM
and OFA, xxi
benefits of, 1-9
characteristics of failure groups, 1-9
compared to logical volume managers, 1-9
compared to RAID, 1-9
database recovery area for, 1-10
disk groups, 1-9
failure groups, 1-9
overview, 1-8
recommendations for disk groups, 1-9
redundancy levels, 1-9
ASM only installation, 1-10
Automatic Storage Management
block device names, 3-19
changing owner and permissions of raw devices
on Linux, 3-11, 3-24
characteristics of failure groups, 3-14
checking disk availability, 3-9, 3-18, 3-22
configuring disks for Automatic Storage
Management, 3-16
disk groups, 3-13
displaying attached disks, 3-18
on Linux, 3-9, 3-22
failure groups, 3-13
examples, 3-14
identifying, 3-14
identifying available disks, 3-18
identifying available disks on Linux, 3-9, 3-22
identifying disks, 3-18
identifying disks on Linux, 3-9, 3-22
recommendations for disk groups, 3-13
space required for preconfigured database, 3-13
storage option for data files, 3-2
automatic undo management, 9-3
Available
service configuration policy, 6-2

B

background process
crsd, 4-9
evmd, 4-9
ocssd, 4-9
oprocd, 4-9
backups
and converting to cluster database, D-1
base directory
See Oracle base directory
Bash shell
default user startup file, 2-16
setting shell limits on Linux x86, 2-27
.bash_profile file, 2-16
Basic
TAF failover policy, 6-3
block device
device name, 3-19
Bourne shell
default user startup file, 2-16
setting shell limits on Linux x86, 2-27

C

C compiler
requirement on Linux x86, 2-23
C shell
default user startup file, 2-16
setting shell limits on Linux x86, 2-27
checking existence of the nobody user, 2-9
chmod command, 2-32, 2-33, 3-7, 3-24
chown command, 2-32, 2-33, 3-7, 3-24
cluster database
installed configuration, 9-2
cluster file system
database recovery area and, 1-10
single-instance storage option for data files, 3-2
storage option for data files, 3-2
cluster name
requirements for, 4-5

cluster nodes
private node names, 4-6
public node names, 4-5
specifying uids and gids, 2-10
virtual node names, 4-6
Cluster Synchronization Service Daemon, xxii, 1-12
Cluster Synchronization Services, xxii, 1-5, 1-12, 4-4
Cluster Verification Utility
about, xix, 1-2
DBCA database creation stage readiness check, 6-3
hardware and operating system setup stage verification, 2-27
Oracle Clusterware configuration check, 4-1
shared storage area check, 3-4
user equivalency troubleshooting, 4-2
verifying readiness for database installation, 5-1
clusterware diagnostics, A-2
components
created when using DBCA, 9-2
configuration types
Advanced, 5-4
Data Warehouse, 5-4
Do not create a starter database, 5-4
General Purpose, 5-4
Transaction Processing, 5-4
connection load balancing, 9-4
control file
installed configuration, 9-3
control files, 1-12
described, 9-3
raw devices for
on Linux, 3-21
convert to cluster database
administrative considerations, D-1
from non-cluster system, D-2
from single-instance, D-4
from single-instance to Real Application Clusters, D-1
post-conversion, D-13
reasons not to convert, D-2
to Real Application Clusters from single-instance Oracle databases, D-1
converting
from single-instance to Real Application Clusters, B-1, D-11
to Real Application Clusters from single-instance Oracle databases, B-1
create database
using scripts, B-1
creating
Real Application Clusters database
with Database Configuration Assistant, 5-5, 6-1, 6-4
cross-node registration, 9-4
CRS
raw device for OCR on Linux, 3-8
CRSD, 4-9
csh.login file, 2-27
csh.login.local file, 2-27
CSS, xxii, 1-5, 1-12, 4-4
CCSD, xxii
OCCSD, 1-12, 4-4
CSSD, xxii
custom database
failure groups for Automatic Storage Management, 3-14
requirements when using ASM, 1-10
requirements when using Automatic Storage Management, 3-13
Custom installation type
reasons for choosing, 2-5
CVU
See Cluster Verification Utility

