Oracle® Real Application Clusters

Installation Guide

12*c* Release 2 (12.2) for Microsoft Windows

E85702-01
May 2017

Oracle Real Application Clusters Installation Guide, 12*c* Release 2 (12.2) for Microsoft Windows

E85702-01

Copyright © 2012, 2017, Oracle and/or its affiliates. All rights reserved.

Primary Authors: Subhash Chandra, Janet Stern, Aparna Kamath

Contributing Authors: Mark Bauer, Jonathan Creighton, Reema Khosla, Markus Michalewicz, Satish Panchumarthy, Douglas Williams

Contributors: David Austin, Hanlin Chien, Alexander Keh, Sumit Kumar, Kevin Jernigan, Philip Newlan, Jacqueline Sideri, Janelle Simmons, Khethavath Singh, Richard Strohm, Roy Swonger, Ara Shakian, James Williams, Jiangqi Yang

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Intended Audience xi
Documentation Accessibility xi
Java Access Bridge and Accessibility xi
Related Documents xii
Conventions xiii

Changes In This Release for Oracle Real Application Clusters

Changes In Oracle Database 12c Release 2 xiv
Changes In Oracle Database 12c Release 1 xvii

1 Oracle RAC Installation Checklist

1.1 Deployment Checklist for Oracle RAC Database 1-1
1.2 Server Hardware and Software Review Checklist for Oracle RAC Installation 1-2
1.3 Installer Planning Checklist for Oracle Database Installation 1-4
1.4 Upgrade Checklist for Oracle RAC 1-8

2 Installing Oracle RAC and Oracle RAC One Node

2.1 Deciding Between Multitenant Container Databases and Non-CDBs in Oracle RAC 2-2
2.2 Simplified Upgrade of TIMESTAMP WITH TIME ZONE Data 2-3
2.3 Overview of Installation Directories for Oracle RAC 2-3
2.3.1 Overview of Oracle Base Directories 2-4
2.3.2 Overview of Oracle Home Directories 2-5
2.4 Selecting Installation Options for Oracle RAC 2-5
2.4.1 Selecting a Security Notification Contact 2-6
2.4.2 Selecting an Installation Option 2-7
2.4.3 Selecting the Database Type for Oracle Grid Infrastructure Deployments 2-7
2.4.4 Choosing the Cluster Database Management Type 2-8
2.4.5 Selecting an Installation Type 2-8

2.4.5.1 Preconfigured Database Types Supplied with Oracle Database 2-9
2.4.5.2 Using Advanced Database Configuration 2-9
2.4.5.3 About Installing Oracle Database with Other Languages 2-9
2.4.6 Selecting an Oracle Database Configuration Type 2-10
2.4.7 Selecting a Database Name 2-10
2.4.8 Requirements for Database Passwords 2-12
2.4.9 About Automatic Memory Management Installation Options 2-12
2.4.10 About Character Set Selection During Installation 2-12
2.4.11 Managing Database Services After Installation 2-13
2.5 Specify Oracle Home User Name and Password 2-14
2.6 Running Oracle Universal Installer to Install Oracle Database 2-15
2.6.1 Using Oracle Universal Installer to Install Oracle RAC 2-15
2.6.2 Using Oracle Universal Installer to Install Oracle RAC One Node 2-16
2.7 Installing the Oracle Database Vault Option 2-17
2.7.1 Starting the Listener with Oracle Database Vault Installations 2-17
2.7.2 Configuring Oracle Database Vault Using DBCA 2-17
2.7.3 Perform Postinstallation Configuration for Oracle Database Vault 2-18
2.8 Updating Environment Variables on Remote Nodes 2-18
2.9 Creating an Oracle RAC Database on Direct NFS 2-19
2.9.1 Perform a Software-Only Installation of Oracle Database 2-19
2.9.2 Use ASMCA to Configure an ACFS Mount Point 2-20
2.9.3 Use DBCA to Create and Configure the Oracle RAC Database 2-20
2.9.4 Enable and Configure Direct NFS 2-22
2.9.5 Use ASMCA to Remove the ACFS Mount Point 2-22

3 Creating Oracle RAC or Oracle RAC One Node Databases with DBCA

3.1 Using DBCA with Oracle RAC or Oracle RAC One Node 3-1
3.2 Benefits of Using Database Configuration Assistant 3-2
3.3 Automatic Listener Migration from Earlier Releases 3-3
3.4 Verifying Requirements for DBCA 3-3
3.5 Tasks to Complete Before Using DBCA to Create an Oracle RAC Database 3-4
3.5.1 Decide on a Naming Convention to Use for Your Oracle RAC Database 3-4
3.5.2 Configure Shared Storage for the Oracle RAC Database 3-5
3.5.3 Obtain the Password for the Oracle Home User 3-6
3.6 Selecting DBCA Options to Create an Oracle RAC or Oracle RAC One Node Database 3-7
3.6.1 Starting DBCA 3-7
3.6.2 Cluster Detection and Node Selection when Using DBCA 3-8
3.6.3 Using DBCA to Select Storage to Use With Oracle RAC Database 3-9

3.6.4 Using DBCA to Specify Database Initialization Parameters for Oracle RAC 3-9
3.6.5 Actions Performed By DBCA for Oracle RAC Databases 3-9
3.7 Using DBCA to Create an Oracle RAC One Node Database 3-10
3.8 Deleting an Oracle RAC Database Using DBCA 3-11

4 Oracle Real Application Clusters Postinstallation Procedures

4.1 Required Postinstallation Tasks 4-1
4.1.1 Determine If Any Patches Are Required For Your New Software 4-2
4.1.2 Configure Exceptions for the Windows Firewall 4-2
4.1.3 Create the OraMTS Service for Microsoft Transaction Server 4-2
4.1.4 Recompiling All Invalid Objects 4-3
4.1.5 Configuring Services on Oracle RAC with CDBs 4-3
4.2 Recommended Postinstallation Tasks 4-4
4.2.1 Setting Up Additional User Accounts 4-4
4.2.2 Setting the Oracle User Environment Variables 4-4
4.2.3 Downloading and Installing the ORAchk Health Check Tool 4-5
4.2.4 About Using CVU Cluster Healthchecks After Installation 4-5
4.3 Product-Specific Postinstallation Tasks 4-7
4.3.1 Configuring Oracle Database Vault 4-8
4.3.2 Configuring Oracle Label Security 4-8
4.3.3 Configuring the OraClrAgnt Service for Oracle Database Extensions for .NET 4-9
4.3.4 Configuring Oracle XML DB 4-9
4.3.5 Configure Storage for External Tables, Shared Files, or Directory Objects 4-10
4.4 Configuring the Oracle Home User 4-10
4.5 Oracle Configuration Manager Postinstallation Configuration for Oracle RAC 4-11
4.6 Enabling and Disabling Database Options After Installation 4-12

5 Using Server Pools with Oracle RAC

5.1 Policy-Managed Clusters and Capacity Management 5-1
5.1.1 Server Pools and Server Categorization 5-2
5.1.2 Server Pools and Policy-Based Management 5-2
5.1.3 How Server Pools Work 5-3
5.1.4 Default Server Pools 5-3
5.1.4.1 The Free Server Pool 5-3
5.1.4.2 The Generic Server Pool 5-3
5.2 Oracle RAC Database and Server Pools 5-4
5.3 About Creating Server Pools for Oracle RAC Databases 5-5

5.4 Oracle RAC One Node and Server Pools 5-6

6 Understanding the Oracle RAC Installed Configuration

6.1 Understanding the Configured Environment in Oracle RAC 6-2
6.2 Understanding Operating System Privileges Groups 6-2
6.3 Understanding Time Zone Settings on Cluster Nodes 6-3
6.4 Understanding the Server Parameter File for Oracle RAC 6-3
6.5 Multiple Oracle Home Directories on Windows 6-4
6.5.1 Changing the Current Setting for Oracle Home 6-4
6.6 About Pluggable Databases in Oracle RAC 6-5
6.7 Database Components Created Using Database Configuration Assistant 6-5
6.7.1 About Tablespaces and Data Files 6-6
6.7.2 About Control Files 6-6
6.7.3 About Online Redo Log Files 6-7
6.8 About Managing Undo Tablespaces in Oracle RAC 6-7
6.9 About Initialization Parameter Files 6-8
6.10 Oracle Net Services Configuration for Oracle RAC Databases 6-8
6.10.1 Database Services for an Oracle RAC Database 6-9
6.10.2 Naming Methods and Connect Descriptors 6-9
6.10.3 Easy Connect Naming Method 6-9
6.10.4 Understanding SCANs 6-10
6.10.4.1 About the SCAN 6-10
6.10.4.2 About SCAN VIP Addresses 6-11
6.10.4.3 About SCAN Listeners 6-12
6.10.5 About Connecting to an Oracle RAC Database Using SCANs 6-12
6.10.6 About Listener Configuration for an Oracle RAC Database 6-13
6.10.7 About Service Registration for an Oracle RAC Database 6-14
6.10.8 How Database Connections are Created When Using SCANs 6-15
6.11 Performance Features of Oracle Net Services and Oracle RAC 6-16
6.11.1 Load Balancing of Connections to Oracle RAC Databases 6-17
6.11.2 Connection Failover for Oracle RAC Databases 6-17
6.11.3 Shared Server Configuration for an Oracle RAC Database 6-17
6.12 Oracle Net Services Configuration Files and Parameters 6-18
6.12.1 Database Initialization Parameters for Database Service Registration 6-18
6.12.2 Net Service Names and the tnsnames.ora File 6-19
6.12.3 Net Service Names Created by DBCA 6-20
6.12.3.1 Net Service Names for Database Connections 6-20
6.12.3.2 Net Service Names for Instance Connections 6-21
6.12.4 Listener Configuration and the listener.ora File 6-21
6.12.4.1 Local Listener for an Oracle RAC Database 6-22

6.12.4.2 Remote Listeners for an Oracle RAC Database 6-22
6.12.4.3 Managing Multiple Listeners for an Oracle RAC Database 6-23
6.12.4.4 How Oracle Database Uses the Listener File (listener.ora) 6-23
6.12.5 Net Services Profile File (sqlnet.ora) 6-24

7 Removing Oracle Real Application Clusters Software

7.1 Overview of Deinstallation Procedures 7-1
7.2 Identifying All Instances On a Cluster 7-2
7.2.1 Identifying All Instances On a Cluster Using SRVCTL 7-2
7.2.2 Identifying All Instances On a Cluster Using the Windows Services Control Manager 7-3
7.3 Removing Oracle RAC Software 7-3
7.3.1 About the Deinstallation Tool 7-3
7.3.2 Deinstallation Tool Command Reference 7-4
7.3.3 Using the Deinstallation Tool to Remove Oracle RAC 7-7
7.3.3.1 Running the Deinstallation Tool From an Oracle Home 7-7
7.3.3.2 Running the Deinstallation Tool Interactively From the Installer 7-8
7.3.3.3 Running the Deinstallation Tool with a Response File 7-9
7.3.3.4 Generating a Response File For Use With the Deinstallation Tool 7-9
7.4 Cleaning Up After a Failed Installation 7-9

A Using Scripts or Response Files to Create Oracle RAC Databases

A.1 Using DBCA to Generate Installation Scripts for Oracle RAC A-2
A.2 About DBCA Noninteractive (Silent) Configuration for Oracle RAC A-3
A.3 Using DBCA Commands for Noninteractive (Silent) Configuration of Oracle RAC A-3
A.4 How Response Files Work A-4
A.4.1 Deciding to Use Silent Mode or Response File Mode A-5
A.4.2 Creating a Database Using Oracle ASM for Database Files in Silent Mode A-6
A.4.3 Using Response Files A-7
A.5 Preparing Response Files A-7
A.5.1 About Response File Templates A-8
A.5.2 Editing a Response File Template A-9
A.5.3 Recording Response Files A-9
A.6 Running Oracle Universal Installer Using a Response File A-11
A.7 Running Configuration Assistants Using Response Files A-12
A.7.1 Silent Mode of Database Configuration Assistant A-12
A.7.2 Running Database Configuration Assistant in Response File Mode A-12
A.7.3 Running Oracle Net Configuration Assistant Using Response Files A-14

A.8 Postinstallation Configuration Using Response File Created During Installation A-15
A.8.1 Using the Installation Response File for Postinstallation Configuration A-15
A.8.2 Running Postinstallation Configuration Using Response File A-16
A.9 Postinstallation Configuration Using the ConfigToolAllCommands Script A-17
A.9.1 About the Postinstallation Configuration File A-18
A.9.2 Creating a Password Response File A-18
A.9.3 Performing Postinstallation Configuration Using a Response File A-19

B Directory Structure for Oracle RAC Environments

B.1 Understanding the Oracle RAC Directory Structure B-1
B.2 Directory Structures for Oracle RAC B-1

C Preparing to Upgrade an Existing Oracle RAC Database

C.1 Backing Up the Oracle RAC Database C-1
C.2 Using CVU to Validate Readiness for Oracle RAC Upgrades C-2
C.2.1 Using the CVU Database Upgrade Validation Command Options C-2
C.2.2 Example of Verifying System Upgrade Readiness for Oracle RAC Infrastructure C-3
C.2.3 Verifying System Readiness for Oracle Database Upgrades C-3

D Managing Oracle Database Port Numbers

D.1 About Managing Ports D-1
D.2 About Viewing Port Numbers and Access URLS D-2
D.3 Setting UDP and TCP Dynamic Port Range for Oracle RAC Installations D-2
D.4 Port Numbers and Protocols of Oracle Components D-2
D.5 Changing the Oracle Services for Microsoft Transaction Server Port D-6

Index

List of Examples

2-1	Global Database Name and Related Initialization Parameters	2-11
2-2	DB_UNIQUE_NAME and Related ORACLE_SID Values	2-11
3-1	Using CVU Prior to Verify Your System is Prepared for an Oracle RAC Installation	3-4
3-2	Performing Clusterware Diagnostics If DCBA Fails To Detect A Two-Node Cluster	3-9
4-1	Running a Cluster Healthcheck After the Software Installation	4-7
4-2	Running a Healthcheck for Oracle RAC Database	4-7
4-3	Running the Chopt Tool	4-12
6-1	Connecting to Oracle RAC Using an Oracle Net Connect Descriptor	6-13
6-2	Adding a Second Listener to an Oracle RAC Database	6-20
6-3	Net Service Name Entry for a Database Connection	6-21
6-4	Example Net Service Name Entry for an Instance Connection	6-21
6-5	Example listener.ora File for an Oracle RAC Node	6-24
7-1	Running deinstall.bat From Within the Oracle Home	7-7
7-2	Running the Deinstallation Tool from the Software Installation Media	7-8
A-1	Using DBCA in Silent Mode to Create an Oracle RAC Database	A-3
A-2	Response File Passwords for Oracle Grid Infrastructure	A-15
A-3	Response File Passwords for Oracle Grid Infrastructure for a Standalone Server (Oracle Restart)	A-16
A-4	Response File Passwords for Oracle Database	A-16
A-5	Sample Password Response File for Oracle RAC Databases	A-19
A-6	Running Configuration Assistants in Response File Mode	A-19

List of Tables

1-1	Deployment Checklist for Oracle RAC Database	1-1
1-2	Hardware and Software Checklist for Oracle RAC	1-2
1-3	Supported Storage Options for Oracle Database and Recovery Files	1-4
1-4	Oracle Universal Installer Planning Checklist for Oracle Database Installation	1-5
1-5	Upgrade Checklist for Oracle RAC	1-8
4-1	Configuration Options and Requirements for Oracle Label Security	4-8
4-2	Database Options for Chopt Tool Command	4-12
6-1	Tablespace Names Used with Oracle Real Application Clusters Databases	6-6
7-1	Options for the Deinstallation Tool	7-5
A-1	Reasons for Using Silent Mode or Response File Mode	A-5
A-2	Response Files for Oracle Database and Oracle Grid Infrastructure	A-8
B-1	Directory Structure for a Sample OFA-Compliant Environment	B-1
C-1	Command Options for CLUVFY Pre-upgrade Check	C-2
D-1	Ports Used in Oracle Components	D-2

Preface

This guide explains how to install and configure Oracle Real Application Clusters (Oracle RAC).

Before you use this guide, you must first complete an installation of Oracle Clusterware, as described in the *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*.

Topics:

- Intended Audience
- Documentation Accessibility
- Java Access Bridge and Accessibility
- Related Documents
- Conventions

Intended Audience

Oracle Real Application Clusters Installation Guide for Microsoft Windows provides database installation information for database administrators (DBAs) who install and configure Oracle RAC.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Java Access Bridge and Accessibility

Java Access Bridge enables assistive technologies to read Java applications running on the Windows platform. Assistive technologies can read Java-based interfaces, such as Oracle Universal Installer and Oracle Enterprise Manager Database Express.

> **See Also:**
>
> *Oracle Database Installation Guide for Microsoft Windows* for more information about installing Java Access Bridge

Related Documents

The related documentation for Oracle Database products includes the following manuals:

Installation Guides

- *Oracle Grid Infrastructure Installation Guide* for your platform
- *Oracle Database Installation Guide* for your platform
- *Oracle Universal Installer (OUI) User's Guide*
- *Oracle Database Client Installation Guide for Microsoft Windows*
- *Oracle Database Examples Installation Guide*
- *Oracle Application Express Installation Guide*
- *Oracle Database SQL Developer Installation Guide*

Operating System-Specific Administrative Guides

- *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems*
- *Oracle Database Platform Guide for Microsoft Windows*

Oracle Real Application Clusters Management

- *Oracle Clusterware Administration and Deployment Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Database 2 Day DBA*
- *Oracle Database 2 Day + Real Application Clusters Guide*

Generic Documentation

- *Oracle Database New Features Guide*
- *Oracle Database Concepts*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Database Reference*
- *Oracle Database Sample Schemas* describes the sample schemas provided for Oracle Database. Many of the examples in the Oracle Database Documentation Library use these schemas.

Error Messages

Oracle error message documentation is available only in HTML. You can browse the error messages by range in *Oracle Database Error Messages Reference*. 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.

Other Documentation

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 website:

http://www.oracle.com/technetwork/community/join/overview/index.html

For any other documentation, you can access the documentation library at Oracle Help Center:

http://docs.oracle.com

Conventions

The following text conventions are used in this document:

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

Changes In This Release for Oracle Real Application Clusters

This preface lists changes in Oracle Real Application Clusters Installation Guide

Topics:

- Changes In Oracle Database 12c Release 2
- Changes In Oracle Database 12c Release 1

Changes In Oracle Database 12*c* Release 2

The following are changes in Oracle Real Application Clusters Installation Guide for Oracle Database 12c Release 2.