D

data files
creating separate directories for, 3-6
setting permissions on data file directories, 3-7
single-instance database storage options, 3-2
storage options, 3-2
data loss
minimizing with ASM, 1-9
minimizing with Automatic Storage Management, 3-14
Data Warehouse
configuration type, 5-4
database
ASM requirements, 1-10
components, created when using DBCA, 9-2
Database Configuration Assistant
ASM Disk Groups page, 6-5
components created by, 9-2
control files, 9-3
creating Real Application Clusters database
after installation, 6-4
during installation, 5-5, 6-1
datafiles, 9-2
deleting databases with, 6-6
deleting Real Application Clusters databases, 6-6
initialization parameter files, 9-4
Initialization Parameters page, 6-5
List of Cluster Databases page, 6-6
Operations page, 6-6
raw storage requirements, C-1
Recovery Configuration page, 6-5
redo log files, 9-3
rollback segments, 9-3
Summary dialog, 6-7
tablespaces, 9-2
using, 6-1
Welcome page, 6-4
database configuration type
selecting, 5-4
database creation using scripts, B-1
database recovery area
shared storage requirement for, 1-10
databases
Automatic Storage Management

requirements, 3-13
datafiles, 1-12
and DBCA, 9-2
described, 9-2
managing with ASM, 1-8
dba group
and Automatic Storage Management disks on Linux, 3-24
and SYSDBA privilege, 2-5
creating, 2-7
creating on other nodes, 2-10
description, 2-5
raw device group on Linux, 3-23
DBCA
delete database, 6-6
DBCA_RAW_CONFIG environment variable
on Linux, 3-25
dedicated servers, 9-4
default file mode creation mask
setting, 2-15
de-install Oracle Clusterware software, 5-9
de-install Oracle Database software, 5-7
delete database with DBCA, 6-6
device names
for raw devices on Linux, 3-10, 3-23
IDE disks, 3-18
IDE disks on Linux, 3-9, 3-22
RAID, 3-19
RAID on Linux, 3-9, 3-22
SCSI disks, 3-18
SCSI disks on Linux, 3-9, 3-22
diagnostics, A-2
directory
creating separate data file directories, 3-6
Oracle base directory, 2-29
Oracle home directory, 2-30
Oracle Inventory directory, 2-29
oraInventory, 2-29
permission for data file directories, 3-7
directory structure, E-1
disk devices
managing with ASM, 1-8
disk group
ASM, 1-8, 1-9
Automatic Storage Management, 3-13
recommendations for ASM disk groups, 1-9
recommendations for Automatic Storage Management disk groups, 3-13
disk space
checking, 2-19
requirements for preconfigured database in Automatic Storage Management, 3-13
disks
checking availability for Automatic Storage Management, 3-9, 3-18, 3-22
displaying attached disks, 3-18
on Linux, 3-9, 3-22
raw voting disk on Linux, 3-8
DISPLAY environment variable
setting, 2-15, 2-16
Do not create a database
configuration type, 5-4
documentation
Oracle Clusterware, 1-1
Oracle Real Application Clusters Administration and Deployment Guide, 1-2
Real Application Clusters, 1-1

E

Easy Connect, 9-11
emulator
installing from X emulator, 2-4
env command, 2-34
environment
checking settings, 2-34
configuring for oracle user, 2-15
environment variables
DBCA_RAW_CONFIG, 3-25
DISPLAY, 2-15, 2-16
ORACLE_BASE, 2-15, 2-29, 2-32
ORACLE_HOME, 2-15, 2-33, 2-35
ORACLE_SID, 2-15
PATH, 2-15
removing from shell startup file, 2-16
SHELL, 2-15
TEMP and TMPDIR, 2-16, 2-18
TNS_ADMIN, 2-33
errata
Linux kernel errata, 2-24
error messages
Real Application Clusters management tools, A-2
/etc/csh.login file, 2-27
/etc/csh.login.local file, 2-27
/etc/pam.d/login file, 2-27
/etc/profile file, 2-27
/etc/profile.local file, 2-27
/etc/raw file, 3-11, 3-24
/etc/security/limits.so file, 2-27
/etc/sysconfig/rawdevices file, 3-10, 3-23
/etc/sysctl.conf file, 2-26
EVMD, 4-9
EXAMPLE tablespace
raw device for on Linux, 3-21
examples
Automatic Storage Management failure groups, 3-14
Oracle base directories, 2-29
external jobs
UNIX user required for, 2-5
external redundancy
ASM redundancy level, 1-9
extjob executable
UNIX user required for, 2-5

F

failover

and service registration, 9-4
of single-instance databases using Oracle Clusterware, 3-2
failure group
ASM, 1-9
Automatic Storage Management, 3-13
characteristics of ASM failure group, 1-9
characteristics of Automatic Storage Management failure group, 3-14
examples of Automatic Storage Management failure groups, 3-14
fdisk command, 3-9, 3-18, 3-22
features, new, xix
file mode creation mask
setting, 2-15
file system
appropriate for Oracle base directory, 2-32
storage option for data files, 3-2
storage option for single instance data files, 3-2
file-max file, 2-25
file-max parameter
recommended value on Linux x86, 2-25
files
.bash_profile, 2-16
control files
raw devices for on Linux, 3-21
default shell startup file, 2-27
editing shell startup file, 2-16
/etc/csh.login, 2-27
/etc/csh.login.local, 2-27
/etc/pam.d/login, 2-27
/etc/profile, 2-27
/etc/profile.local, 2-27
/etc/raw, 3-11, 3-24
/etc/security/limits.so, 2-27
/etc/sysconfig/rawdevices, 3-10, 3-23
/etc/sysctl.conf, 2-26
.login, 2-16
oraInst.loc, 2-7, 2-31
oratab, 2-31
password file
raw device for on Linux, 3-21
/proc/sys/fs/file-max, 2-25
/proc/sys/kernel/sem, 2-25
/proc/sys/kernel/shmall, 2-25
/proc/sys/kernel/shmmax, 2-25
/proc/sys/kernel/shmmni, 2-25
/proc/sys/net/ipv4/ip_local_port_range, 2-25
.profile, 2-16
profile.local, 2-27
raw device mapping file
on Linux, 3-24
redo log files
raw devices for on Linux, 3-21
server parameter file
raw devices for on Linux, 3-21
SPFILE
raw devices for on Linux, 3-21
filesets, 2-22
Flash Recovery
and Oracle-Managed Files, D-12
for raw devices on Linux, 3-11, 3-24

G

General Purpose
configuration type, 5-4
gid
identifying existing, 2-10
specifying, 2-10
specifying on other nodes, 2-10
group IDs
identifying existing, 2-10
specifying, 2-10
specifying on other nodes, 2-10
groups
checking for existing oinstall group, 2-7
creating identical groups on other nodes, 2-10
creating the dba group, 2-7
creating the oinstall group, 2-7
creating the oper group, 2-8
specifying when creating UNIX users, 2-10
UNIX OSDBA group (dba), 2-5
UNIX OSOPER group (oper), 2-5

H

hardware requirements, 2-18
high redundancy
ASM redundancy level, 1-9
home directory
See Oracle home directory

I

id command, 2-9, 2-10
IDE disks
device names, 3-18
device names on Linux, 3-9, 3-22
initialization parameter files, 9-4
for instances, 7-1
listener parameters, 9-5
initialization parameters
DISPATCHERS, 9-4
LOCAL_LISTENER, 9-4
REMOTE_LISTENER, 9-4, 9-8
installation
directory structure, E-1
ldap.ora file, 9-6
listener.ora file, 9-5
non-interactive, 1-7, B-1
of additional products after installation is completed, 6-6
overview, 1-7
requirements, hardware, 1-2
requirements, software, 1-2
tnsnames.ora file, 9-7
verifying raw devices, 6-4
installation types
and ASM requirements, 1-10
and Automatic Storage Management

requirements, 3-13
instance
instance identifier (SID), 2-15
preferences, 6-2
instances
initialization parameter files, 7-1
ip_local_port_range file, 2-25
ip_local_port_range parameter
recommended value on Linux x86, 2-25