- New Features for Oracle Real Application Clusters 12c Release 2 (12.2.0.1)
- Deprecated Features
- Desupported Features

New Features for Oracle Real Application Clusters 12*c* Release 2 (12.2.0.1)

The following features are new in this release.

- Parallel NFS Support in Oracle Direct NFS Client

  Starting with Oracle Grid Infrastructure 12*c* release 2 (12.2), Oracle Direct NFS Client supports parallel NFS. Parallel NFS is an NFS v4.1 option that allows direct client access to file servers, enabling scalable distributed storage.

  See *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*.

- Direct NFS Dispatcher Support

  Starting with Oracle Grid Infrastructure 12*c* release 2 (12.2), Oracle Direct NFS Client supports adding a dispatcher or I/O slave infrastructure. For very large database deployments running Oracle Direct NFS Client, this feature facilitates scaling of sockets and TCP connections to multi-path and clustered NFS storage.

  See *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*.

- Windows Direct NFS Client Supports All Widely Accepted NFS Path Format

  Previously, Oracle's Windows DNFS Client could only use Windows-specific NFS paths. This new feature enhances the DNFS client so that it can support all widely accepted NFS path formats, including both Windows-style and UNIX-style NFS

paths. Administrators now have the flexibility to use the NFS path format they prefer.

See *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*.

- Kerberos Authentication for Direct NFS

  Oracle Database now supports Kerberos implementation with Direct NFS communication. This feature solves the problem of authentication, message integrity, and optional encryption over unsecured networks for data exchange between Oracle Database and NFS servers using Direct NFS protocols.

  See *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*.

- Separation of Duty for Administering Oracle Real Application Clusters

  Starting with Oracle Database 12*c* release 2 (12.2), Oracle Database provides support for separation of duty best practices when administering Oracle Real Application Clusters (Oracle RAC) by introducing the SYSRAC administrative privilege for the clusterware agent. This feature removes the need to use the powerful SYSDBA administrative privilege for Oracle RAC.

  SYSRAC, like SYSDG, SYSBACKUP and SYSKM, helps enforce separation of duties and reduce reliance on the use of SYSDBA on production systems. This administrative privilege is the default mode for connecting to the database by the clusterware agent on behalf of the Oracle RAC utilities such as `srvctl`.

- AL32UTF8 As the Default Database Character Set

  Starting with Oracle Database 12*c* release 2 (12.2), the default database character set of a database created from the General Purpose/Transaction Processing or the Data Warehousing template is Unicode AL32UTF8. Oracle recommends that you use Unicode AL32UTF8 as the database character set.

  For more information, see About Character Set Selection During Installation.

- Support for UDP on Windows Operating System

  Starting with Oracle Database 12*c* Release 2 (12.2), the User Datagram Protocol (UDP) is supported on Windows. The UDP protocol allows for larger clusters.

  For more information, see Port Numbers and Protocols of Oracle Components.

- Support for Windows Group Managed Service Accounts (MSAs)

  Group MSAs are a type of account introduced by Microsoft in Windows Server 2008 R2 to overcome the password management problems encountered when authenticating services for domain user accounts. You can use Group MSAs for the Oracle Home user when installing Oracle RAC on Windows.

  See Server Hardware and Software Review Checklist for Oracle RAC Installation.

- Postinstallation Configuration of Oracle Software using the `-executeConfigTools` option

  Starting with Oracle Database 12*c* release 2 (12.2), you can perform postinstallation configuration of Oracle products by running the Oracle Database or Oracle Grid Infrastructure installer with the `-executeConfigTools` option. You can use the same response file created during installation to complete postinstallation configuration.

See Postinstallation Configuration Using Response File Created During Installation.

- SCAN Listener Supports HTTP Protocol

  Starting with Oracle Database 12*c* release 2 (12.2), SCAN listener enables connections for the recovery server coming over HTTP to be redirected to different machines based on the load on the recovery server machines.

  See About SCAN Listeners.

Deprecated Features

The following features are deprecated in this release, and may be desupported in another release. See *Oracle Database Upgrade Guide* for a complete list of deprecated features in this release.

- Deprecation of `configToolAllCommands` script

  The `configToolAllCommands` script runs in the response file mode to configure Oracle products after installation and uses a separate password response file. Starting with Oracle Database 12*c* Release 2 (12.2), the `configToolAllCommands` script is deprecated and is subject to desupport in a future release.

  To perform postinstallation configuration of Oracle products, you can now run the Oracle Database or Oracle Grid Infrastructure installer with the `-executeConfigTools` option. You can use the same response file created during installation to complete postinstallation configuration.

- Deprecation of Oracle Administration Assistant for Windows

  Oracle Administration Assistant for Windows is a tool for creating database administrators, operators, users, and roles in Windows. It also allows database service, startup/shutdown configuration, and Windows Registry parameter management. Oracle Administration Assistant for Windows is deprecated in Oracle Database 12*c* Release 2 (12.2). Instead of using Oracle Administration Assistant for Windows, use native Windows administration tools.

Desupported Features

The following feature is desupported in this release. See *Oracle Database Upgrade Guide* for a complete list of desupported features in this release.

- Desupport of Direct File System Placement for Oracle Cluster Registry (OCR) and Voting Files

  Starting with Oracle Grid Infrastructure 12*c* Release 2 (12.2), the placement of Oracle Clusterware files (the Oracle Cluster Registry (OCR), and the Voting Files) directly on a shared file system is desupported. The Oracle Clusterware files are now managed by Oracle Automatic Storage Management (Oracle ASM). You cannot place Oracle Clusterware files directly on a shared file system. If you need to use a supported shared file system, either a Network File System, or a shared cluster file system instead of native disks devices, then you must create Oracle ASM disks on the shared file systems that you plan to use for hosting Oracle Clusterware files before installing Oracle Grid Infrastructure. You can then use the Oracle ASM disks in an Oracle ASM disk group to manage Oracle Clusterware files.

Changes In Oracle Database 12*c* Release 1

The following are changes in Oracle Real Application Clusters Installation Guide for Oracle Database 12*c* Release 1.

- New Features for Oracle Real Application Clusters 12c Release 1 (12.1.0.2)
- New Features for Oracle Real Application Clusters 12c Release 1 (12.1.0.1)
- Deprecated Features
- Desupported Features
- Other Changes

New Features for Oracle Real Application Clusters 12*c* Release 1 (12.1.0.2)

The following features are new in this release.

- Standard Edition Replacement

  Starting with Oracle Database 12c Release 1(12.1.0.2), Standard Edition 2 (SE2) replaces Standard Edition. Standard Edition 2 can run on systems with up to two sockets and can support a two-node Oracle RAC cluster.

  > **See Also:**
  >
  > My Oracle Support Note 2027072.1 for more information:

  https://support.oracle.com/rs?type=doc&id=2027072.1

- Oracle Clusterware and Oracle RAC Support for the Diagnosability Framework

  The Diagnosability Framework enables Oracle products to use a standardized and simplified way of storing and analyzing diagnosability data.

  Note:

  The Oracle Trace File Analyzer (TFA) Collector is not supported on Windows operating systems.

  See *Oracle Real Application Clusters Administration and Deployment Guide* for information about locating Oracle RAC trace files.

- Automatic Installation of Grid Infrastructure Management Repository

  The Grid Infrastructure Management Repository is automatically installed with Oracle Grid Infrastructure 12*c* Release 1 (12.1.0.2).

- IPv6 Support for Public Networks

  Oracle RAC 12*c* Release 1 (12.1.0.2) on Windows supports IPv6-based public IP and VIP addresses.

  IPv6-based IP addresses have become the latest standard for the information technology infrastructure in today's data centers. With this release, Oracle RAC and Oracle Grid Infrastructure support this standard. You can configure cluster

nodes during installation with either IPv4 or IPv6 addresses on the same network. Database clients can connect to either IPv4 or IPv6 addresses. The Single Client Access Name (SCAN) listener automatically redirects client connects to the appropriate database listener for the IP protocol of the client request.

See *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)*.

New Features for Oracle Real Application Clusters 12*c* Release 1 (12.1.0.1)

The following features are new in this release.

- Database Upgrade Automation Using DBUA

  There are three areas that are being enhanced for upgrade ease-of-use. First, in the pre-upgrade phase, the existing manual steps are eliminated and give more explicit advice or even generate a fixup script to find issues identified in the pre-upgrade phase. Second, in the post-upgrade phase, there is a post-upgrade health check that indicates that the upgrade was successful. Finally, partner documents (such as SAP) and major customer upgrade documents are used to further identify manual steps that may be automated and generalized to a wider customer base.

  Automating the upgrade process provides major improvements in usability and ease-of-use. There is also better integration of database upgrade with Oracle Grid Infrastructure for a cluster and Oracle Enterprise Manager Cloud Control.

  See *Oracle Database Upgrade Guide*.

- Oracle Home User Support for Oracle RAC on Windows

  Starting with Oracle Database 12*c* Release 1 (12.1), Oracle Database supports the use of an Oracle Home User, which can be specified at installation time. The Oracle Home User is associated with a Windows domain user. The Windows domain user should be a standard (non-Administrator) account to ensure that the Oracle Home User has a limited set of privileges, thus ensuring that the Oracle Database services have only those privileges required to run Oracle products.

  Windows Administrator user privileges are still required to perform Oracle software maintenance tasks including install, upgrade, patching and so on. Oracle Database administrative tools have been enhanced to ask for the password of the Oracle Home User, if needed. In Oracle RAC environments, you can store the password for the Oracle Home User in a secure wallet. If such a wallet exists, then the Oracle Database administrative tools automatically use the password from the wallet and do not require the user to enter the password for the Oracle Home User.

  See *Oracle Database Platform Guide for Microsoft Windows*.

- Oracle Home User Control Utility

  This release introduces a new Windows utility called Oracle Home User Control. This is a command line tool that displays the Oracle Home User name associated with the current Oracle Home. This utility also enables you to modify the Windows Services used by Oracle to use a new password when the password for Oracle Home User is changed.

- Oracle ACFS Auditing and Support for Importing Auditing Data into Oracle Audit Vault Server

This feature provides auditing for Oracle ACFS security and encryption. In addition, this feature also generates an XML file containing Oracle ACFS audit trail data which can be imported by Oracle Audit Vault Server.

See *Oracle Automatic Storage Management Administrator's Guide*.

- DBCA Support for Multitenant Container Database and Pluggable Database Configurations

  Starting with Oracle Database 12*c* Release 1 (12.1), Oracle Database Configuration Assistant (DBCA) allows you to create either a multitenant container database (CDB) or a non-CDB. You can create the CDB with zero, one, or more pluggable databases (PDBs).

  You can also create a CDB with one PDB during the database installation.

  See "Introduction to Pluggable Databases" in *Oracle Database Administrator's Guide*.

- Oracle Enterprise Manager Database Express 12*c*

  Oracle Database 12*c* introduces Oracle Enterprise Manager Database Express (Oracle EM Express), a web management product built into Oracle Database without any need for special installation or management. Using Oracle EM Express, you can perform administrative tasks such as managing user security, and managing database memory and storage. You can also view performance and status information about your database.

  See *Oracle Database 2 Day DBA* for information about Oracle Enterprise Manager Database Express.

- Policy-Based Cluster Management and Administration

  Oracle Grid Infrastructure allows running multiple applications in one cluster. Using a policy-based approach, the workload introduced by these applications can be allocated across the cluster using a policy. In addition, a policy set enables different policies to be applied to the cluster over time as required. Policy sets can be defined using a web-based interface or a command-line interface.

  Hosting various workloads in the same cluster helps to consolidate the workloads into a shared infrastructure that provides high availability and scalability. Using a centralized policy-based approach allows for dynamic resource reallocation and prioritization as the demand changes.

  See *Oracle Clusterware Administration and Deployment Guide*.

- Simplified Oracle Database Vault Installation

  Oracle Database Vault installs by default as part of the Oracle Database installation. You can also configure, enable, or disable Oracle Database Vault after completing the Oracle Database installation, either by using Oracle DBCA, or by running SQL statements.

  See Installing the Oracle Database Vault Option for more information about Oracle Database Vault installation.

- Simplified Oracle Label Security Installation

  Oracle Label Security installs by default as part of the Oracle Database installation. You can also configure Oracle Label Security after the Oracle Database installation using Oracle DBCA.

  See *Oracle Label Security Administrator's Guide*.

- Support for Separation of Database Administration Duties

Oracle Database 12*c* Release 1 (12.1) provides support for separation of administrative duties for Oracle Database by introducing task-specific and least-privileged administrative privileges that do not require the SYSDBA administrative privilege. These new privileges are: SYSBACKUP for backup and recovery, SYSDG for Oracle Data Guard, and SYSKM for encryption key management.

See *Oracle Grid Infrastructure Installation Guide* for your platform and "Managing Administrative Privileges" in *Oracle Database Security Guide*.

- Unified Database Audit Configuration

  Starting with Oracle Database 12*c*, you can create named audit policies. An audit policy contains a set of audit options, which is stored in the database as an object. The advantage of creating a named audit policy is that it reduces the number of commands that are required to create a database audit policy, and it simplifies the implementation of an audit configuration for security and compliance with conditional auditing. This new audit policy framework is included with the database installation.

  See "Auditing Activities with Unified Audit Policies and the AUDIT Statement" in *Oracle Database Security Guide*.

Deprecated Features

The following features are deprecated in this release, and may be desupported in another release. See *Oracle Database Upgrade Guide* for a complete list of deprecated features in this release.

- Single-letter SRVCTL command-line interface (CLI) options

  All SRVCTL commands have been enhanced to accept full-word options instead of the single-letter options. All new SRVCTL command options added in this release support full-word options, only, and do not have single-letter equivalents. The use of single-letter options with SRVCTL commands might be desupported in a future release.

- Standalone Deinstallation Tool

  The deinstallation tool is now integrated with the database installation software.

  See "About the Deinstallation Tool"

- The `-cleanupOBase` flag of the deinstallation tool

  The `-cleanupOBase` flag of the deinstallation tool is deprecated in this release. There is no replacement for this flag.

- Oracle Enterprise Manager Database Express replaces Oracle Enterprise Manager Database Control.
- A deinstall option using Oracle Universal Installer (OUI) replaces the standalone deinstall utility.

Desupported Features

The following features are no longer supported by Oracle. See *Oracle Database Upgrade Guide* for a complete list of desupported features in this release.

- Oracle Enterprise Manager Database Control
- `CLEANUP_ORACLE_BASE` property removed

- Deinstall standalone utility
- Oracle Counters for Windows Performance Monitor
- Oracle Cluster File System for Windows (OCFS for Windows)

Other Changes

This topic lists the other changes in Oracle Database 12*c* Release 1 relevant to installing Oracle Real Application Clusters.

- LREG (listener registration process) replaces the PMON (process monitor) for notifying listeners about instances, services, handlers and endpoints.
- Oracle RAC is supported on Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2.

1
Oracle RAC Installation Checklist

Review these checklists for installing Oracle Real Application Clusters (Oracle RAC).

Topics:

- Deployment Checklist for Oracle RAC Database
  Use the checklist to review the deployment methods for Oracle Real Application Clusters.
- Server Hardware and Software Review Checklist for Oracle RAC Installation
  Use the checklist to check minimum hardware and software requirements for Oracle RAC.
- Installer Planning Checklist for Oracle Database Installation
  Use the checklist to assist you to be prepared before starting Oracle Universal Installer.
- Upgrade Checklist for Oracle RAC
  Review the checklist for additional requirements related to upgrading an existing Oracle Real Application Clusters (Oracle RAC) installation to Oracle Database 12*c* Release 2 (12.2).

1.1 Deployment Checklist for Oracle RAC Database

Use the checklist to review the deployment methods for Oracle Real Application Clusters.

Table 1-1 Deployment Checklist for Oracle RAC Database

Item	Task
To Deploy Oracle RAC or Oracle RAC One Node software	Install Oracle RAC or Oracle RAC One Node Database software using Oracle Universal Installer (OUI). See Using Oracle Universal Installer to Install Oracle RAC or Using Oracle Universal Installer to Install Oracle RAC One Node for the procedure to install Oracle RAC using OUI.
To Deploy Oracle Database software and create Oracle RAC or Oracle RAC One Node databases	Install Oracle RAC or Oracle RAC One Node Database software using Oracle Universal Installer (OUI) and choose to create a database.. See Using Oracle Universal Installer to Install Oracle RAC or Using Oracle Universal Installer to Install Oracle RAC One Node for the procedure to install Oracle RAC using OUI.
To Create Oracle RAC database in an already-installed Oracle home.	Use Oracle Database Configuration Assistant (DBCA). See Creating Oracle RAC or Oracle RAC One Node Databases with DBCA for more information about creating database using DBCA.
Complete the installation by completing the post-installation tasks	See Oracle Real Application Clusters Postinstallation Procedures.

1.2 Server Hardware and Software Review Checklist for Oracle RAC Installation

Use the checklist to check minimum hardware and software requirements for Oracle RAC.

It is assumed that the servers were prepared as described in the Oracle Grid Infrastructure Installation Guide for your platform.

Table 1-2 Hardware and Software Checklist for Oracle RAC

Item	Task
Server Hardware on each node	• Use identical server hardware on each node, to simplify server maintenance. • Avoiding resource contention issues by not installing Oracle RAC on a primary domain controller or backup domain controller. • Review "Checking the Hardware Requirements" in *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)* to ensure that your system has enough RAM. • Verify the `TEMP` environment variable points to a location that has enough available space for the installation. • For both the Enterprise and Standard Editions of Oracle RAC, the hard disk requirements for Oracle Database components include 32 MB required to install Java Runtime Environment (JRE) and Oracle Universal Installer (OUI) on the disk partition where the operating system is installed. If sufficient space is not detected, then the installation fails and an error message appears.
Operating System General Requirements	• Install the supported operating system version and install the operating system packages and patches required for that version. For some operating systems, Oracle may require updates, such as service packs and individual patches. If such requirements exist, then they are stated in the Release Notes for a particular release. You can also apply other operating system patches as recommended by Microsoft, if there are no "certification exceptions" listed in the Release Notes. Refer to your operating system vendor for required operating system updates. **Note:** You must use the same operating system on each node in the cluster. Oracle strongly recommends that you use the same software configurations on each node of the cluster. Oracle Clusterware and Oracle RAC do not support heterogeneous platforms (each server must run the same Oracle software binaries) in the same cluster. • (Optional) Stage all of the software on one node for installation (the "local node").