J

JDK requirements, 2-22

K

kernel
Linux errata, 2-24
kernel parameters
checking on Linux x86, 2-25
configuring on Linux x86, 2-25
making changes persist on Linux, 2-26
setting on Linux x86, 2-25
Korn shell
default user startup file, 2-16
setting shell limits on Linux x86, 2-27
ksh
See Korn shell

L

ldap.ora file, 9-6
creating, 9-6
default configuration, 9-6
Lightweight Directory Access Protocol (LDAP), 9-6
limit command, 2-27
limits.so file, 2-27
Linux
binding raw devices to partitions, 3-10, 3-23
creating a raw device mapping file, 3-24
creating partitions, 3-9, 3-22
displaying attached disks, 3-9, 3-22
IDE disk device names, 3-9, 3-22
identifying disks for Automatic Storage Management, 3-9, 3-22
kernel errata, 2-24
making kernel parameter changes persist, 2-26
RAID device names, 3-9, 3-22
raw device sizes, 3-8
SCSI disk device names, 3-9, 3-22
setting shell limits, 2-27
specifying owner and permissions on raw devices, 3-23
Linux x86
checking distribution, 2-24
checking kernel parameters, 2-25
checking version, 2-24
configuring kernel parameters, 2-25
setting kernel parameters, 2-25
setting shell limits, 2-27
List of Cluster Databases page, 6-6
listener
automatic migration from 10.1 to 10.2, 6-3
identifying Oracle home for, 2-35
local, 9-4
lsnrctl command, 2-36
registration, 9-6
stopping, 2-35, 2-36
stopping existing listener process, 2-34
listener.ora file, 9-5
configuring, 9-5
default configuration, 9-5
listeners
listener.ora file, 9-5
load balancing
and service registration, 9-4
local listener, 9-4
logical volume manager
See LVM
.login file, 2-16
login file, 2-27
lsdev command, 3-9, 3-18, 3-22
lsnrctl command, 2-36
LVM
compared to ASM, 1-9
recommendations for ASM, 1-9
recommendations for Automatic Storage Management, 3-13

M

mapping file
for raw devices
on Linux, 3-24
mask
setting default file mode creation mask, 2-15
memory requirements, 2-18
MetaLink
Oracle patches, 8-1
migration
from single-instance, D-2
mkdir command, 2-32, 2-33, 3-7
mode
setting default file mode creation mask, 2-15
mount point
for Oracle base directory, 2-29
multiple voting disks, xxii, 1-5

N

Net naming methods, 9-11
net service names, 9-7
network configuration files
ldap.ora.ora, 9-6
listener.ora, 9-5
sqlnet.ora, 9-11
tnsnames.ora, 9-7
network directory, E-2
Network Information Services
See NIS
new features, xix

NIS
alternative to local users and groups, 2-6
nobody user
checking existence of, 2-9
description, 2-5
node applications, 6-1
nofile
shell limit on Linux x86, 2-27
None
TAF failover policy, 6-3
non-interactive installation, 1-7
normal redundancy
ASM redundancy level, 1-9
Not used
service configuration policy, 6-2
nproc
shell limit on Linux x86, 2-27