Table 1-2 (Cont.) Hardware and Software Checklist for Oracle RAC

Item	Task
Create the required users and configure the environments	• Review the users created during the Oracle Grid Infrastructure installation. • To install the Oracle Real Application Clusters software, you must use either a local or domain user. In either case, the **Oracle Installation user** must be an explicit member of the Administrators group on all nodes of the cluster. • To install Oracle RAC software on Oracle Automatic Storage Management Cluster File System (Oracle ACFS) you must use a Windows Domain user account for the Oracle Home user. • You can use a Local User to perform the installation provided the user has the same password on all nodes and is an explicit member of the Administrators group on all nodes of the cluster. • During installation, you can specify an **Oracle Home user**. The Oracle Home User may be a Windows Domain User Account or a Windows Group Managed service account (GMSA).

Configure the network interfaces	• Verify that each node in your cluster can communicate with the other nodes using the net use command, for example, on `node1` you can use the following command:
	`C:\> net use \\node2\c$`
	`The command completed successfully.`
	• Set up the domain name forwarding for Grid Naming Service (GNS) if you plan to deploy GNS or Multi-Cluster GNS, and set up the network addresses in the DNS and on the server as needed.

Set up the required shared storage.	• All Oracle RAC database instances share the control file, server parameter file (SPFILE), redo log files, and all data files. These files must be placed on shared storage, and all the cluster database instances on cluster nodes must have access to these files. 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 recommends that you choose Oracle ASM as the shared storage option for database and recovery files. • You can store shared files using Oracle ASM, Oracle ACFS, or on a Network File Server (NFS) using the Direct NFS client (DNFS). • Files supported by Oracle ACFS include database and application executable files, trace files, alert logs, application reports, BFILEs, and configuration files. • For Standard Edition and Standard Edition 2 (SE2) Oracle RAC installations, Oracle ASM is the only supported shared storage option for database or recovery files. • If you do not have a storage option that provides external file redundancy, then you must configure at least three voting file areas to provide voting file redundancy. • Processing Node instances access storage from Core Nodes that have ASM I/O Server running on them to provide storage services.

Table 1-2 (Cont.) Hardware and Software Checklist for Oracle RAC

Item	Task
Time Zone Requirement	Upgrade the Time Zone File and TSTZ Data. As part of an installation of Oracle Database 12*c* Release 2 (12.2), time zone version files from 1 to 12 are installed in the path `%ORACLE_HOME%\oracore\zoneinfo/`. You can continue to use the current time zone version or upgrade to the latest version. Oracle recommends that you upgrade the server to the latest time zone version. See *Oracle Database Globalization Support Guide* for information about how to upgrade the time zone file and TSTZ data
Platform-Specific Server Configuration	• Ensure an external jobs user is created for HP-UX. • Configure a Windows Domain user account to use when installing Oracle RAC on Oracle ACFS on Windows platforms.

Table 1-3 Supported Storage Options for Oracle Database and Recovery Files

Storage Option	File Types Supported / Database	File Types Supported / Recovery Area
Oracle ASM	Yes	Yes
Oracle ACFS	Yes (Oracle Database 12c Release 12.1 and later) except on Windows platforms	Yes
Direct NFS Client	Yes	Yes
Local Storage	No	No
Shared unformatted partitions	No	No

See Also:

Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit) for more information about users and groups.

1.3 Installer Planning Checklist for Oracle Database Installation

Use the checklist to assist you to be prepared before starting Oracle Universal Installer.

Table 1-4 Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task
Review the Documentation	• Review the Oracle Database Readme, which is available at the following location: *Oracle Database Readme* • Review the Oracle Database Release Notes for your platform, which are available for your release at the following URL: http://docs.oracle.com/en/database/database.html • Be familiar with the installation steps for Oracle RAC software and creating an Oracle RAC database.
Review the Licensing Information	You are permitted to use only those components in the Oracle Database media pack for which you have purchased licenses. For more information about licenses, refer to the following URL: *Oracle Database Licensing Information*
Obtain your My Oracle Support account information.	During installation, you require a My Oracle Support user name and password to configure security updates, download software updates, and other installation tasks. You can register for My Oracle Support at the following URL: https://support.oracle.com/
Review Oracle Support Certification Matrix	New platforms and operating system software versions can be certified after this guide is published, review the certification matrix on the My Oracle Support website for the most up-to-date list of certified hardware platforms and operating system versions: https://support.oracle.com/ You must register online before using My Oracle Support. After logging in, from the menu options, select the Certifications tab. On the Certifications page, use the Certification Search options to search by Product, Release, and Platform. You can also search using the Certification Quick Link options such as Product Delivery, and Lifetime Support.
Review online information to assist with installation	• Log on to My Oracle Support to access certifications for your installation for your platform. • Refer to Oracle.com (http://www.oracle.com) for additional resources about planning for specific implementation scenarios, best practices, and other information that can help you with your installation plan. In particular, refer to the Oracle Real Application Clusters pages on the Oracle Technology Network at http://www.oracle.com/goto/rac

Table 1-4 (Cont.) Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task
Run Oracle Universal Installer (OUI) with CVU and use fixup scripts	Oracle Universal Installer is fully integrated with Cluster Verification Utility (CVU), automating many CVU prerequisite checks. Oracle Universal Installer runs all prerequisite checks and creates fixup scripts when you run the installer. You can run OUI up to the Summary screen without starting the installation. • Obtain the latest version of CVU at the following URL: http://www.oracle.com/technetwork/database/options/clustering/downloads/cvu-download-homepage-099973.html • You can also run CVU commands manually to check that your system is prepared for installation before you start an Oracle RAC installation. If you have vendors performing hardware or operating system configuration steps, then ask the vendors to complete the relevant CVU checks of the cluster after they complete their work to ensure that your system is configured correctly. • Run OUI and DBCA from a node where an Oracle RAC Oracle database instance is located. • In case of an upgrade failure, follow common industry standards for data recovery planning, including backing up your existing database.

Download and run ORAchk for runtime and upgrade checks, or runtime health checks	The ORAchk utility provides system checks that can help to prevent issues before and after installation. These checks include kernel requirements, operating system resource allocations, and other system requirements.
	Use the ORAchk Upgrade Readiness Assessment to obtain an automated upgrade-specific system health check for upgrades to 11.2.0.3, 11.2.0.4, 12.1.0.1 and 12.2, For example:
	• Before you perform a fresh database installation:
	`./orachk -profile preinstall`
	• To upgrade your existing database to a higher version or release:
	`./orachk -o pre`
	The ORAchk Upgrade Readiness Assessment automates many of the manual pre- and post-upgrade checks described in Oracle upgrade documentation. ORAchk is supported on Windows 2008 and Windows 2012 on a Cygwin environment only. For more information refer to the following URL:
	https://support.oracle.com/rs?type=doc&id=1268927.1

Verify if Oracle Grid Infrastructure is installed	• To install Oracle RAC 12*c* Release 2 (12.2), you must have Oracle Grid Infrastructure (Oracle Clusterware and Oracle ASM) 12c release 2 (12.2) installed on your cluster. The Oracle Clusterware version must be equal to or greater than the Oracle RAC version that you plan to install. • Currently, there are no supported clusterware products other than Oracle Clusterware for the Microsoft Windows platforms. If you intend to install Oracle RAC, then you must first install Oracle Grid Infrastructure for a cluster, which includes Oracle Clusterware.

Table 1-4 (Cont.) Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task
Check running Oracle processes, and shut down if necessary	• On a standalone database not using Oracle ASM: You do not need to shut down the database while you install Oracle Grid Infrastructure. • On a standalone database using Oracle ASM: Stop the existing Oracle ASM instances. The Oracle ASM instances are restarted during installation. • On an Oracle RAC Database node: This installation requires an upgrade of Oracle Clusterware, as Oracle Clusterware is required to run Oracle RAC. As part of the upgrade, you must shut down the database one node at a time as the rolling upgrade proceeds from node to node.
Ensure Task Scheduler jobs do not run during installation	If the installer is running when daily scheduled jobs start, then you may encounter unexplained installation problems if your scheduled job is performing cleanup, and temporary files are deleted before the installation is finished. Oracle recommends that you complete installation before daily scheduled jobs are run, or disable daily scheduled jobs that perform cleanup until after the installation is completed.
Decide on an Oracle Database management tool	By default, Oracle Database 12*c* is managed by Oracle Enterprise Manager Database Express. If you have an existing Oracle Management Agent, and decide to use Oracle Enterprise Manager Cloud Control to centrally manage your database, then obtain the following information to enter during the database installation: • OMS host • OMS port • EM admin username • EM admin password • Specify password of ASMSNMP user You need a web browser to access documentation, to use Oracle Enterprise Manager Database Express, and to use Oracle Application Express. Web browsers must support JavaScript and the HTML 4.0 and Cascading Style Sheets (CSS) 1.0 standards.
Review memory allocation and Automatic Memory Management feature	You can enable automatic memory management either during, or after Oracle Database installation. If you enable automatic memory management after installation, then you must shut down and restart of the database. With Automatic Memory Management, Oracle Database instances automatically manage and tune memory. You choose a memory target, and the instance automatically distributes memory between the system global area (SGA) and the instance program global area (instance PGA). As memory requirements change, the instance dynamically redistributes memory between the SGA and instance PGA. For more information, refer to the following URL:

See Also:

- *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)* to check status of Oracle Clusterware components.
- *Oracle Enterprise Manager Cloud Control Administrator's Guide* for more information about Oracle Enterprise Manager Cloud Control.
- *Oracle Database Administrator's Guide* for more information about Automatic Memory Management.
- *Oracle Clusterware Administration and Deployment Guide* for more information about CVU.

1.4 Upgrade Checklist for Oracle RAC

Review the checklist for additional requirements related to upgrading an existing Oracle Real Application Clusters (Oracle RAC) installation to Oracle Database 12*c* Release 2 (12.2).

The system must meet the following requirements:

- Each server must use the same Windows user account as the Oracle Home User across all Oracle homes.
- When upgrading, if the Oracle home from which you run the database upgrade uses a Windows Domain User as the Oracle Home User, then the Oracle Home User on the target version must use the same Windows Domain User.
- When upgrading, if the Oracle home from which you run the database upgrade uses the built-in account (LocalSystem) as the Oracle Home User, then the Oracle Home User on the target version may use the built-in account, a Windows Domain User account, or a Windows Group Managed service account (GMSA).

See Also:

Oracle Database Upgrade Guide for information about how to prepare for upgrading an existing database

Table 1-5 Upgrade Checklist for Oracle RAC

Item	Task
Review Existing Oracle Installations and Upgrade Plans	To install Oracle RAC 12*c* release 2 (12.2), you must have Oracle Grid Infrastructure (Oracle Clusterware and Oracle ASM) 12*c* Release 2 (12.2) installed on your cluster. See Also: *Oracle Database Upgrade Guide* *Oracle Grid Infrastructure Installation Guide*

Table 1-5 (Cont.) Upgrade Checklist for Oracle RAC

Item	Task
Ensure that Oracle RAC Databases You are Installing are Compatible with Existing Databases	• Oracle Clusterware and Oracle ASM are both 12*c* release 2 (12.2) after you perform an Oracle Grid Infrastructure 12*c* release 2 (12.2) installation or upgrade. • If you have an existing Oracle home, then you can create a new Oracle home and install Oracle Database 12*c* release 2 (12.2) into the new Oracle home. Ensure that Oracle Grid Infrastructure is installed in a separate Oracle home. Do not install Oracle Grid Infrastructure in the same Oracle base directory that is used for Oracle Database homes.
Migrate files off of RAW devices	If you have any database data stored on RAW devices, then before you start Oracle Grid Infrastructure and Oracle RAC installation, you must use RMAN to copy that data to Oracle ASM or to another supported file system.
Prepare to upgrade all existing nodes	The Oracle RAC database 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 Oracle RAC database running on a three-node cluster, then you must select all three nodes when upgrading the database using Oracle Universal Installer. You cannot upgrade only two nodes of the cluster, removing the third instance in the upgrade.
Ensure that Oracle RAC database version is equal to or earlier than the version of Oracle Clusterware	You can have multiple Oracle homes for Oracle databases on your cluster. However, the Oracle RAC database software in these homes must be from a version that is equal to or before the version of Oracle Clusterware that is installed; you cannot have a version of Oracle Database running on Oracle Clusterware that was released after the version of Oracle Clusterware that you use. For example: • If your servers use Oracle Grid Infrastructure 12*c* release 2 (12.2), then you can have an Oracle Database 12*c* release 2 (12.2) single-instance database running on one node, and separate Oracle Database 11*g* release 2 (11.2), Oracle RAC 12*c* release 1 (12.1), or Oracle RAC 12*c* release 2 (12.2) databases also running on the cluster. • You cannot have Oracle Grid Infrastructure 11*g* release 2 (11.2) installed on your cluster, and install Oracle RAC 12*c* release 1 (12.1).

2

Installing Oracle RAC and Oracle RAC One Node

After installing Oracle Clusterware, as described in *Oracle Grid Infrastructure Installation Guide* for your platform, you can install Oracle RAC Database software.

Topics:

- Deciding Between Multitenant Container Databases and Non-CDBs in Oracle RAC
  Review the information to decide how to deploy your Oracle Database using the CDB or Non-CDB options.
- Simplified Upgrade of TIMESTAMP WITH TIME ZONE Data
  Time zone files are upgraded when you install Oracle Real Application Clusters.
- Overview of Installation Directories for Oracle RAC
  Both an Oracle Base directory and an Oracle Home directory are used for every installation of Oracle Database software.
- Selecting Installation Options for Oracle RAC
  Review the topics to select options for installing Oracle RAC.
- Specify Oracle Home User Name and Password
  Starting with Oracle Database 12*c* Release 1 (12.1), Oracle Database supports the use of an Oracle Home User.
- Running Oracle Universal Installer to Install Oracle Database
  Review the topics about running Oracle Universal Installer (OUI) to install Oracle Real Application Clusters (Oracle RAC) and Oracle RAC One Node Databases.
- Installing the Oracle Database Vault Option
  Installing and configuring Oracle Database Vault requires actions during and after installation.
- Updating Environment Variables on Remote Nodes
  Changes made to registry and environment variables are not immediately visible to user sessions on remote notes.
- Creating an Oracle RAC Database on Direct NFS
  There are different configuration processes you must perform when installing and create an Oracle RAC database that uses Direct NFS (dNFS) for the database files.

See Also:

Oracle Database Installation Guide for your platform for additional information, such as how to open compressed files, and other information about setting up files for installation

Also, during installation, click **Help** to find explanations of options and prompts for each installation screen, and click **Details** to see the log file

2.1 Deciding Between Multitenant Container Databases and Non-CDBs in Oracle RAC

Review the information to decide how to deploy your Oracle Database using the CDB or Non-CDB options.

Starting in Oracle Database 12*c* Release 1 (12.1), you must create a database as either a multitenant container database (CDB) or as an Oracle database that is non-CDB. This also applies to Oracle RAC databases. The only difference to the installation process is to choose whether to create the Oracle RAC database as a CDB or non-CDB.

A pluggable database (PDB) is a portable collection of schemas, schema objects, and nonschema objects that appears to an Oracle Net client as a non-CDB. PDBs can be plugged into CDBs. A CDB can contain multiple PDBs. Each PDB appears on the network as a separate database.

If you create an Oracle RAC database as a CDB and plug one or more PDBs into the CDB, then, by default, a PDB is not started automatically on any instance of the Oracle RAC CDB. With the first dynamic database service assigned to the PDB (other than the default database service which has the same name as the database name), the PDB is made available on those instances on which the service runs.

Whether or not a PDB is available on more than one instance of an Oracle RAC, CDB is typically managed by the services running on the PDB. You can manually enable PDB access on each instance of an Oracle RAC CDB by starting the PDB manually on that instance.

See Also:

- *Oracle Database Concepts* for more information about PDB concepts
- *Oracle Database Administrator's Guide* for more information about managing PDBs
- *Oracle Real Application Clusters Administration and Deployment Guide* for information specific to the administration of Oracle RAC CDBs

2.2 Simplified Upgrade of TIMESTAMP WITH TIME ZONE Data

Time zone files are upgraded when you install Oracle Real Application Clusters.

As part of an installation of Oracle Database 12*c* release 2 (12.2), time zone files from versions 1 to 12 are installed in the path `Oracle_home\oracore\zoneinfo`. You can continue to use the current time zone file or upgrade to the latest version. Oracle recommends that you upgrade the server to the latest version of the time zone file. Upgrading to a new version of the time zone file may cause existing TIMESTAMP WITH TIME ZONE data to become stale. Using the newly provided DBMS_DST PL/SQL package, you can update the TIMESTAMP WITH TIME ZONE data transparently, with minimal manual procedures and system downtime.

All instances of an Oracle RAC database must use the same time zone. The Oracle RAC database time zone defaults to the time zone setting of the Grid user, unless an instance is started with SQL*Plus. When you use SQL*Plus, you must be sure to use the same time zone setting for the database instance that is used for Oracle Clusterware. You can change the time zone Oracle Clusterware uses for a database by using the following command, where *time_zone* is the time zone to which you want to change:

```
srvctl setenv database -env "TZ=time_zone"
```

Time zone version files are also installed with Oracle Client installations. You do not have to upgrade Oracle Client time zone files immediately. Upgrades can be done at a time when it is most convenient to the system administrator. However, there could be a small performance penalty when client and server use different time zone versions.

See Also:

- *Oracle Database Upgrade Guide* for information about preparing to upgrade TIMESTAMP WITH TIME ZONE data
- *Oracle Database Globalization Support Guide* for information about how to upgrade the time zone file and TIMESTAMP WITH TIME ZONE data
- *Oracle Call Interface Programmer's Guide* for information about performance effects of clients and servers operating with different versions of time zone files

2.3 Overview of Installation Directories for Oracle RAC

Both an Oracle Base directory and an Oracle Home directory are used for every installation of Oracle Database software.

Additionally, on the Windows operating system, Oracle provides a home name for each Oracle Home directory.

- Overview of Oracle Base Directories
  The Oracle base directory is the location where Oracle software and configuration files installed are stored.
- Overview of Oracle Home Directories
  Oracle homes have a name associated with them, which is automatically assigned by the installer.

2.3.1 Overview of Oracle Base Directories

The Oracle base directory is the location where Oracle software and configuration files installed are stored.

By default, Oracle Universal Installer (OUI) installs the Oracle Database software binary files by version and Oracle Home Name in a subdirectory of the Oracle base directory. An Oracle base directory can be used for multiple installations of software by a given installation owner. A separate Oracle base directory is created for each Oracle Home user you specify during installing of the Oracle Database software.

The Oracle Home User has complete control over the Oracle base directory. For reasons of security, different Windows User Accounts used as Oracle Home Users for different Oracle home directories are not allowed to share the same Oracle base directory. However, to support Oracle Database upgrade, Oracle supports the sharing of an Oracle base directory between a Built-in Account and a Windows User Account. If you choose to reuse an Oracle base directory from an earlier release of Oracle Database in Oracle Database 12*c* Release, then the Oracle Home User of the Oracle Database 12*c* Release has complete control over the Oracle base directory of the earlier release.

> **Note:**
>
> The base directory for Oracle Grid Infrastructure 12*c* and the base directory for Oracle RAC 12*c* must be different from the directories used by the Oracle RAC 11*g* release 2 installation.

The default Oracle base path contains the Oracle Home User name if an Oracle Home User is specified during installation of the Oracle Database software. In a default Windows installation, the Oracle base directory appears as follows, where *username* is the Oracle Installation user if you choose Windows Built-in Account as the Oracle Home User, or it is the Oracle Home user if one is specified:

```
DRIVE_LETTER:\app\username
```

If you have separate Oracle Home Users for the Oracle Grid Infrastructure installation and the Oracle RAC installation, then you have two Oracle base paths that are in accordance with Optimal Flexible Architecture (OFA) guidelines. For example, if the user `grid` is the Oracle Home User for the Oracle Grid infrastructure installation and the user `oracle` is the Oracle Home User for the Oracle Database installation, then you have two Oracle base directories. In the following examples, `X:` represents a mounted disk:

- `X:\app\grid`—This is the Oracle base for the Grid user (`grid` in this example), which is the Oracle Home User for the Oracle Grid Infrastructure installation

- `X:\app\oracle`—This is the Oracle base for the Oracle user (`oracle` in this example), which is the Oracle Home User for the Oracle Database installation

> **Caution:**
>
> After installing Oracle Database 12*c* Release 1 (12.1) (or later) release with a Windows User Account as Oracle Home User, do not install older versions of Oracle Databases that share the same Oracle Base Directory. During installation of the software for older releases, the ACLs are reset and Oracle Database 12*c* Release 1 (12.1) (or later) services may not be able to access the Oracle Base directory and files.

2.3.2 Overview of Oracle Home Directories

Oracle homes have a name associated with them, which is automatically assigned by the installer.

The Oracle home directory is located under the Oracle base directory. For example, in a default Windows installation, if you name the Oracle home directory `dbhome_1`, it appears in the Oracle base directory as follows, where *username* is the Oracle Installation user if you do not choose Windows security, or it is the Oracle Home user if one is specified:

```
DRIVE_LETTER:\app\username\product\12.2.0\dbhome_1
```

Ensure that the paths that you select for Oracle software, such as Oracle home paths and the Oracle base path, use only ASCII characters. Because some Oracle software directory paths use installation user names by default, this ASCII character restriction applies to user names, file names, and directory names.

2.4 Selecting Installation Options for Oracle RAC

Review the topics to select options for installing Oracle RAC.

- Selecting a Security Notification Contact
  During installation, you are asked in the Configure Security Updates screen to provide a security contact.
- Selecting an Installation Option
  You must choose one of the installation options for installing the software.
- Selecting the Database Type for Oracle Grid Infrastructure Deployments
  During installation, Oracle Universal Installer (OUI) detects if you have Oracle Grid Infrastructure for a cluster installed. If you do, then you must specify the type of database you plan to create.
- Choosing the Cluster Database Management Type
  When creating an Oracle RAC database, you can choose one of two types of databases to create.
- Selecting an Installation Type
  When you run Oracle Universal Installer (OUI) to install Oracle RAC, you can select the **Typical** or the **Advanced** installation type.

- Selecting an Oracle Database Configuration Type
  The Advanced Installation option enables you to select an Oracle Database configuration type that best matches your expected workload.
- Selecting a Database Name
  The database name is comprised of various strings and must contain only permitted characters. Review the following guidelines when selecting a database name.
- Requirements for Database Passwords
  To secure your database, use passwords that satisfy the Oracle recommended password requirements, even the passwords for predefined user accounts.
- About Automatic Memory Management Installation Options
  Decide if you want to configure Automatic Memory Management during installation.
- About Character Set Selection During Installation
  Review character set options before you start installation.
- Managing Database Services After Installation
  Use the Server Control Utility (SRVCTL), or use Oracle Enterprise Manager Database Express, or Oracle Enterprise Manager Cloud Control for all administration and monitoring of database services for an Oracle RAC database.

2.4.1 Selecting a Security Notification Contact

During installation, you are asked in the Configure Security Updates screen to provide a security contact.

Oracle issues security alerts as needed for vulnerability fixes that are determined to be too critical to wait for distribution in the next Critical Patch Update.

1. Optional: Provide security contact information in one of the following forms:
   - An email address to receive security information for your installation.
   - A My Oracle Support email address or account name to receive security information for your installation, and to enroll your system for Security Updates. You can receive information about alerts through My Oracle Support.

   The information collected by Security Updates is limited to configuration information. The data collected does not include personally identifiable information (except a local contact name in case of transmission problems). You may still use all licensed Oracle functionality if you decline to enable Security Updates

   If you provide your My Oracle Support credentials, then Security Updates automatically gathers configuration information regarding your installed Oracle products and uploads it to Oracle's support systems. You can access the information it collects through your My Oracle Support account, and review health check recommendations, patch recommendations and other recommendations for your system in addition to security alerts.
2. Optional: To choose not to receive security notifications, leave all fields in the Configure Security Updates screen blank.

   You can choose not to provide this information, but Oracle strongly recommends that you configure a security notification contact.
3. Click **Next** to continue.

> **See Also:**
>
> The Oracle Security Policies page, which is available from the following URL: http://www.oracle.com/us/support/assurance/fixing-policies/index.html

2.4.2 Selecting an Installation Option

You must choose one of the installation options for installing the software.

1. On the Select Installation Option page, select one of the following options:
   - **Create and Configure a Database**: Provides you with the option to create a database using a preconfigured database template designed for particular system load demands, such as an online transaction processing (OLTP) database, or a decision support or data warehouse database.
   - **Install database software only**: Installs Oracle Database software; you must complete the database configuration after the installation completes using the installed utilities.
   - **Upgrade an existing database**: Upgrades an existing database.
2. Optional: If you are installing Oracle Database software, then Oracle recommends that you use a preconfigured database option, or select the **Advanced** option on the Select Configuration page, and configure a custom starter database.
3. If you have an existing Oracle installation, then write down the version numbers, patches, and other configuration information, and review upgrade procedures for your existing installation.

> **See Also:**
>
> - *Oracle Database Upgrade Guide* before proceeding with the installation, to decide how you want to proceed.
> - Selecting an Oracle Database Configuration Type for more information about the different preconfigured database options.

2.4.3 Selecting the Database Type for Oracle Grid Infrastructure Deployments

During installation, Oracle Universal Installer (OUI) detects if you have Oracle Grid Infrastructure for a cluster installed. If you do, then you must specify the type of database you plan to create.

- Determine which type of database you plan to create after installing the software:
  - A single-instance database
  - An Oracle RAC database
  - An Oracle RAC One Node database

If you plan to create databases of different types on this cluster, then choose the most advanced option.

For example, if you plan to create only single-instance and Oracle RAC One Node databases, then choose the Oracle RAC One Node database option. If you plan to create single-instance databases and Oracle RAC databases, then choose the Oracle RAC database option.

- If you plan to install Oracle RAC One Node, then you can install the Oracle RAC software on two or more nodes in the cluster.

  An Oracle RAC One Node installation starts an instance on one of the nodes you select as an Oracle RAC One Node pool member. If that instance goes down, then the Oracle RAC One Node instance fails over to another pool member. This feature relocates database instances and connections to other cluster nodes for high availability.

> **See Also:**
>
> - *Oracle Database High Availability Overview* for information about Oracle RAC One Node
> - *Oracle Real Application Clusters Administration and Deployment Guide* for information about how to convert single-instance databases to Oracle RAC

2.4.4 Choosing the Cluster Database Management Type

When creating an Oracle RAC database, you can choose one of two types of databases to create.

- A **policy-managed database**: The database instances are automatically managed based on server pools for effective resource utilization.
- An **administrator-managed database**: The database instances are tied to specific servers in the cluster.

> **See Also:**
>
> *Oracle Real Application Clusters Administration and Deployment Guide* for more information about server pools and the different cluster database management types

2.4.5 Selecting an Installation Type

When you run Oracle Universal Installer (OUI) to install Oracle RAC, you can select the **Typical** or the **Advanced** installation type.

The Typical installation type installs a default configuration of Oracle Database, with basic configuration choices. Oracle recommends that most users select Typical as their installation type.

The Advanced installation type is for customized installations. Use Advanced installation only when you have a specific requirement for it, such as:

- Adding specific components to your installation
- Requiring different passwords for the `SYS`, `SYSTEM` and `DBSNMP` accounts
- Using a different database character set than is in use on your servers
- Changing product languages
- Other nonstandard configurations
- Preconfigured Database Types Supplied with Oracle Database
  The General Purpose and Transaction Processing type and the Data Warehouse type use preconfigured database templates optimized for each type of database.
- Using Advanced Database Configuration
  You use the Advanced Database Configuration option when you have special requirements for your Oracle Database.
- About Installing Oracle Database with Other Languages
  To use languages other than the default (English), either for the database or for applications running on the database, you must use the Advanced Installation method.

2.4.5.1 Preconfigured Database Types Supplied with Oracle Database

The General Purpose and Transaction Processing type and the Data Warehouse type use preconfigured database templates optimized for each type of database.

During installation, Oracle Universal Installer (OUI) starts Oracle Net Configuration Assistant (NETCA) and Oracle Database Configuration Assistant (DBCA), and installs the preconfigured database without further input. During database installation, OUI displays a progress indicator.

DBCA processing for these two configuration types creates a starter database, and configures the Oracle network services.

2.4.5.2 Using Advanced Database Configuration

You use the Advanced Database Configuration option when you have special requirements for your Oracle Database.

Advanced configuration options available using this installation type include Oracle RAC, Automatic Storage Management, backup and recovery configuration, integration with Oracle Enterprise Manager Cloud Control, more fine-grained memory tuning, and other options.

2.4.5.3 About Installing Oracle Database with Other Languages

To use languages other than the default (English), either for the database or for applications running on the database, you must use the Advanced Installation method.

Tip:

By default, Oracle Universal Installer (OUI) configures the character set of a new database based on the language of the operating system.

See Also:

- *Oracle Database Globalization Support Guide* for detailed information about character set and language support
- *Oracle Database Installation Guide for Microsoft Windows* for information about running OUI in different languages

2.4.6 Selecting an Oracle Database Configuration Type

The Advanced Installation option enables you to select an Oracle Database configuration type that best matches your expected workload.

When you choose the Advanced Installation option in Oracle Universal Installer (OUI), then you can select the **General Purpose/Transaction Processing**, **Data Warehouse**, or **Advanced** database configuration type. If you choose the Typical Installation Option, then the database configuration type defaults to General Purpose/ Transaction Processing.

If you select Advanced configuration, then you can use Oracle Database Configuration Assistant (DBCA) to create the database.

Oracle recommends that you use DBCA to create your database.

2.4.7 Selecting a Database Name

The database name is comprised of various strings and must contain only permitted characters. Review the following guidelines when selecting a database name.

The database name input field sets the following Oracle initialization parameter values:

- `DB_NAME`
- `DB_UNIQUE_NAME`
- `DB_DOMAIN`

In Oracle RAC environments, the database name (`DB_UNIQUE_NAME`) portion is a string of no more than 30 characters that can contain alphanumeric, underscore (_), dollar ($), and pound (#) characters, but must begin with an alphabetic character. No other special characters are permitted in a database name. The `DB_NAME` parameter for a database is set to the first 8 characters of the database name.

The domain portion of the global database name (`DB_DOMAIN`) can be no more than 128 characters. Domain names using underscores (_) are not allowed. The values for `DB_UNIQUE_NAME.DB_DOMAIN` in its entirety must be unique within the enterprise.

Database Name and ORACLE_SID

The Oracle Service Identifier (SID) prefix is the first 8 characters of the database name. The SID prefix can contain only the characters a-z, A-Z, and 0-9. The SID prefix cannot contain operating system special characters, so if you use special characters in the first 8 characters of the database name, then these special characters are omitted in the SID prefix. There is a single SID prefix for every database. The SID prefix for a database must be unique within the cluster.

For an Oracle RAC database, each instance has a unique identifier, `ORACLE_SID`, which consists of the SID prefix and an instance number. The `ORACLE_SID` prefix can contain up to 12 characters. The `ORACLE_SID` for Oracle RAC database instances is generated differently, depending on how you choose to manage the database. If you select a policy-managed database, then Oracle generates the SID in the format *name_#*, where name is the first eight alphanumeric characters of `DB_UNIQUE_NAME`, and # is the instance number. If you select an administator-managed database, then Oracle Database Configuration Assistant generates the default SID for the instance names, using the format *name#*, where name is the first eight alphanumeric characters of `DB_UNIQUE_NAME`, and # is the instance number. However, during installation or database creation you can specify a nondefault value for the SID. The instance number is automatically added to the end of this string for each instance.

For an Oracle RAC One Node database, the instance name is *ORACLE_SID_1*, which consists of `_1` appended to the SID prefix. During online relocation, a second instance *ORACLE_SID_2* is started, which becomes the only instance after the relocation completes. The next online relocation uses *ORACLE_SID_1* for the new instance.

Example 2-1 Global Database Name and Related Initialization Parameters

If your database has a global database name of `orl$racprod2551.example.com` which you supplied during installation, then the following values are used for initialization parameters:

Parameter	Value
`DB_UNIQUE_NAME`	`orl$racprod2551`
`DB_DOMAIN`	`example.com`
`DB_NAME`	`orl$racp`

Example 2-2 DB_UNIQUE_NAME and Related ORACLE_SID Values

If the `DB_UNIQUE_NAME` for a database is `orl$racprod2551`, then the following SID values are used:

Database or Instance Type	Value Used for ORACLE_SID
Single-instance Oracle database	`orlracpr`
Policy-managed Oracle RAC instance	`orlracpr_1`
Administrator-managed Oracle RAC instance	`orlracpr1`
Oracle RAC One Node database instance	`orlracpr_1`

2.4.8 Requirements for Database Passwords

To secure your database, use passwords that satisfy the Oracle recommended password requirements, even the passwords for predefined user accounts.

Oracle Database provides a set of predefined user accounts. Create passwords in a secure fashion. If you have default passwords, change these passwords to secure passwords.

You can manage the security for Oracle Database users in various ways:

- Enforce restrictions on the way that passwords are created
- Create user profiles
- Use user resource limits to further secure user accounts

Related Topics:

- *Oracle Database Security Guide*

2.4.9 About Automatic Memory Management Installation Options

Decide if you want to configure Automatic Memory Management during installation.

During a Typical installation, you create your database with Oracle Database Configuration Assistant (DBCA), and automatic memory management is enabled. If you choose advanced installation, then you can either specify memory allocation manually, or enable automatic memory management.

With automatic memory management, the Oracle Database instances automatically manage and tune memory for you. With automatic memory management, you choose a memory target, and the instance automatically distributes memory between the system global area (SGA) and the instance program global area (instance PGA). As memory requirements change, the instance dynamically redistributes memory between the SGA and instance PGA.

You can enable automatic memory management either during, or after the database installation. Enabling automatic memory management after installation involves a shutdown and restart of the database.

> **Note:**
>
> By default, automatic memory management is disabled when you perform typical installation on a node that has more than 4 GB of RAM.

Related Topics:

- *Oracle Database Administrator's Guide*

2.4.10 About Character Set Selection During Installation

Review character set options before you start installation.

After a database is created, changing its character set is usually very expensive in terms of time and resources. Such operations may require converting all character data by exporting the whole database and importing it back. Therefore, it is important that you carefully select the database character set at installation time.

Oracle Database uses character sets for the following:

- Data stored in SQL character data types (`CHAR`, `VARCHAR2`, `CLOB`, and `LONG`).
- Identifiers such as table names, column names, and PL/SQL variables.
- Stored SQL and PL/SQL source code, including text literals embedded in this code.

Starting with Oracle Database 12*c* Release 2 (12.2), the default database character set of a database created from the General Purpose/Transaction Processing or the Data Warehousing template is Unicode AL32UTF8. Oracle recommends that you use Unicode AL32UTF8 as the database character set.

Unicode is the universal character set that supports most of the currently spoken languages of the world. It also supports many historical scripts (alphabets). Unicode is the native encoding of many technologies, including Java, XML, XHTML, ECMAScript, and LDAP. Unicode is ideally suited for databases supporting the Internet and the global economy.

Because AL32UTF8 is a multibyte character set, database operations on character data may be slightly slower when compared to single-byte database character sets, such as WE8ISO8859P1 or WE8MSWIN1252. Storage space requirements for text in most languages that use characters outside of the ASCII repertoire are higher in AL32UTF8 compared to legacy character sets supporting the language. English data may require more space only if stored in `CLOB` (character large object) columns. Storage for non-character data types, such as `NUMBER` or `DATE`, does not depend on a character set. The universality and flexibility of Unicode usually outweighs these additional costs.

Consider legacy character sets only when the database need to support a single group of languages and the use of a legacy character set is critical for fulfilling compatibility, storage, or performance requirements. The database character set to be selected in this case is the character set of most clients connecting to this database.

The database character set of a multitenant container database (CDB) determines which databases can be plugged in later. Ensure that the character set you choose for the CDB is compatible with the database character sets of the databases to be plugged into this CDB.

> **See Also:**
>
> *Oracle Database Globalization Support Guide* for more information about choosing a database character set for a multitenant container database (CDB)

2.4.11 Managing Database Services After Installation

Use the Server Control Utility (SRVCTL), or use Oracle Enterprise Manager Database Express, or Oracle Enterprise Manager Cloud Control for all administration and monitoring of database services for an Oracle RAC database.

You cannot use Oracle Database Configuration Assistant (DBCA) to manage database services for Oracle RAC databases.

Oracle Enterprise Manager Cloud Control is available separately on the Oracle Enterprise Manager Cloud Control installation media, and on the Oracle Technology Network website at the following website:

http://www.oracle.com/technetwork/indexes/documentation/index.html

> **See Also:**
>
> *Oracle Enterprise Manager Online Help* for service management using Oracle Enterprise Manager

2.5 Specify Oracle Home User Name and Password

Starting with Oracle Database 12*c* Release 1 (12.1), Oracle Database supports the use of an Oracle Home User.

An Oracle Home User is a standard Windows User Account (not an Administrator account), specified during installation, that runs the Windows services required by Oracle Database for the Oracle home.