O

OCCSD, 4-4, 4-9
OCFS
checking on Linux, 2-24, 3-6
download location for Linux, 2-24, 3-6
restriction for Oracle base directory on Linux, 2-32, 2-33
OCR
installed configuration, 9-1
mirroring, 3-5
raw device for on Linux, 3-8
OCR contents, 9-1
OCR. *See* Oracle Cluster Registry
OCSSD, 1-12
OCSSD. *See* CSSD
OFA
and ASM, xxi
and Oracle Clusterware, xx
recommendations for Oracle base directory, 2-29
recommended path for Oracle base directory, 2-29
recommended path for Oracle home directory, 2-30
recommended path for Oracle Inventory directory, 2-29
oinstall group
checking for existing, 2-7
creating, 2-7
creating on other nodes, 2-10
description, 2-5
olsnodes command, A-2
oper group
and SYSOPER privilege, 2-5
creating, 2-8
creating on other nodes, 2-10
description, 2-5
operating system
checking distribution and version of Linux x86, 2-24
operating system account users
requirements, 2-5
operating system groups
requirements, 2-5
operating system requirements, 2-22
OPROCD, 4-9
Optimal Flexible Architecture
and ASM, xxi
and Oracle Clusterware, xx
See OFA
Optimal Flexible Architecture (OFA), 5-5
Oracle base directory
and ORACLE_BASE environment variable, 2-29
creating, 2-32, 2-33
creating new, 2-32
description, 2-29
determining disk space on, 2-31
equivalent directory on Windows, 2-29
examples, 2-29
identifying appropriate file system, 2-32
identifying existing, 2-30
mount point for, 2-29
OCFS restriction on Linux, 2-32, 2-33
recommended path, 2-29
relationship with Oracle Software Owner user, 2-29
requirement for, 2-29
requirement for RAC installations, 2-32, 2-33
requirements for existing directory, 2-31
Oracle Cluster Registry
configuration of, 4-6
See OCR
Oracle Clusterware
and OFA, xx
and single-instance databases, 3-2
API libraries for, xxii
documentation, 1-1
installing, 4-1
installing with Oracle Universal Installer, 4-7
raw device for voting disk on Linux, 3-8
Oracle Database
creating data file directories, 3-6
data file storage options, 3-2
privileged groups, 2-5
requirements with ASM, 1-10
requirements with Automatic Storage Management, 3-13
setting ORACLE_SID environment variable, 2-15
single instance data file storage options, 3-2
Oracle Enterprise Manager
post-installation configuration, 8-3
using to convert single-instance databases to Real Application Clusters, D-11
Oracle home directory
description, 2-30
identifying for listener, 2-35
recommended path, 2-30
requirement for, 2-30
requirements, 2-30
using to identify Oracle base directory, 2-31
Oracle home name, 2-30

Oracle Inventory
description, 2-29
pointer file, 2-7
Oracle Inventory directory
description, 2-29
recommended path, 2-29
Oracle Inventory group
checking for existing, 2-7
creating, 2-7
creating on other nodes, 2-10
description, 2-5
Oracle Net
identifying Oracle home for listener, 2-35
lsnrctl command, 2-36
stopping existing listener, 2-34
stopping listener, 2-35
stopping the listener, 2-36
Oracle Notification Server Configuration Assistant, 4-7
Oracle Private Interconnect Configuration Assistant, 4-7
Oracle Software Owner user
and Automatic Storage Management disks, 3-24
configuring environment for, 2-15
creating, 2-8
creating on other nodes, 2-10
description, 2-6
determining default shell, 2-15
raw device owner
on Linux, 3-23
relationship with Oracle base directory, 2-29
required group membership, 2-6
setting shell limits for on Linux x86, 2-27
Oracle Universal Installer
and Oracle Clusterware, 4-7
overview of processing, 1-7
Welcome page, 4-7, 5-5, 5-10
oracle user
and Automatic Storage Management disks, 3-24
configuring environment for, 2-15
creating, 2-8, 2-9
creating on other nodes, 2-10
description, 2-6
determining default shell, 2-15
raw device owner
on Linux, 3-23
relationship with Oracle base directory, 2-29
required group membership, 2-6
setting shell limits for on Linux x86, 2-27
ORACLE_BASE environment variable, 2-29, 2-32
removing from shell startup file, 2-16
setting, 2-15
ORACLE_HOME environment variable
removing from shell startup file, 2-16
setting, 2-35, 8-4
unsetting, 2-33
ORACLE_SID environment variable
removing from shell startup file, 2-16
setting, 2-15, 8-4
Oracle9<italic>i upgrades
stopping processes, 2-34
OracleMetaLink, 8-1
oraInst.loc file, 2-31
location, 2-7
location of, 2-7
oraInventory
creating, 2-7
oraInventory directory
See Oracle Inventory directory
oratab file, 9-2
format, 2-31
location of oratab file, 2-31
OSDBA group
and Automatic Storage Management disks on Linux, 3-24
and SYSDBA privilege, 2-5
creating, 2-7
creating on other nodes, 2-10
description, 2-5
raw device group
on Linux, 3-23
OSOPER group
and SYSOPER privilege, 2-5
creating, 2-8
creating on other nodes, 2-10
description, 2-5
OUI
see Oracle Universal Installer