The Oracle Home User is associated with an Oracle Home and it cannot be changed post installation. Different Oracle Homes on a system can share the same Oracle Home User or use different Oracle Home User names. For Oracle RAC databases, the Windows user account for the Oracle Home must be a domain account and it has to be an existing account.

For Administrator-managed databases, you can store the password for the Oracle Home User in a secure wallet in the Oracle Cluster Registry (OCR). If such a wallet exists in the OCR, then Oracle Database administration tools automatically use the password from the wallet and do not require you to enter the password for the Oracle Home User during administrative operations.

For Policy-managed databases, you must store the password for the Oracle Home User in a secure wallet in the OCR. Oracle Database Configuration Assistant automatically creates the wallet (if one does not exist) when a Policy-managed database is created.

> **See Also:**
>
> - *Oracle Grid Infrastructure Installation Guide for Microsoft Windows x64 (64-Bit)* for information on configuring the Oracle Installation User and Oracle Home User.
> - *Oracle Database Platform Guide for Microsoft Windows* for information about creating Oracle wallets for Oracle Database services

2.6 Running Oracle Universal Installer to Install Oracle Database

Review the topics about running Oracle Universal Installer (OUI) to install Oracle Real Application Clusters (Oracle RAC) and Oracle RAC One Node Databases.

- Using Oracle Universal Installer to Install Oracle RAC
  Oracle Universal Installer is the primary tool you use to install the Oracle RAC software.
- Using Oracle Universal Installer to Install Oracle RAC One Node
  Oracle Universal Installer (OUI) is the primary tool you use to install Oracle RAC One Node software.

2.6.1 Using Oracle Universal Installer to Install Oracle RAC

Oracle Universal Installer is the primary tool you use to install the Oracle RAC software.

1. Log in to Windows as a user with Administrator privileges.

   If using a different user than the one that installed Oracle Grid Infrastructure, then the installation user must be a member of the ASMDBA and ASMADMIN operating system groups.

2. Verify that you have Administrator privileges on the other nodes.

   To verify privileges, enter the following command for each node that is a part of the cluster where *node_name* is the name of the remote node:

   ```
   net use \\node_name\C$
   ```

3. Run the `setup.exe` command from the base directory of the Oracle Database 12*c* release 2 (12.2) installation media, and select the database type to install.

4. Provide information when prompted by Oracle Universal Installer (OUI).

   If you need assistance during installation, then click **Help**.

   If you encounter problems during installation, then click **Details** to examine the log file. The log file is located in the Oracle Inventory directory with a name that includes the time stamp (*date_time*) of the installation process, as shown in this example:

   ```
   C:\Program Files\Oracle\Inventory\logs\installActionsdate_time.log
   ```

   In the preceding example, the variables *date* and *time* represent the date and the time of the log file.

> **Caution:**
>
> The Oracle home name and path that you provide during database installation *must be different* from the home that you used for the Oracle Grid Infrastructure software. DO NOT install the Oracle Database with Oracle RAC software into the same home in which you installed Oracle Grid Infrastructure software.

> **Note:**
>
> Because Oracle ASM is part of the Oracle Grid Infrastructure home, it cannot be upgraded during an Oracle RAC installation.

5. When you have completed this second and final phase of the installation, proceed to "Oracle Real Application Clusters Postinstallation Procedures" to perform the postinstallation tasks.

> **Caution:**
>
> After you complete the installation create the database, to install additional Oracle Database 12*c* release 2 (12.2) products in the database, you must stop *all* processes running in the Oracle home before you attempt to install the additional products. See "Preparing to Upgrade an Existing Oracle RAC Database" for additional information

2.6.2 Using Oracle Universal Installer to Install Oracle RAC One Node

Oracle Universal Installer (OUI) is the primary tool you use to install Oracle RAC One Node software.

1. Login to Windows as a user with Administrator user privileges.
2. Run the `setup.exe` command from the base directory of the Oracle Database 12*c* release 2 (12.2) installation media.
3. Select to configure an Oracle RAC One Node database, and select to install Oracle RAC on all nodes where you want to run Oracle RAC One Node.
4. Provide information when prompted by OUI. If you need assistance during installation, then click **Help**. Click **Details** to see the log file.

If you encounter problems during installation, then examine the OUI actions recorded in the installation log file. The log file is located in the Oracle Inventory directory with a name that includes the timestamp (*date_time*) of the install process, as shown in this example:

```
C:\Program Files\Oracle\Inventory\logs\installActionsdate_time.log
```

> **Caution:**
>
> The Oracle home name and path that you provide during database installation *must be different* from the home that you used for the Oracle Grid Infrastructure software. DO NOT install the Oracle Database with Oracle RAC software into the same home in which you installed Oracle Grid Infrastructure software.

> **Note:**
>
> Because Oracle ASM is part of the Oracle Grid Infrastructure home, it cannot be upgraded during an Oracle RAC installation.

2.7 Installing the Oracle Database Vault Option

Installing and configuring Oracle Database Vault requires actions during and after installation.

- Starting the Listener with Oracle Database Vault Installations
  You must start the listener and database instance on all Oracle RAC nodes other than the one on which the installation is performed.
- Configuring Oracle Database Vault Using DBCA
  You can configure Oracle Database Vault after installation using Oracle Database Configuration Assistant (DBCA), or choose not to configure Oracle Database Vault.
- Perform Postinstallation Configuration for Oracle Database Vault
  After you install the Oracle Database Vault option, you may be required to make additional changes to your database.

2.7.1 Starting the Listener with Oracle Database Vault Installations

You must start the listener and database instance on all Oracle RAC nodes other than the one on which the installation is performed.

- Use Server Control (SRVCTL) to start and stop the Oracle RAC instances being configured for Oracle Database Vault.

  Do not use SQL*Plus to start and stop Oracle RAC instances.

2.7.2 Configuring Oracle Database Vault Using DBCA

You can configure Oracle Database Vault after installation using Oracle Database Configuration Assistant (DBCA), or choose not to configure Oracle Database Vault.

To install Oracle Database Vault using DBCA:

1. After installing Oracle RAC, create the database.
2. Start DBCA and select the option **Configure Database**.
3. In the component list, select **Oracle Label Security** and **Oracle Database Vault**.

4. Provide the required Oracle Database Vault user IDs and passwords to proceed with configuration. To enable a separate Oracle Database Vault administrator, choose to configure the `DV_ACCTMGR` user.
5. After you have finished, restart each database instance to finish the software configuration.

> **See Also:**
>
> *Oracle Database Vault Administrator's Guide* for information about using Oracle Data Guard with Oracle Database Vault

2.7.3 Perform Postinstallation Configuration for Oracle Database Vault

After you install the Oracle Database Vault option, you may be required to make additional changes to your database.

1. Refer to *Oracle Database Vault Administrator's Guide* for required postinstallation steps.
2. If you use other Oracle Database products, then refer to *Oracle Database Vault Administrator's Guide* for information about integrating Database Vault with other Oracle products, such as Transparent Data Encryption or Oracle Data Guard.

Related Topics:

- Configuring Oracle Database Vault
  Oracle Universal Installer (OUI) installs Oracle Database Vault by default when you install the Oracle RAC software, but requires additional configuration steps.

2.8 Updating Environment Variables on Remote Nodes

Changes made to registry and environment variables are not immediately visible to user sessions on remote notes.

During the installation, the Windows registry and environment variables are modified on each node on which you installed Oracle RAC. The new registry entries and environment variable settings are visible on the node where the installation was performed (the local node). However, the new settings on the remote nodes are not immediately available to your user session. Attempting to run scripts or applications from the newly installed Oracle home can produce errors similar to the following:

- `ORACONFIG.exe - Unable to Locate Component`
- `OCI.dll not found`

To make the modified environment variables available on the remote nodes, you can do either of the following actions:

1. Close your current session on the remote nodes (log off) and then log on to the remote nodes to create a new session.
2. Make the environment variables available to the remote nodes:
   a. From the Start menu, right-click **My Computer**, then select **Properties**

      Alternatively, you can enter `SYSDM.CPL` in the Run window.

b. Select the **Advanced** tab.

c. Click **Environment Variables**.

   The modified environment variables are now visible.

d. Click **OK** to close the System Properties window.

2.9 Creating an Oracle RAC Database on Direct NFS

There are different configuration processes you must perform when installing and create an Oracle RAC database that uses Direct NFS (dNFS) for the database files.

- Perform a Software-Only Installation of Oracle Database
  In a software-only installation you install the Oracle Database software but do not create a database as part of the installation process.
- Use ASMCA to Configure an ACFS Mount Point
  Oracle Automatic Storage Management Configuration Assistant (ASMCA) enables you to create an Oracle Automatic Storage Management Cluster File System (Oracle ACFS) mount point which is used in the "common file location" step of Oracle Database Configuration Assistant (DBCA).
- Use DBCA to Create and Configure the Oracle RAC Database
  Use Oracle Database Configuration Assistant (DBCA) to create an Oracle Real Application Clusters (Oracle RAC) database that uses Direct NFS for datafile storage.
- Enable and Configure Direct NFS
  You must manually enable the Direct NFS option after installing the Oracle Database software.
- Use ASMCA to Remove the ACFS Mount Point
  Now you have configured the mount point using Direct NFS, you can remove the Oracle Automatic Storage Management Cluster File System (Oracle ACFS) mount point using Oracle Automatic Storage Management Configuration Assistant (ASMCA).

2.9.1 Perform a Software-Only Installation of Oracle Database

In a software-only installation you install the Oracle Database software but do not create a database as part of the installation process.

1. Start Oracle Universal Installer (OUI) by running `setup.exe` from the software staging location.
2. On the Select Installation Option screen select **Install database software only**.
3. On the Grid Installation Options screen, select the product to install, for example "Oracle Real Application Clusters database software installation."
4. Select the nodes on which you want to install the database software.
5. Choose the languages to use with the installed software.
6. Select the database edition to install. For Oracle RAC databases, you must choose Enterprise Edition.
7. Specify an Oracle Home user, or choose to use a Windows-built in user for the software installation owner.

8. On the Specify Installation Location screen, enter a path to the Oracle base directory and the software location (Oracle home directory).
9. On the Summary screen, verify your selections, then click **Install**.

2.9.2 Use ASMCA to Configure an ACFS Mount Point

Oracle Automatic Storage Management Configuration Assistant (ASMCA) enables you to create an Oracle Automatic Storage Management Cluster File System (Oracle ACFS) mount point which is used in the "common file location" step of Oracle Database Configuration Assistant (DBCA).

When creating Oracle ACFS file systems on Windows, run ASMCA as a Windows domain user who is also an administrator of the computer.

1. From the *`Grid_home`*`/bin` directory, run `asmca.exe` to start the ASMCA.
2. Select the **Disk Groups** tab.
3. Right-click **Disk Group Name** and select **Create ACFS for Database use**.
4. In the Create ACFS for Database window specify the mount point location, volume name, and size, then click **OK**.

   For example, you can specify the following:

   - Mount Point: `C:\oradatamnt`
   - Volume Name: `dbnfs`
   - Size (GB): `70`
5. Click **OK** in the informational pop-up window that appears.

   This window summarizes the actions performed by ASMCA.
6. Select the **ASM Cluster File Systems** tab.

   The mount point you just created is displayed on this page.

> **See Also:**
>
> *Oracle Automatic Storage Management Administrator's Guide* for more information about creating an Oracle ACFS file system for database use

Related Topics:

- Use DBCA to Create and Configure the Oracle RAC Database
  Use Oracle Database Configuration Assistant (DBCA) to create an Oracle Real Application Clusters (Oracle RAC) database that uses Direct NFS for datafile storage.

2.9.3 Use DBCA to Create and Configure the Oracle RAC Database

Use Oracle Database Configuration Assistant (DBCA) to create an Oracle Real Application Clusters (Oracle RAC) database that uses Direct NFS for datafile storage.

1. From the *`Oracle_home`*`\bin` directory, run `dbca.exe` to start the Database Configuration Assistant.

2. On the Database Operation screen, select **Create Database**.
3. On the Creation Mode screen, select **Advanced Mode**.
4. On the Database Template screen, select **Oracle Real Application Clusters (RAC) database** for the Database Type.

   For the Configuration Type, you can choose either Policy-Managed or Administrator-Managed. Select the template most appropriate for the type of database you want to create.
5. For the next four screen, make selections and provide information that best meet your business requirements.
6. Perform the following steps on the Storage Locations screen:
   a. Select **File System** for the Database files Storage type.
   b. Select the option **Use Common Location for All Database Files**.
   c. In the File location field, specify the location of the ACFS mount point, for example, `C:\oradatamnt`.
7. On the Specify Database Options screen, choose any additional configuration you want for your database.
8. On the Initialization Parameters screen, use the default settings, or provide customized values for the initialization parameters.
9. On the Creation Options screen, select the option **Generate Database Creation Scripts**. Specify a destination directory for the script file, or use the default value.
10. After the Prerequisite checks complete, on the Summary screen, minimize the installation window. **DO NOT** click Finish at this point.
11. Enable the Direct NFS option.
12. Remove the virtual mount point you created with Oracle Automatic Storage Management Configuration Assistant (ASMCA).
13. Create all the directories needed locally on each node as well as on the NFS server.

    For this example, you can create the following, where `orcl` represents the database SID and `pdb1` represents the Pluggable Database (PDB) name:
    - On each node, create the directory `c:\oracle\oradatamnt\orcl\pdb1`
    - On the NFS server, create the directory `/export/abcd/orcl/pdb1`
14. Return to the DBCA window and click **Finish**.
15. Run the generated scripts on the cluster node to create the database.
16. Map a drive letter to a CIFS share on the NFS server that represents the location of the database files.

    Use a command similar to the following:

    ```
    NET USE * \\filer\vol0\orcl
    ```

    After you complete this step, both Oracle and the Windows OS can access the location where the database files reside. Oracle is using DNFS, but the Windows OS uses CIFS to access the same location on the NFS server.
17. Verify Direct NFS is configured for the database.

a. Start SQL*Plus.

b. Connect to the newly created database as a DBA user.

c. Run the following SQL command:

```
SELECT * FROM v$dnfs_servers;
```

Related Topics:

- Use ASMCA to Configure an ACFS Mount Point
  Oracle Automatic Storage Management Configuration Assistant (ASMCA) enables you to create an Oracle Automatic Storage Management Cluster File System (Oracle ACFS) mount point which is used in the "common file location" step of Oracle Database Configuration Assistant (DBCA).
- Enable and Configure Direct NFS
  You must manually enable the Direct NFS option after installing the Oracle Database software.
- Use ASMCA to Remove the ACFS Mount Point
  Now you have configured the mount point using Direct NFS, you can remove the Oracle Automatic Storage Management Cluster File System (Oracle ACFS) mount point using Oracle Automatic Storage Management Configuration Assistant (ASMCA).

2.9.4 Enable and Configure Direct NFS

You must manually enable the Direct NFS option after installing the Oracle Database software.

1. Run the program *`Oracle_home`*`\bin\enable_dnfs.bat`.
2. Create an `oranfstab` file.

   See for details.

Related Topics:

- Use DBCA to Create and Configure the Oracle RAC Database
  Use Oracle Database Configuration Assistant (DBCA) to create an Oracle Real Application Clusters (Oracle RAC) database that uses Direct NFS for datafile storage.
- *Oracle Grid Infrastructure Installation Guide for Microsoft Windows x64 (64-Bit)*

2.9.5 Use ASMCA to Remove the ACFS Mount Point

Now you have configured the mount point using Direct NFS, you can remove the Oracle Automatic Storage Management Cluster File System (Oracle ACFS) mount point using Oracle Automatic Storage Management Configuration Assistant (ASMCA).

When creating Oracle ACFS file systems on Windows, run ASMCA as a Windows domain user who is also an administrator of the computer.

1. From the *`Grid_home`*`/bin` directory, run `asmca.exe` to start the ASMCA.
2. Select the **ASM Cluster File System** tab.
3. Select the mount point created earlier (`C:\oradatamnt`), then click **Dismount All**.

4. Select the **Volumes** tab.
5. Right-click the mount point and then select **Delete**.

Related Topics:

- Use DBCA to Create and Configure the Oracle RAC Database
  Use Oracle Database Configuration Assistant (DBCA) to create an Oracle Real Application Clusters (Oracle RAC) database that uses Direct NFS for datafile storage.

3

Creating Oracle RAC or Oracle RAC One Node Databases with DBCA

Use Oracle Database Configuration Assistant (DBCA) in standalone mode to create and delete Oracle Real Application Clusters (Oracle RAC) databases.

Topics:

- Using DBCA with Oracle RAC or Oracle RAC One Node
  Oracle Database Configuration Assistant (DBCA) is a tool for creating and configuring an Oracle database.
- Benefits of Using Database Configuration Assistant
  Understand the benefits of using Oracle Database Configuration Assistant (DBCA) to create your Oracle RAC databases.
- Automatic Listener Migration from Earlier Releases
  Review this information for listener migration from earlier database releases.
- Verifying Requirements for DBCA
  Use Cluster Verification Utility (CVU) to verify that your system is prepared for configuration changes.
- Tasks to Complete Before Using DBCA to Create an Oracle RAC Database
  Before you can create an Oracle RAC database using Oracle Database Configuration Assistant, you must configure your system to meet the software requirements, if this was not done as part of the Oracle Grid Infrastructure installation.
- Selecting DBCA Options to Create an Oracle RAC or Oracle RAC One Node Database
  Review this information to use Database Configuration Assistant (DBCA) to create Oracle RAC or Oracle RAC One Node Database.
- Using DBCA to Create an Oracle RAC One Node Database
  If you have selected to install only the Oracle RAC software on cluster nodes, then you can use Oracle Database Configuration Assistant (DBCA) to configure Oracle RAC One Node.
- Deleting an Oracle RAC Database Using DBCA
  Deleting an Oracle RAC database using Oracle Database Configuration Assistant (DBCA) involves deleting the database and database objects.

3.1 Using DBCA with Oracle RAC or Oracle RAC One Node

Oracle Database Configuration Assistant (DBCA) is a tool for creating and configuring an Oracle database.

DBCA has the following primary database functions:

- Create and delete databases
- Create database templates

- Create, plug, unplug, and delete pluggable databases (PDBs)
- Add and delete database instances
- Register databases in Oracle Enterprise Manager Cloud Control
- Configure and register database options (such as Oracle Database Vault) with the Directory Server

> **Note:**
>
> Cluster Managed Services are no longer managed through DBCA. Instead, use the Cluster Managed Services page in Oracle Enterprise Manager Cloud Control, if available, or SRVCTL. For more information, see *Oracle Real Application Clusters Administration and Deployment Guide*.