P

packages
checking on Linux x86, 2-24
parameter file search order, 7-2
parameters
initialization, 7-1
partition
using with ASM, 1-9
using with Automatic Storage Management, 3-13
partitions
binding to raw devices on Linux, 3-10, 3-23
creating on Linux, 3-9, 3-22
creating raw partitions, 3-8, 3-21
required sizes for raw devices on Linux, 3-8
passwd command, 2-11
password file
raw device for
on Linux, 3-21
patches
download, 8-1
install, 8-1
OracleMetaLink, 8-1
PATH environment variable
setting, 2-15
PC X server
installing from, 2-4
permissions
for data file directories, 3-7
for Oracle base directory, 2-32, 2-33
physical RAM requirements, 2-18

post-installation
Oracle Enterprise Manager configuration, 8-3
patch download and install, 8-1
product configuration, 8-2
root.sh back up, 8-3
user accounts setup, 8-3
Precompilers
requirements on Linux x86, 2-23
preconfigured database
Automatic Storage Management disk space requirements, 3-13
requirements when using ASM, 1-10
requirements when using Automatic Storage Management, 3-13
preconfigured database installation types, 5-4
Preferred
service configuration policy, 6-2
pre-installation
raw device creation, C-1
privileged groups
for Oracle Database, 2-5
Pro*C/C++
requirements on Linux x86, 2-23
process
stopping existing, 2-34
stopping existing listener process, 2-34
stopping listener process, 2-35
processor
checking system architecture, 2-19
/proc/sys/fs/file-max file, 2-25
/proc/sys/kernel/sem file, 2-25
/proc/sys/kernel/shmall file, 2-25
/proc/sys/kernel/shmmni file, 2-25
/proc/sys/net/core/rmem_default file, 2-25
/proc/sys/net/core/rmem_max file, 2-25
/proc/sys/net/core/wmem_default file, 2-25
/proc/sys/net/core/wmem_max file, 2-25
/proc/sys/net/ipv4/ip_local_port_range file, 2-25
.profile file, 2-16
profile file, 2-27
profile.local file, 2-27
ps command, 2-35

R

RAC
configuring disks for Automatic Storage Management on Linux, 3-18
configuring disks for raw devices on Linux, 3-21
requirements for Oracle base directory, 2-32, 2-33
RAC high availability extensions
configuring services for, 6-2
TAF policies for, 6-2
RAID
and mirroring OCR and voting disk, 3-5
compared to ASM, 1-9
device names, 3-19
device names on Linux, 3-9, 3-22
recommended ASM redundancy level, 1-9
recommended Automatic Storage Management redundancy level, 3-13
RAM requirements, 2-18
raw command, 3-10, 3-23
raw device
for OCR
on Linux, 3-8
for password file
on Linux, 3-21
for server parameter file
on Linux, 3-21
for SPFILE
on Linux, 3-21
for voting disk
on Linux, 3-8
raw devices
binding to partitions on Linux, 3-10, 3-23
changing permissions and owner for Automatic Storage Management on Linux, 3-11, 3-24
creating a raw device mapping file
on Linux, 3-24
creating partitions on Linux, 3-9, 3-22
creating raw partitions, 3-8, 3-21
Database Configuration Assistant, C-1
device names on Linux, 3-10, 3-23
device names, 3-11, 3-24
for control files
on Linux, 3-21
for EXAMPLE tablespace
on Linux, 3-21
for redo log files
on Linux, 3-21
for SYSAUX tablespace
on Linux, 3-21
for SYSTEM tablespace, 3-21
for TEMP tablespace
on Linux, 3-21
for UNDOTBS tablespace
on Linux, 3-21
for USER tablespace
on Linux, 3-21
required sizes
on Linux, 3-8
setting up, C-1
specifying owner and permissions on Linux, 3-23
storage option for data files, 3-2
value for DBCA_RAW_CONFIG environment variable
on Linux, 3-25
verification, 6-4
raw file, 3-11, 3-24
raw storage
Database Configuration Assistant requirements, C-1
rawdevices file, 3-10, 3-23
RBS tablespace
description, 9-3
Real Application Clusters
components, 1-11
databases, deleting, 6-6
installation requirements, 1-2