Related Topics:

- Selecting DBCA Options to Create an Oracle RAC or Oracle RAC One Node Database
  Review this information to use Database Configuration Assistant (DBCA) to create Oracle RAC or Oracle RAC One Node Database.

> **See Also:**
>
> - *Oracle Database Net Services Administrator's Guide* to resolve problems —for example, with the listener configuration—and for more information about Lightweight Directory Access Protocol (LDAP)-compliant directory support
> - *Oracle Database 2 Day + Real Application Clusters Guide* for more information about using Oracle Enterprise Manager

3.2 Benefits of Using Database Configuration Assistant

Understand the benefits of using Oracle Database Configuration Assistant (DBCA) to create your Oracle RAC databases.

Oracle recommends that you use DBCA to create your Oracle RAC database, because preconfigured databases optimize your environment for features such as the server parameter file (SPFILE), and automatic undo management.

DBCA enables you to create both policy-managed and administrator-managed databases. With DBCA, you can also 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 Oracle RAC environment for various Oracle high availability features, such as cluster administration tools. DBCA also starts any database instances required to support your defined configuration.

You can use DBCA to create a database from templates supplied by Oracle, or from templates that you create. The templates contain settings optimized for a particular type of workload.

Oracle ships templates for the following two workload types:

- General purpose or transaction processing
- Data warehouse

For more complex environments, you can select the **Custom Database** option. This option does not use templates and results in a more extensive interview, which means that it takes longer to create your database.

Click **Show Details** to see the configuration for each type of database. Select the template suited to the type of workload your database supports. If you are not sure which to choose, then select the default **General Purpose or Transaction Processing** template.

3.3 Automatic Listener Migration from Earlier Releases

Review this information for listener migration from earlier database releases.

If your system has an Oracle Database 11*g* or 12*c* installation, and you install Oracle Database 12*c* Release 2 (12.2) either to coexist with or to upgrade the Oracle Database 11.2 or 12.1 installation, then most installation types automatically migrate the existing Oracle Database listener to the 12*c* Release 2 (12.2) Oracle home. During migration, the upgrade process configures and starts a default Oracle Net Listener using the same TCP/IP port as the existing listener, with the IPC key value.

During the Oracle Clusterware upgrade, the default listener (`LISTENER_`*`nodename`*) is migrated to the Oracle Grid Infrastructure home (Grid home). Oracle Database Configuration Assistant always uses the default listener.

The listener migration process stops the listener in the existing Oracle home, and restarts it in the new Oracle home. If the database was using the default listener (`LISTENER_`*`nodename`*), then Oracle Net Configuration Assistant (NETCA) migrates the listener automatically to the Grid home as part of the Oracle Grid Infrastructure upgrade. If the database was using a nondefault listener, then Database Upgrade Assistant (DBUA) migrates the nondefault listener to the Oracle Database home.

> **Note:**
>
> During migration, client applications may not be able to connect to any databases that are registered to the listener that is being migrated.

3.4 Verifying Requirements for DBCA

Use Cluster Verification Utility (CVU) to verify that your system is prepared for configuration changes.

- Prior to using Database Configuration Assistant (DBCA) to change the database configuration, run Cluster Verification Utility (CVU) to verify that your system is prepared for configuration changes using the following command syntax:

```
Grid_home\bin\cluvfy stage -pre dbcfg -n node_list -d Oracle_home [-verbose]
```

In the preceding syntax example, the variable *Grid_home* is the Oracle Grid Infrastructure home, 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 Oracle Universal Installer (OUI) creates or modifies the database.
You can select the option `-verbose` to receive progress updates as CVU performs its system checks, and detailed reporting of the test results.

If the CVU summary indicates that the cluster verification check fails, then review and correct the relevant system configuration steps, and run the test again.

Example 3-1 Using CVU Prior to Verify Your System is Prepared for an Oracle RAC Installation

To verify that your system is prepared for an Oracle Database with Oracle RAC installation on a two-node cluster with nodes `node1` and `node2`, with the Grid home `C:\app\12.2.0\grid`, and with the Oracle home path `C:\app\oracle\product\12.2.0\dbhome_1`, enter the following command:

```
C:\app\12.2.0\grid\bin> cluvfy stage -pre dbcfg -n node1,node2 \
-d C:\app\oracle\product\12.2.0\dbhome_1
```

3.5 Tasks to Complete Before Using DBCA to Create an Oracle RAC Database

Before you can create an Oracle RAC database using Oracle Database Configuration Assistant, you must configure your system to meet the software requirements, if this was not done as part of the Oracle Grid Infrastructure installation.

- Decide on a Naming Convention to Use for Your Oracle RAC Database
  The global database name for an Oracle RAC database must meet the naming requirements. The global database name consists of the database name and the domain name.
- Configure Shared Storage for the Oracle RAC Database
  Before starting Oracle Database Configuration Assistant (DBCA) to configure an Oracle RAC database, you must install Oracle Grid Infrastructure for a cluster, and configure shared storage areas for Oracle RAC files.
- Obtain the Password for the Oracle Home User
  When you use Oracle Database Configuration Assistant to create an Oracle database or an Oracle RAC database, and specify an Oracle Home User during installation, then you need to enter the password for this user.

3.5.1 Decide on a Naming Convention to Use for Your Oracle RAC Database

The global database name for an Oracle RAC database must meet the naming requirements. The global database name consists of the database name and the domain name.

1. Choose a name for your database that has the following characteristics:
   a. Up to 30 characters in length

   b. Begins with an alphabetic character

2. Determine the domain name portion of the global database name, that satisfies these requirements:

   a. Is up to 128 characters in length

   b. Contains only alphabetic and numeric characters, and the period (.) character

3. Determine the `ORACLE_SID` values for each instance.

   The maximum number of characters you can use for the SID prefix is 8 characters. Oracle Database Configuration Assistant uses the SID prefix to generate a unique value for the variable `ORACLE_SID` for each instance. The SID prefix must begin with an alphabetic character.

3.5.2 Configure Shared Storage for the Oracle RAC Database

Before starting Oracle Database Configuration Assistant (DBCA) to configure an Oracle RAC database, you must install Oracle Grid Infrastructure for a cluster, and configure shared storage areas for Oracle RAC files.

1. Login as a user with SYSASM system privileges.

   Storage administration tasks require the SYSASM system privileges, which are granted to members of the OSASM operating system group. This group may not be the same as the OSDBA group, whose members are granted the SYSDBA system privileges.

2. On Windows-based systems, if you plan to use Oracle ASM storage, then before you use DBCA to create a database, you must perform the following steps:

   a. Create logical partitions without primary partitions on the disk drives.

   b. Delete the drive letters for these partitions on all nodes.

      Alternatively, you can stamp these partitions with `asmtool`.

   c. After you have configured the disks to be used by Oracle ASM, you must create the disk groups that is used by the database.

      You can create disk groups by using SQL*Plus, Oracle Enterprise Manager, or Oracle Automatic Storage Management Configuration Assistant (ASMCA).

> **See Also:**
>
> - *Oracle Database Installation Guide for Microsoft Windows* for more information about `asmtoolg`
> - *Oracle Grid Infrastructure Installation Guide* for your platform more information on shared storage configuration requirements
> - *Oracle Automatic Storage Management Administrator's Guide* for more information about creating disk groups

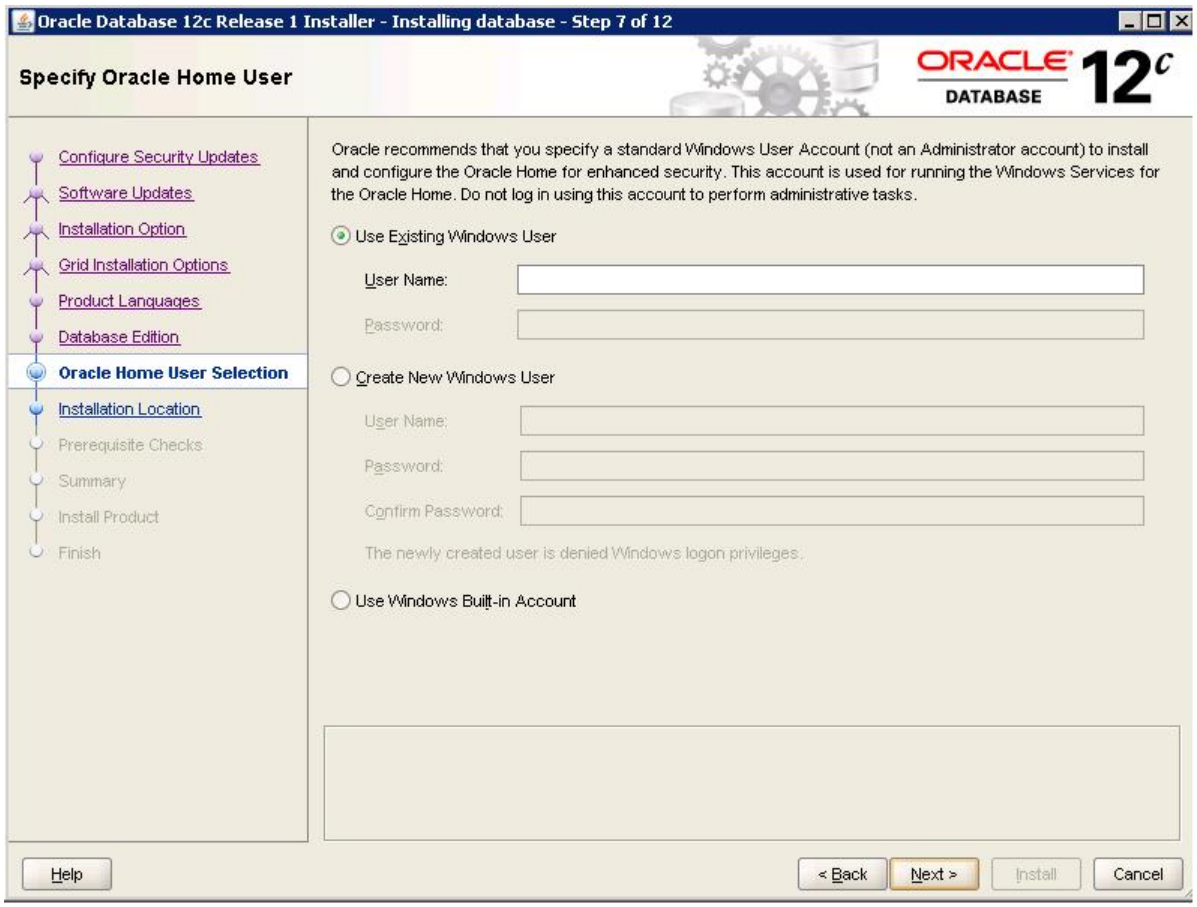
3.5.3 Obtain the Password for the Oracle Home User

When you use Oracle Database Configuration Assistant to create an Oracle database or an Oracle RAC database, and specify an Oracle Home User during installation, then you need to enter the password for this user.

Contact the user that installed the Oracle Database software and perform the following task:

- Determine if an Oracle Home user was specified and if an Oracle Home user was specified, get the password specified for that user account.

Figure 3-1 Specifying the Password for the Oracle Home User



> **See Also:**
>
> *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)* for more information about the Oracle Home user.

3.6 Selecting DBCA Options to Create an Oracle RAC or Oracle RAC One Node Database

Review this information to use Database Configuration Assistant (DBCA) to create Oracle RAC or Oracle RAC One Node Database.

You no longer need to set the operating system environment variables `ORACLE_HOME` to the Oracle RAC database home, or `ORACLE_UNQNAME` to the database unique name.

> **Note:**
>
> You can no longer set up email notification for Oracle RAC databases either from DBCA or Oracle Universal Installer (OUI).

- Starting DBCA
  You can either start the Oracle Database Configuration Assistant (DBCA) utility from the command line or from the Windows Start menu.
- Cluster Detection and Node Selection when Using DBCA
  When you start Oracle Database Configuration Assistant (DBCA), it automatically shows options for Oracle RAC if it detects from the central Oracle Inventory that the Oracle Home is enabled for Oracle RAC.
- Using DBCA to Select Storage to Use With Oracle RAC Database
  You can choose to use either Oracle ASM Disk groups or a shared file system as storage for Oracle RAC database files.
- Using DBCA to Specify Database Initialization Parameters for Oracle RAC
  Set the `CLUSTER_DATABASE_INSTANCES` parameter to the expected number of instances.
- Actions Performed By DBCA for Oracle RAC Databases
  Review this information to understand about Oracle Database Configuration Assistant (DBCA) actions during Oracle RAC database creation.

Related Topics:

- Using DBCA with Oracle RAC or Oracle RAC One Node
  Oracle Database Configuration Assistant (DBCA) is a tool for creating and configuring an Oracle database.

3.6.1 Starting DBCA

You can either start the Oracle Database Configuration Assistant (DBCA) utility from the command line or from the Windows Start menu.

> **Note:**
>
> To run DBCA, you *do not* have to set operating system environment variables `ORACLE_HOME` to the Oracle RAC database home, or `ORACLE_UNQNAME` to the database unique name.

1. Log in as an Administrator user.

   The user must also be a member of `ORA_DBA` or `ORA_Homename_DBA` group and must also be a member of `ORA_ASMDBA` if Oracle ASM is used as storage for the Oracle RAC database.

   You are prompted to enter the password for the Oracle Home User if you are administering an Administrator-managed Oracle RAC database and chose not to store the password in an Oracle Wallet.

2. To start DBCA from the command line:
   a. Open a command prompt window.
   b. Navigate to the `Oracle_home\bin` directory.
   c. Enter the command `dbca`.
3. To start DBCA from the Start menu:
   a. Click **Start**.
   b. Select Programs.
   c. Under Programs, select **Oracle - *Oracle_home name***.
   d. Select **Configuration and Migration Tools**.
   e. Select **Database Configuration Assistant**.
4. After you have started DBCA, to create an Oracle RAC database, you select the following:
   - **Create Database** on the Database Operation/Welcome page
   - **Advanced Mode** on the Creation Mode page
   - **Oracle RAC database** on the Database Template page

Related Topics:

- Deleting an Oracle RAC Database Using DBCA
  Deleting an Oracle RAC database using Oracle Database Configuration Assistant (DBCA) involves deleting the database and database objects.

3.6.2 Cluster Detection and Node Selection when Using DBCA

When you start Oracle Database Configuration Assistant (DBCA), it automatically shows options for Oracle RAC if it detects from the central Oracle Inventory that the Oracle Home is enabled for Oracle RAC.

If DBCA does not detect the Oracle home as an Oracle RAC home, check that the Oracle Universal Installer (OUI) inventory is correctly located in the directory `C:\Program Files\Oracle\Inventory`, and that the `oraInventory` file is not corrupted. Also, perform clusterware diagnostics by using the following CVU command syntax:

```
Grid_home\bin\cluvfy\cluvfy.bat stage -post crsinst -n nodelist
```

When using DBCA, if nodes that are part of your cluster installation do not appear on the Node Selection page, then run the Opatch `lsinventory` command to perform inventory diagnostics. Also use CVU to perform clusterware diagnostics.

Example 3-2 Performing Clusterware Diagnostics If DCBA Fails To Detect A Two-Node Cluster

If the Grid Home is `D:\app\12.2.0\grid`, and the nodes are named `node1` and `node2`, then run the following command to perform clusterware diagnostics:

```
D:\app\12.2.0\grid\bin> cluvfy stage -post crsinst -n node1,node2
```

3.6.3 Using DBCA to Select Storage to Use With Oracle RAC Database

You can choose to use either Oracle ASM Disk groups or a shared file system as storage for Oracle RAC database files.

- On the Specify Database Storage Options page, if you do not see the diskgroups in Oracle Database Configuration Assistant (DBCA), then either Oracle ASM is not configured, or the diskgroups are not mounted.

  You can create diskgroups using Oracle Automatic Storage Management Configuration Assistant (ASMCA) in the Grid Infrastructure home before starting DBCA.
- If you are using Oracle ASM, then you can select the Fast Recovery Area and size on the Specify Database Storage Options page.

  When using Oracle ASM, the Fast Recovery Area defaults to the Oracle ASM Disk Group.

3.6.4 Using DBCA to Specify Database Initialization Parameters for Oracle RAC

Set the `CLUSTER_DATABASE_INSTANCES` parameter to the expected number of instances.

1. On the Initialization Parameters page, if you intend to add more nodes in your cluster than you have during the current Oracle Database Configuration Assistant session, then click **All Initialization Parameters**, and change the parameter `CLUSTER_DATABASE_INSTANCES` to the total number of nodes that you plan to add to the cluster.
2. In addition, if you click **All Initialization Parameters**, note that if your global database name is longer than 8 characters, then the database name value (in the `DB_NAME` parameter) is truncated to the first 8 characters, and the `DB_UNIQUE_NAME` parameter value is set to the global name.

> **See Also:**
>
> *Oracle Database Administrator's Guide* for information about initialization parameters

3.6.5 Actions Performed By DBCA for Oracle RAC Databases

Review this information to understand about Oracle Database Configuration Assistant (DBCA) actions during Oracle RAC database creation.

After you respond to DBCA prompts, review the Summary dialog information and click **OK**, DBCA performs several actions.

- Creates an Oracle RAC database, and its instances
- Creates the Oracle RAC data dictionary views
- Starts the Oracle services if you are on a Windows-based platform
- Starts the Oracle Clusterware high availability services
- Starts the database instances across cluster nodes

> **Caution:**
>
> After you have created the Oracle RAC database, if you decide to install additional Oracle Database products in the Oracle RAC database you have created, then before you attempt to install the products, you must stop all processes running in the Oracle RAC database homes.
>
> You must stop all processes running in the Oracle RAC homes so that Oracle Universal Installer can relink certain executables and libraries. Refer to "Preparing to Upgrade an Existing Oracle RAC Database" for additional information.

3.7 Using DBCA to Create an Oracle RAC One Node Database

If you have selected to install only the Oracle RAC software on cluster nodes, then you can use Oracle Database Configuration Assistant (DBCA) to configure Oracle RAC One Node.

After installation of Oracle Real Application Clusters (Oracle RAC) software, start Oracle Database Configuration Assistant (DBCA).

1. From the Database Operation page, select the option **Create Database**.
2. On the Creation Mode page, select **Advanced Mode**.
3. On the Database Template page, select **Oracle RAC One Node database**.

   Selecting one node deploys Oracle RAC One Node on a single node. Oracle recommends that you select all nodes in the cluster to which you want Oracle RAC One Node to be able to fail over.

   If you select less than two nodes, or create server pools with a cardinality of 2 or more, then DBCA posts a warning message that the configuration you select cannot support failover of the Oracle RAC One Node instance.

   When you create an administrator-managed Oracle RAC One Node database, note that while the database is started on only one of the pool of nodes you installed the binaries, all the candidate servers are placed into the Generic server pool. If the servers are not already in Generic or Free, then this may result in stopping resources that are running on candidate servers.

When you use DBCA to create an Oracle RAC One Node database, a failover service is automatically configured.

3.8 Deleting an Oracle RAC Database Using DBCA

Deleting an Oracle RAC database using Oracle Database Configuration Assistant (DBCA) involves deleting the database and database objects.

DBCA first deletes the database, and then removes the database's initialization parameter files, instances, Optimal Flexible Architecture (OFA) structure, and the Oracle network configuration for the database.

1. Start DBCA on one of your cluster nodes.

   DBCA displays the Operations page, displaying different database deployment options.

2. Select **Delete a database**, and click **Next**.

   DBCA displays a list of all Oracle RAC and single-instance databases running from the Oracle home where DBCA is run.

3. 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 for a user account that has SYSDBA privileges.

4. Select the database to delete, and click **Finish**.

   After you click Finish, DBCA displays a dialog box to confirm the database and instances that you have configured DBCA to delete.

5. 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 the associated instances for this database. DBCA also removes the parameter files and password files.

At this point, you have accomplished the following:

- Deleted the selected Oracle RAC database from the cluster
- Deleted the selected Oracle RAC Database Oracle services on Windows-based platforms
- Deleted high availability services assigned to the Oracle RAC database
- Deleted the Oracle Net configuration for the Oracle RAC database
- Deconfigured Oracle Enterprise Manager for the Oracle RAC database
- Deleted the OFA directory structure for that Oracle RAC database from the cluster
- Deleted the Oracle RAC database data files

Related Topics:

- Starting DBCA
  You can either start the Oracle Database Configuration Assistant (DBCA) utility from the command line or from the Windows Start menu.

4

Oracle Real Application Clusters Postinstallation Procedures

After you have installed the Oracle Database 12*c* release 2 (12.2) with Oracle Real Application Clusters (Oracle RAC) software, there are postinstallation tasks to complete.

Topics:

- Required Postinstallation Tasks
  Perform these tasks after completing your installation.
- Recommended Postinstallation Tasks
  Oracle recommends that you complete these tasks after completing an Oracle RAC installation.
- Product-Specific Postinstallation Tasks
  Many Oracle products and options must be configured before you use them for the first time.
- Configuring the Oracle Home User
  Under certain circumstances you may have to perform additional configuration steps for the Oracle Home user.
- Oracle Configuration Manager Postinstallation Configuration for Oracle RAC
  If you have installed Oracle Configuration Manager, then you must run a script to create a database account to collect database configuration collections.
- Enabling and Disabling Database Options After Installation
  When you install Oracle Database, some options are enabled and the others disabled. You can view the enabled Oracle Database options by querying the V$OPTION view using SQL*Plus.

See Also:

- *Oracle Database Platform Guide for Microsoft Windows* and the product administration and tuning guides for more detailed configuration and tuning information
- *Oracle Database Installation Guide for Microsoft Windows* for additional postinstallation configuration information

4.1 Required Postinstallation Tasks

Perform these tasks after completing your installation.

- Determine If Any Patches Are Required For Your New Software
  If you did not choose to download software options during installation, then after installing Oracle RAC, verify if there are any patches needed for your system.
- Configure Exceptions for the Windows Firewall
  If the Windows Firewall feature is enabled on one or more nodes of your Oracle RAC cluster, then you must create exceptions for Oracle RAC applications and ports.
- Create the OraMTS Service for Microsoft Transaction Server
  Oracle Services for Microsoft Transaction Server (OraMTS) permit Oracle databases to be used as resource managers in Microsoft application coordinated transactions.
- Recompiling All Invalid Objects
  Oracle recommends you run the `utlrp.sql` script after you install, patch, or upgrade a database, to identify and recompile invalid objects.
- Configuring Services on Oracle RAC with CDBs
  During installation, if you select a multitenant container database (CDB), and configure pluggable databases (PDBs), then Oracle recommends that you add services to the PDBs after installation.

4.1.1 Determine If Any Patches Are Required For Your New Software

If you did not choose to download software options during installation, then after installing Oracle RAC, verify if there are any patches needed for your system.

- To determine if any patches are required for your system, review the *Oracle Database Readme* and the *Oracle Database Release Notes for Microsoft Windows*.

4.1.2 Configure Exceptions for the Windows Firewall

If the Windows Firewall feature is enabled on one or more nodes of your Oracle RAC cluster, then you must create exceptions for Oracle RAC applications and ports.

Enabling the Windows Firewall feature on one or more of the nodes in your cluster blocks virtually all TCP network ports to incoming connections. As a result, any Oracle product that listens for incoming connections on a TCP port does not receive any of those connection requests and the clients making those connection requests report errors.

- Review the instructions in *Oracle Grid Infrastructure Installation Guide* for details on how to configure exceptions for the Windows Firewall, if you have not done so already as part of the Oracle Grid Infrastructure installation.

4.1.3 Create the OraMTS Service for Microsoft Transaction Server

Oracle Services for Microsoft Transaction Server (OraMTS) permit Oracle databases to be used as resource managers in Microsoft application coordinated transactions.

OraMTS acts as a proxy for the Oracle database to the Microsoft Distributed Transaction Coordinator (MSDTC). As a result, OraMTS provides client-side connection pooling and allows client components that leverage Oracle to participate in promotable and distributed transactions. In addition, OraMTS can operate with Oracle

databases running on any operating system, given that the services themselves are run on Windows.
On releases earlier than Oracle Database 12*c*, the OraMTS service was created as part of a software-only installation. Starting with Oracle Database 12*c*, you must use a configuration tool to create this service.

To create the OraMTS service after performing a software-only installation of Oracle RAC or after adding a node to an existing cluster, perform the following steps:

1. Open a command window.
2. Change directories to `%ORACLE_HOME%\bin`.
3. Run the `OraMTSCtl` utility to create the OraMTS Service, where *host_name* is a list of nodes on which you want to create the service:

   ```
   C:\..bin> oramtsctl.exe -new -host host_name
   ```

> **See Also:**
>
> *Oracle Services for Microsoft Transaction Server Developer's Guide for Microsoft Windows* for more information about installing OraMTS

4.1.4 Recompiling All Invalid Objects

Oracle recommends you run the `utlrp.sql` script after you install, patch, or upgrade a database, to identify and recompile invalid objects.

The `utlrp.sql` script recompiles all invalid objects. Run the script immediately after installation, to ensure that users do not encounter invalid objects.

1. Start SQL*Plus:

   ```
   sqlplus "/ AS SYSDBA"
   ```

2. Run the `utlrp.sql` script, where *Oracle_home* is the Oracle home path:

   ```
   SQL> @Oracle_home/rdbms/admin/utlrp.sql
   ```

The `utlrp.sql` script automatically recompiles invalid objects in either serial or parallel recompilation, based on both the number invalid objects, and on the number of CPUs available. CPUs are calculated using the number of CPUs (`cpu_count`) multiplied by the number of threads for each CPU (`parallel_threads_per_cpu`). On Oracle Real Application Clusters (Oracle RAC), this number is added across all Oracle RAC nodes.

4.1.5 Configuring Services on Oracle RAC with CDBs

During installation, if you select a multitenant container database (CDB), and configure pluggable databases (PDBs), then Oracle recommends that you add services to the PDBs after installation.

If you do not add services to PDBs, and then the Oracle RAC One Node CDB fails over to another node, or you manually relocate the CDB to another node, then by default, all PDBs associated with that CDB that do not have registered services are restarted in `MOUNTED` state.

PDBs are opened in Read Write mode after failover or relocation only after you have configured the PDBs to have associated services. If you have not associated services to PDBs, then the PDBs remains in `MOUNTED` state when the CDB instance restarts.

- Use the following `srvctl` command syntax, where *cdbname* is the name of the CDB, *service_name* is the name of the service, and *pdbname* is the name of the PDB:

  ```
  srvctl add service -d cdbname -s service_name -pdb pdbname
  ```

After you add services to your PDBs, if you relocate the CDB with which the PDBs are associated, or the CDB fails over, then the PDBs associated with that CDB automatically open in `Read Write` state.

4.2 Recommended Postinstallation Tasks

Oracle recommends that you complete these tasks after completing an Oracle RAC installation.

- Setting Up Additional User Accounts
  You can set up additional user accounts to manage your database.
- Setting the Oracle User Environment Variables
  Unlike on other platforms, do not set `ORACLE_HOME` as a fixed environment variable when running Oracle Database on Windows operating systems. This is because the Oracle software determines where executable files reside at run time.
- Downloading and Installing the ORAchk Health Check Tool
  Download and install the ORAchk utility to perform proactive heath checks for the Oracle software stack.
- About Using CVU Cluster Healthchecks After Installation
  You can use the CVU `healthcheck` command to check your Oracle Clusterware and Oracle Database installations for their compliance with mandatory requirements and best practices guidelines, and to ensure that they are functioning properly.

4.2.1 Setting Up Additional User Accounts

You can set up additional user accounts to manage your database.

- Refer to *Oracle Database Platform Guide for Microsoft Windows* and *Oracle Database Security Guide* for information about setting up additional optional user accounts.

4.2.2 Setting the Oracle User Environment Variables

Unlike on other platforms, do not set `ORACLE_HOME` as a fixed environment variable when running Oracle Database on Windows operating systems. This is because the Oracle software determines where executable files reside at run time.

When you invoke an Oracle executable program on Windows, for example `sqlplus.exe`, the `ORACLE_HOME`, `ORACLE_BASE`, and `ORACLE_SID` variables are determined by the `PATH` environment variable and the location of the executable program (which Oracle home it resides in). To use SQL*Plus to manage a different database or Oracle ASM instance, click the Windows Start button, select the correct Oracle Home for the instance you want to manage, and then select the SQL*Plus utility.

You can use Oracle Universal Installer (OUI) to specify an Oracle home as the default Oracle home and update the PATH environment variable to point to that Oracle home.

> **See Also:**
>
> "Multiple Oracle Home Directories on Windows" for detailed instructions on how to change the default Oracle home.

4.2.3 Downloading and Installing the ORAchk Health Check Tool

Download and install the ORAchk utility to perform proactive heath checks for the Oracle software stack.

ORAchk replaces the RACCheck utility. ORAchk extends health check coverage to the entire Oracle software stack, and identifies and addresses top issues reported by Oracle users. ORAchk proactively scans for known problems with Oracle products and deployments, including the following:

- Standalone Oracle Database
- Oracle Grid Infrastructure
- Oracle Real Application Clusters
- Maximum Availability Architecture (MAA) Validation
- Upgrade Readiness Validations
- Oracle Golden Gate

Oracle is continuing to expand checks, based on customer requests.

ORAchk is supported on Windows Server 2012 and Windows Server 2016 on a Cygwin environment only.

Oracle recommends that you download and run the latest version of ORAchk from My Oracle Support. For information about downloading, configuring and running ORAchk utility, refer to My Oracle Support note 1268927.2:

https://support.oracle.com/epmos/faces/DocContentDisplay?id=1268927.2&parent=DOCUMENTATION&sourceId=USERGUIDE

Related Topics:

- *Oracle ORAchk and EXAchk User's Guide*

4.2.4 About Using CVU Cluster Healthchecks After Installation

You can use the CVU `healthcheck` command to check your Oracle Clusterware and Oracle Database installations for their compliance with mandatory requirements and best practices guidelines, and to ensure that they are functioning properly.

Syntax

```
cluvfy comp healthcheck [-collect {cluster|database}] [-db db_unique_name]
[-bestpractice|-mandatory] [-deviations] [-html] [-save [-savedir directory_path]
```

Options

Option	Description
`-collect [cluster | database]`	Use this option to specify that you want to perform checks for Oracle Clusterware (cluster) or Oracle Database (database). If you do not use the collect flag with the `healthcheck` command, then `cluvfy comp healthcheck` performs checks for both Oracle Clusterware and Oracle Database.
`-db` *`db_unique_name`*	Use this flag to specify checks on the database unique name that you enter after the `–db` option. CVU uses JDBC to connect to the database as the user CVUSYS to verify various database parameters. For this reason, if you want checks to be performed for the database you specify with the `-db` option, then you must first create the CVUSYS user on that database, and grant that user the CVU-specific role, CVUSAPP. You must also grant members of the CVUSAPP role SELECT permissions on system tables. The SQL script `cvusys.sql` is included in the `CVU_home\cv\admin` directory to facilitate the creation of this user. Use this SQL script to create the CVUSYS user on all the databases that you want to verify using CVU. If you use the `-db` option but do not provide a database unique name, then CVU discovers all the Oracle databases on the cluster. To perform best practices checks on these databases, you must create the CVUSYS user on each database, and grant that user the CVUSAPP role with the SELECT privileges needed to perform the best practice checks.
`[-bestpractice | -mandatory] [-deviations]`	• Use the `-bestpractice` option to specify best practice checks • Use the `-mandatory` option to specify mandatory checks • Add the `-deviations` option to specify that you want to see only the deviations from either the best practice recommendations or the mandatory requirements • If you specify neither `-bestpractice` or `-mandatory`, then both best practices and mandatory requirements are displayed. You can specify either the `-bestpractice` or `-mandatory` option, but not both options.

Option	Description
`-html`	Use the `-html` option to generate a detailed report in HTML format. If you specify the `-html` option, and a browser that is recognized by CVU is available on the system, then the browser is started and the report is displayed on the browser when the checks are complete. If you do not specify the `-html` option, then the detailed report is generated in a text file.
`-save [-savedir dir_path]`	Use the `-save` or `-save -savedir` options to save validation reports (`cvuchecdkreport_timestamp.txt` and `cvucheckreport_timestamp.htm`), where *timestamp* is the time and date of the validation report. If you use the `-save` option by itself, then the reports are saved in the path `CVU_home\cv\report`, where *CVU_home* is the location of the CVU executable files. If you use the options `-save -savedir`, and enter a path where you want the CVU reports saved, then the CVU reports are saved in the path you specify.

Example 4-1 Running a Cluster Healthcheck After the Software Installation

To run a healthcheck for your Oracle Grid Infrastructure cluster, to check for any deviations from best practices, and display the results in HTML format, use the following command:

```
C:\> cd app\12.2.0\grid\bin
C:\..bin> cluvfy comp healthcheck -html -bestpractice -deviations
```

Example 4-2 Running a Healthcheck for Oracle RAC Database

To run a healthcheck for your Oracle RAC cluster, to check best practices recommendations and mandatory requirements, and display the results in HTML format, use the following command:

```
C:\> cd app\12.2.0\grid\bin
C:\..bin> cluvfy comp healthcheck -html
```

4.3 Product-Specific Postinstallation Tasks

Many Oracle products and options must be configured before you use them for the first time.

Before using individual Oracle Database 12*c* release 2 (12.2) products or options, refer to the appropriate manual in the product documentation library, which is available online at: http://docs.oracle.com.

Refer to the following topics for information about configuring various products and features after installation.

- Configuring Oracle Database Vault
  Oracle Universal Installer (OUI) installs Oracle Database Vault by default when you install the Oracle RAC software, but requires additional configuration steps.
- Configuring Oracle Label Security
  After installation, you must configure Oracle Label Security in a database before you use it.
- Configuring the OraClrAgnt Service for Oracle Database Extensions for .NET
  Oracle Database Extensions (ODE) for .NET depends on a Windows service to operate properly. This service is called the OraClrAgnt service.
- Configuring Oracle XML DB
  Oracle XML DB is a required component of the Oracle Database installation. However, you must manually configure the FTP and HTTP ports for Oracle XML DB.
- Configure Storage for External Tables, Shared Files, or Directory Objects
  If your Oracle RAC database uses files that are external to the database, then the external files must be located on shared storage that is accessible to all nodes.

4.3.1 Configuring Oracle Database Vault

Oracle Universal Installer (OUI) installs Oracle Database Vault by default when you install the Oracle RAC software, but requires additional configuration steps.

1. Register Oracle Database Vault with the Oracle RAC database.
2. Create the Database Vault Owner user and, optionally, the Database Vault Account Manager administrative user accounts.

Related Topics:

- Perform Postinstallation Configuration for Oracle Database Vault
  After you install the Oracle Database Vault option, you may be required to make additional changes to your database.

4.3.2 Configuring Oracle Label Security

After installation, you must configure Oracle Label Security in a database before you use it.

You can configure Oracle Label Security in two ways: with Oracle Internet Directory integration and without Oracle Internet Directory integration.

Table 4-1 Configuration Options and Requirements for Oracle Label Security

Configuration	Requirements
With Oracle Internet Directory integration	To configure Oracle Label Security with Oracle Internet Directory integration, Oracle Internet Directory must be installed in your environment and the Oracle database must be registered in the directory.

Table 4-1 (Cont.) Configuration Options and Requirements for Oracle Label Security

Configuration	Requirements
Without Oracle Internet Directory integration	If you configure Oracle Label Security (OLS) without Oracle Internet Directory integration, then you cannot configure it to use Oracle Internet Directory at a later stage. To configure Oracle Label Security with Oracle Internet Directory on your database at a later time, you must remove the OLS option on the database and configure the OLS with Oracle Internet Directory integration option.

See Also:

Oracle Label Security Administrator's Guide for information about configuring Oracle Label Security.

4.3.3 Configuring the OraClrAgnt Service for Oracle Database Extensions for .NET

Oracle Database Extensions (ODE) for .NET depends on a Windows service to operate properly. This service is called the OraClrAgnt service.

In versions of Oracle Database prior to Oracle Database 12*c*, this CLR service was created automatically by the installer.

- After installation you use the `OracleClrCtl.exe` utility to create, start, stop, and delete the OraClrAgnt service.

  When you use the `OraClrCtl.exe` utility to create the service, a new service is created named Oracle*Homename*ClrAgent, where *Homename* represents an Oracle Home name. The OraClrAgnt service is configured by this tool using the Oracle Home User account specified during the Oracle Database installation.

See Also:

Oracle Database Extensions for .NET Developer's Guide for Microsoft Windows for more information about using the `OraClrCtl.exe` tool and installing and configuring the OraClrAgnt service

4.3.4 Configuring Oracle XML DB

Oracle XML DB is a required component of the Oracle Database installation. However, you must manually configure the FTP and HTTP ports for Oracle XML DB.

- Refer to *Oracle XML DB Developer's Guide* for information on configuring the FTP and HTTP protocols for Oracle XML DB.