installed components of, 1-12
management tools error messages, A-2
overview, 1-1, 6-1, 9-1
raw device setup, C-1
recovery files
shared storage requirement for, 1-10
Red Hat
binding partitions to raw devices, 3-10, 3-23
operating system requirements, 2-22
Red Hat Package Manager
See RPM
redo log file
installed configuration, 9-3
redo log files, 1-12
described, 9-3
raw devices for
on Linux, 3-21
redundancy level
and space requirements for preconfigured database, 3-13
for ASM, 1-9
registration
cross-node, 9-4
requirements
hardware, 2-18
rmem_default file, 2-25
rmem_default parameter
recommended value on Linux, 2-25
rmem_max file, 2-25
rmem_max parameter
recommended value on Linux, 2-25
rollback segments
described, 9-3
root user
logging in as, 2-4
root.sh, 4-7
back up, 8-3
RPM
checking on Linux x86, 2-24
rpm command, 2-24

S

scripts to create a Real Application Clusters database, B-1
SCSI disks
device names, 3-18
device names on Linux, 3-9, 3-22
sem file, 2-25
semmni parameter
recommended value on Linux, 2-25
semmns parameter
recommended value on Linux, 2-25
semmsl parameter
recommended value on Linux, 2-25
semopm parameter
recommended value on Linux, 2-25
server parameter file
raw device for
on Linux, 3-21
server parameter files, 1-12, 7-1, 9-4
errors, 7-4
service command, 3-10, 3-23
Service Management
using, 6-2
service registration
configuring, 9-4
services, 6-2
shared configuration file, 9-1
shared server, 9-4
shared storage
requirement for recovery area, 1-10
shell
default shell startup file, 2-27
determining default shell for oracle user, 2-15
SHELL environment variable
checking value of, 2-15
shell limits
setting on Linux x86, 2-27
shell startup file
editing, 2-16
removing environment variables, 2-16
shmall file, 2-25
shmall parameter
recommended value on Linux x86, 2-25
shmmax file, 2-25
shmmax parameter
recommended value on Linux x86, 2-25
shmmni file, 2-25
shmmni parameter
recommended value on Linux x86, 2-25
SID
setting ORACLE_SID environment variable, 2-15
silent installation. *See* installation
non-interactive
single-instance databases
converting, D-11
software requirements, 2-22
checking software requirements, 2-24
SPFILE
default creation, 7-1
default location, 7-2
managing on raw device, 7-2
migrating to, 7-3
raw device for
on Linux, 3-21
sqlnet.ora file, 9-11
startup file
default shell startup file, 2-27
for shell, 2-16
Summary dialog, 6-7
SUSE
binding partitions to raw devices, 3-10, 3-23
operating system requirements, 2-22
swap space
requirements, 2-18
SYSAUX tablespace
raw device for
on Linux, 3-21
sysctl command, 2-25

sysctl.conf file, 2-26
SYSDBA privilege
  associated UNIX group, 2-5
SYSOPER privilege
  associated UNIX group, 2-5
system architecture
  checking, 2-19
SYSTEM tablespace
  description, 9-2
  raw device for
    on Linux, 3-21

T

tablespaces
  and DBCA, 9-2
  expanding for large sorts, 9-3
  RBS, 9-3
  SYSTEM, 9-2
  TEMP, 9-3
  undo tablespaces for automatic undo
    management, 9-3
  USERS, 9-2
TAF failover policies
  Basic, 6-3
  None, 6-3
tcsh shell
  setting shell limits on Linux x86, 2-27
TEMP environment variable, 2-18
  setting, 2-16
TEMP tablespace
  described, 9-3
  raw device for
    on Linux, 3-21
temporary directory, 2-18
temporary disk space
  checking, 2-18
  freeing, 2-18
  requirements, 2-18
/tmp directory
  checking space in, 2-18
  freeing space in, 2-18
TMPDIR environment variable, 2-18
  setting, 2-16
TNS_ADMIN environment variable
  unsetting, 2-33
tnsnames.ora file, 9-7
  default configuration, 9-7
Transaction Processing
  configuration type, 5-4
Transparent Application Failover (TAF) policies, 6-2
troubleshooting
  user equivalency, 4-2

U

uid
  identifying existing, 2-10
  specifying, 2-10
  specifying on other nodes, 2-10
ulimit command, 2-27
umask, 2-34
umask command, 2-15, 2-34
undo management, 9-3
UNDOTBS tablespace
  raw device for
    on Linux, 3-21
UNIX commands, 2-34
  chmod, 2-32, 2-33, 3-7, 3-24
  chown, 2-32, 2-33, 3-7, 3-24
  env, 2-34
  fdisk, 3-9, 3-18, 3-22
  id, 2-9, 2-10
  limit, 2-27
  lsdev, 3-9, 3-18, 3-22
  mkdir, 2-32, 2-33, 3-7
  passwd, 2-11
  ps, 2-35
  raw, 3-10, 3-23
  rpm, 2-24
  service, 3-10, 3-23
  swap, 2-18
  swapon, 2-18
  sysctl, 2-25
  ulimit, 2-27
  umask, 2-15
  unset, 2-33
  unsetenv, 2-33
  useradd, 2-9, 2-10
  xhost, 2-4
  xterm, 2-4
UNIX groups
  checking for existing oinstall group, 2-7
  creating identical groups on other nodes, 2-10
  creating the dba group, 2-7
  creating the oinstall group, 2-7
  creating the oper group, 2-8
  oinstall, 2-5
  OSDBA (dba), 2-5
  OSOPER (oper), 2-5
  required for oracle user, 2-6
  specifying when creating users, 2-10
  using NIS, 2-6, 2-10
UNIX users
  checking existence of the nobody user, 2-9
  creating identical users on other nodes, 2-10
  creating the oracle user, 2-8
  nobody, 2-5
  oracle, 2-6
  required for external jobs, 2-5
  setting shell limits for on Linux x86, 2-27
  specifying groups when creating, 2-10
  unprivileged user, 2-5
  using NIS, 2-6, 2-10
UNIX workstation
  installing from, 2-4
unprivileged user
  nobody user, 2-5
unset command, 2-33
unsetenv command, 2-33

user accounts
post-installation setup, 8-3
user equivalence
testing, 4-2
user IDs
identifying existing, 2-10
specifying, 2-10
specifying on other nodes, 2-10
USER tablespace
raw device for
on Linux, 3-21
useradd command, 2-9, 2-10
users
checking existence of the nobody user, 2-9
creating identical users on other nodes, 2-10
creating the oracle user, 2-8
Oracle Software Owner user (oracle), 2-6
setting shell limits for UNIX users on Linux
x86, 2-27
specifying UNIX groups when creating, 2-10
UNIX nobody user, 2-5
USERS tablespace
described, 9-2

V

VIP, 1-4
virtual IP.*See* VIP
voting disk
configuration of, 4-6
mirroring, 3-5
raw device for
on Linux, 3-8
voting disks
configuring multiple, xxii, 1-5

W

Welcome page
Oracle Universal Installer, 4-7, 5-5, 5-10
Windows
analogy for Oracle base directory, 2-29
wmem_default file, 2-25
wmem_default parameter
recommended value on Linux, 2-25
wmem_max file, 2-25
wmem_max parameter
recommended value on Linux, 2-25

X

X emulator
installing from, 2-4
X window system
enabling remote hosts, 2-4
xhost command, 2-4
xterm command, 2-4