4.3.5 Configure Storage for External Tables, Shared Files, or Directory Objects

If your Oracle RAC database uses files that are external to the database, then the external files must be located on shared storage that is accessible to all nodes.

- Each node must use the same mount point to access the file.

  Acceptable shared file systems include Database File System (DBFS), Oracle Automatic Storage Management Cluster File System (Oracle ACFS), or a supported network file system (NFS) using the Direct NFS Client.
- The database directory object used to write and read files external to the database must point to a shared storage location.
- Each node must use the same mount point for the same shared storage location.

  For example, each node can have a directory object called DPUMP for the mount point `C:\app\acfsmounts\dpump`, which accesses Oracle ACFS shared storage.

> **Note:**
>
> There is no checking of the contents of the external files or directory object specified as part of the external table to ensure that the directory contents are consistent on each node. To avoid unpredictable results, you must ensure that the same file is accessed from all nodes, or that the same file is used on all nodes.

4.4 Configuring the Oracle Home User

Under certain circumstances you may have to perform additional configuration steps for the Oracle Home user.

The additional configuration steps you might need to perform for the Oracle Home user include:

- Creating an OCR wallet for Oracle Home user
- Changing the password for the Oracle Home user

> **See Also:**
>
> *Oracle Database Platform Guide for Microsoft Windows* for details on creating an OCR wallet and changing password for the Oracle Home user.

4.5 Oracle Configuration Manager Postinstallation Configuration for Oracle RAC

If you have installed Oracle Configuration Manager, then you must run a script to create a database account to collect database configuration collections.

You must create this database account in both *Connected* and *Disconnected* modes. The database account stores the PL/SQL procedures that collect the configuration information, and the account owns the database management system (DBMS) job that performs the collection. After the account has been set up, the account is locked because login privileges are no longer required.

- Run the script `installCCRSQL.exe`.

  The script `installCCRSQL.exe` creates an Oracle Configuration Manager user and loads the PL/SQL procedure into the database defined by the `ORACLE_SID` environment variable. For Oracle RAC, you must run the database script against only one instance, such as the local instance on which you performed the installation. However, Oracle Configuration Manager must be installed in all instance homes.

  You can also specify the database SID by using the `-s` option in the command line, as in the following example, where the SID is `orcl`:

  ```
  %ORACLE_HOME%/ccr/admin/scripts/installCCRSQL.exe collectconfig -s orcl
  ```

  By default, the connection to the database is through OS authentication, "`/as sysdba`." To specify a different SYSDBA user and password, you can use these options:

  `-r` *`SYSDBA-USER`*: The login name of the **SYSDBA** user

  `-p` *`SYSDBA-PASSWORD`*: The password for the **SYSDBA** user

> **Note:**
>
> – If you specify the user name without specifying the password (with the `-p` parameter), then the script prompts you to enter the password.
>
> – If you specify only the password without specifying the user name, then the script uses the user SYS by default.

> **See Also:**
>
> *Oracle Configuration Manager Installation and Administration Guide* for more information

4.6 Enabling and Disabling Database Options After Installation

When you install Oracle Database, some options are enabled and the others disabled. You can view the enabled Oracle Database options by querying the V$OPTION view using SQL*Plus.

> **See Also:**
>
> *Oracle Database Administrator's Guide*

If you need to enable or disable a particular database feature for an Oracle home, then use the `chopt` tool. The `chopt` tool is a command-line utility that is located in the *`ORACLE_HOME`*`\bin` directory. The syntax for `chopt` is as follows:

```
chopt [ enable | disable] db_option
```

The possible values for *`db_option`* described in the following table.

Table 4-2 Database Options for Chopt Tool Command

Value	Description
`oaa`	Oracle Advanced Analytics
`olap`	Oracle OLAP
`partitioning`	Oracle Partitioning
`rat`	Oracle Real Application Testing
`ode_net`	Oracle Database Extensions for .NET

Example 4-3 Running the Chopt Tool

To enable the Oracle Data Mining RDBMS Files option in your Oracle binary files:

1. Shut down the database with `srvctl` or SQL*Plus:

   ```
   srvctl stop database -d myDb
   ```

2. Stop the database service, `OracleService`*`SID`*, using the Services program in Control Panel.

3. Run the following commands:

   ```
   cd ORACLE_HOME/bin
   chopt enable dm
   ```

4. Start the database service, `OracleService`*`SID`*, using the Services program in Control Panel.

5. Start up the database:

   ```
   srvctl start database -d myDb
   ```

5
Using Server Pools with Oracle RAC

Understand the server pool concepts in Oracle Real Application Clusters (Oracle RAC) environments.

Topics:

- Policy-Managed Clusters and Capacity Management
  Starting with Oracle Clusterware 11*g* release 2, resources that Oracle Clusterware manages are contained in logical groups of servers called **server pools**.
- Oracle RAC Database and Server Pools
  Oracle RAC databases support two different management styles and deployment models.
- About Creating Server Pools for Oracle RAC Databases
  You can create a server pool with Oracle Database Configuration Assistant while creating an Oracle RAC database, but Oracle recommends that you create server pools before you deploy database software and databases.
- Oracle RAC One Node and Server Pools
  Oracle RAC One Node supports the use of server pools, with some limitations.

5.1 Policy-Managed Clusters and Capacity Management

Starting with Oracle Clusterware 11*g* release 2, resources that Oracle Clusterware manages are contained in logical groups of servers called **server pools**.

Resources are hosted on a shared infrastructure and are contained within server pools. Resources are no longer defined as belonging to a specific instance or node. Instead, the priority of resource requirements is defined. You can use a cluster configuration policy set to provide dynamic management of cluster policies across the cluster.

- Server Pools and Server Categorization
  You can manage servers dynamically using server pools by identifying servers distinguished by particular attributes, a process called **server categorization**.
- Server Pools and Policy-Based Management
  With policy-based management, database administrators specify the server pool (excluding Generic or Free) in which the database resource runs.
- How Server Pools Work
  Server pools divide the cluster into groups of servers hosting singleton and uniform database services and applications.
- Default Server Pools
  When Oracle Clusterware is installed, two server pools are created automatically: *Generic* and *Free*.

> **See Also:**
>
> *Oracle Real Application Clusters Administration and Deployment Guide*

5.1.1 Server Pools and Server Categorization

You can manage servers dynamically using server pools by identifying servers distinguished by particular attributes, a process called **server categorization**.

In this way, you can manage clusters made up of heterogeneous nodes.

5.1.2 Server Pools and Policy-Based Management

With policy-based management, database administrators specify the server pool (excluding Generic or Free) in which the database resource runs.

Policy-based management:

- Enables dynamic capacity assignment when needed to provide server capacity in accordance with the priorities you set with policies
- Enables allocation of resources by importance, so that applications obtain the required minimum resources, whenever possible, and so that lower priority applications do not take resources from more important applications
- Ensures isolation where necessary, so that you can provide dedicated servers in a cluster for applications and databases
- Enables policies to be configured to change pools in accordance with business needs or application demand, so that pools provide the right service at the right time

Applications and databases running in server pools do not share resources. Because server pools do not share resources, they isolate resources where necessary, but enable dynamic capacity assignments as required. Together with role-separated management, this capability addresses the needs of organizations that have standardized cluster environments, but allow multiple administrator groups to share the common cluster infrastructure.

Oracle Clusterware efficiently allocates different resources in the cluster. You need only to provide the minimum and maximum number of nodes on which a resource can run, combined with a level of importance for each resource that is running on these nodes.

> **See Also:**
>
> - The Oracle Clusterware resource reference in *Oracle Clusterware Administration and Deployment Guide*
> - *Oracle Clusterware Administration and Deployment Guide* for details about managing server pools to respond to business or application demand

5.1.3 How Server Pools Work

Server pools divide the cluster into groups of servers hosting singleton and uniform database services and applications.

Server pools distribute a uniform workload (a set of Oracle Clusterware resources) over several servers in the cluster. For example, you can restrict Oracle databases to run only in certain server pools. When you enable role-separated management, you can grant permission to operating system users to use server pools.

You manage server pools that contain Oracle RAC databases with the Server Control (SRVCTL) utility. Use the Oracle Clusterware Control (CRSCTL) utility to manage all other server pools. Only cluster administrators have permission to create top-level server pools.

Top-level server pools:

- Logically divide the cluster
- Are always exclusive, meaning that one server can only reside in one particular server pool at a certain point in time

5.1.4 Default Server Pools

When Oracle Clusterware is installed, two server pools are created automatically: *Generic* and *Free*.

All servers in a new installation are assigned to the Free server pool, initially. Servers move from Free to newly defined server pools automatically.

- The Free Server Pool
  The Free server pool contains servers that are not assigned to any other server pools.
- The Generic Server Pool
  The Generic server pool stores any Oracle Database that is not policy-managed.

5.1.4.1 The Free Server Pool

The Free server pool contains servers that are not assigned to any other server pools.

The attributes of the Free server pool are restricted, as follows:

- `SERVER_NAMES`, `MIN_SIZE`, and `MAX_SIZE` cannot be edited by the user
- `IMPORTANCE` and `ACL` can be edited by the user

5.1.4.2 The Generic Server Pool

The Generic server pool stores any Oracle Database that is not policy-managed.

Additionally, the Generic server pool contains servers with names you specified in the `SERVER_NAMES` attribute of the server pools that list the Generic server pool as a parent server pool.

The Generic server pool's attributes are restricted, as follows:

- No one can modify configuration attributes of the Generic server pool (all attributes are read-only)
- When Oracle Database Configuration Assistant (DBCA) or SRVCTL specifies a server name in the `HOSTING_MEMBERS` resource attribute, Oracle Clusterware only allows it if the server is one of the following:
  - Online and exists in the Generic server pool
  - Online and exists in the Free server pool, in which case Oracle Clusterware moves the server into the Generic server pool
  - Online and exists in any other server pool *and* the user is either a cluster administrator or is allowed to use the server pool's servers, in which case, the server is moved into the Generic server pool
  - Offline and the user is a cluster administrator

5.2 Oracle RAC Database and Server Pools

Oracle RAC databases support two different management styles and deployment models.

Policy-Managed

Deployment is based on server pools, where database services run within a server pool as *singleton* or *uniform* across all of the servers in the server pool. Databases are deployed in one or more server pools and the size of the server pools determine the number of database instances in the deployment. Policy management allows clusters and databases to expand or shrink as requirements change.

A policy-managed database is defined by **cardinality**, which is the number of database instances you want running during normal operations. A policy-managed database runs in one or more database server pools that the cluster administrator creates in the cluster, and it can run on different servers at different times. A database instance starts on all servers that are in the server pools defined for the database.

Clients can connect to a policy-managed database using the same SCAN-based connect string no matter which servers they happen to be running on at the time.

Administrator-Managed

Deployment is based on the Oracle RAC deployment types that existed before Oracle Database 11*g* release 2 (11.2) and requires that you statically configure each database instance to run on a specific node in the cluster, and that you configure database services to run on specific instances belonging to a certain database using the preferred and available designation.

When you review the database resource for an administrator-managed database, you see a server pool defined with the same name as the Oracle database. This server pool is part of a special Oracle-defined server pool called Generic. Oracle RAC manages the Generic server pool to support administrator-managed databases. When you add or remove an administrator-managed database using either SRVCTL or Oracle Database Configuration Assistant (DBCA), Oracle RAC creates or removes the server pools that are members of the Generic server pool.

See Also:

- "Overview of Server Pools and Policy-Based Management" in *Oracle Clusterware Administration and Deployment Guide*
- "Overview of Cluster Configuration Policies and the Policy Set" in *Oracle Clusterware Administration and Deployment Guide*

5.3 About Creating Server Pools for Oracle RAC Databases

You can create a server pool with Oracle Database Configuration Assistant while creating an Oracle RAC database, but Oracle recommends that you create server pools before you deploy database software and databases.

Oracle also recommends that you perform the following steps:

- Enable role separation before you create the first server pool in the cluster.
- Create and manage server pools using configuration policies and a respective policy set.

You can implement role-separated management in one of two ways: Vertical or Horizontal.

Vertical Implementation (Between Layers)

Vertical implementation describes a role separation approach based on different operating system users and groups used for various layers in the technology stack. Permissions on server pools and resources are granted to different users (and groups) for each layer in the stack using access control lists. Oracle Automatic Storage Management (Oracle ASM) offers setting up role separation as part of the Oracle Grid Infrastructure installation based on a granular assignment of operating system groups for specific roles.

Horizontal Implementation (Within One Layer)

Horizontal implementation describes a role separation approach that restricts resource access within one layer using access permissions for resources that are granted using access control lists assigned to server pools and policy-managed databases or applications.

For example, consider an operating system user called `grid`, that installs Oracle Grid Infrastructure and creates two database server pools. The operating system users `ouser1` and `ouser2` must be able to operate within a server pool, but must not be able to modify those server pools and withdraw hardware resources from other server pools either accidentally or intentionally.

See Also:

- "Overview of Cluster Configuration Policies and the Policy Set" in *Oracle Clusterware Administration and Deployment Guide*
- "Role-Separated Management" in *Oracle Clusterware Administration and Deployment Guide*

5.4 Oracle RAC One Node and Server Pools

Oracle RAC One Node supports the use of server pools, with some limitations.

Note the following about Oracle RAC One Node and server pools:

- Oracle RAC One Node runs only in one server pool. This server pool is treated the same as any other server pool.
- Online relocation of an Oracle RAC One Node database instance permits planned migrations of an Oracle RAC One Node database from one node to another node. Relocations must always be within a server pool.

6

Understanding the Oracle RAC Installed Configuration

There are many ways in which the Oracle Real Application Clusters (Oracle RAC) database is different from a single-instance Oracle database.

Topics:

- Understanding the Configured Environment in Oracle RAC
  Oracle Net Configuration Assistant (NETCA) and Database Configuration Assistant (DBCA) configure your environment to meet the requirements for database creation and Oracle Enterprise Manager discovery of Oracle RAC databases.
- Understanding Operating System Privileges Groups
  As an administrator, you often perform special operations such as shutting down or starting up a database, or configuring storage.
- Understanding Time Zone Settings on Cluster Nodes
  Oracle RAC requires that all cluster nodes have the same time zone setting.
- Understanding the Server Parameter File for Oracle RAC
  When you create the database, Oracle Database creates an SPFILE in the file location that you specify. This location can be either an Oracle ASM disk group or a cluster file system.
- Multiple Oracle Home Directories on Windows
  Install each Oracle product in its own Oracle home.
- About Pluggable Databases in Oracle RAC
  A pluggable database (PDB) is a portable collection of schemas, schema objects, and nonschema objects that appears to an Oracle Net client as a non-CDB.
- Database Components Created Using Database Configuration Assistant
  Database Configuration Assistant (DBCA) create various database components.
- About Managing Undo Tablespaces in Oracle RAC
  Oracle Database stores rollback or undo information in undo tablespaces.
- About Initialization Parameter Files
  Oracle recommends using the server parameter file (SPFILE) for storing Oracle Database initialization parameters.
- Oracle Net Services Configuration for Oracle RAC Databases
  When connecting to an Oracle Database, you can use a connect descriptor or a net service name.
- Performance Features of Oracle Net Services and Oracle RAC
  Oracle RAC databases provide the important benefits of connection load balancing and failover.
- Oracle Net Services Configuration Files and Parameters
  Networking elements for the Oracle Database server and clients are preconfigured for most environments.

6.1 Understanding the Configured Environment in Oracle RAC

Oracle Net Configuration Assistant (NETCA) and Database Configuration Assistant (DBCA) configure your environment to meet the requirements for database creation and Oracle Enterprise Manager discovery of Oracle RAC databases.

> **Note:**
>
> Configuration files are created on each node of your cluster database.

Avoid changing host names after you complete your Oracle RAC installation, including adding or deleting domain qualifications. Node names are created from the host names during an Oracle Clusterware installation and are used extensively with database processes. Nodes with changed host names must be deleted from the cluster and added back with the new host names.

6.2 Understanding Operating System Privileges Groups

As an administrator, you often perform special operations such as shutting down or starting up a database, or configuring storage.

Only an administrator, responsible for these administration decisions, should perform these operations. System privileges for Oracle Database or Oracle Automatic Storage Management (Oracle ASM) administration require a secure authentication scheme.

Membership in special operating system groups enables administrators to authenticate to Oracle Database or Oracle ASM through the operating system rather than with a user name and password. This is known as **operating system authentication**. Each Oracle Database in a cluster can have its own operating system privileges groups, so that operating system authentication can be separated for each Oracle Database on a cluster. Because there can be only one Oracle Grid Infrastructure installation on a cluster, there can be only one set of operating system privileges groups for Oracle ASM.

During installation of Oracle Grid Infrastructure and Oracle Database, the installer creates operating system groups. These operating system groups are designated with the logical role of granting operating system authentication for administration system privileges for Oracle Database and Oracle ASM. Oracle Grid Infrastructure uses operating system authentication to manage Oracle Database. To enable this access, you must set the `AUTHENTICATION_SERVICES` parameter in the `sqlnet.ora` file to contain the value NTS.

You can use a single operating system group as the logical group whose members are granted all system privileges for Oracle Database and Oracle ASM, or you can delegate system privileges to two or more operating system groups. Oracle recommends that you designate separate operating system groups for each logical system privilege. Using separate operating system groups enables you to grant one or more subsets of administrator system privileges to database administrators. These database administrators can then perform standard database administration tasks without requiring the SYSDBA system privileges.

See Also:

- "Net Services Profile File (sqlnet.ora)" for more information on how to set the `AUTHENTICATION_SERVICES` parameter.
- *Oracle Grid Infrastructure Installation and Upgrade Guide for Microsoft Windows x64 (64-Bit)* for more information about operating system groups and Oracle Database system privileges.
- *Oracle Automatic Storage Management Administrator's Guide* for more information about operating system groups and Oracle ASM system privileges.

6.3 Understanding Time Zone Settings on Cluster Nodes

Oracle RAC requires that all cluster nodes have the same time zone setting.

During an Oracle Grid Infrastructure for a cluster installation, the installation process determines the time zone setting of the Oracle Installation user on the node where Oracle Universal Installer (OUI) runs. OUI uses that time zone value on all of the nodes as the default time zone setting for all processes that Oracle Clusterware manages. This default setting is used for databases, Oracle ASM, and any other managed processes.

However, if you start an instance with SQL*Plus, you must ensure that the time zone value that Oracle RAC uses is the same as the Oracle Clusterware time zone. You can change the time zone that Oracle Clusterware uses for a database by running the command:

```
srvctl setenv database -env 'TZ=time zone
```

6.4 Understanding the Server Parameter File for Oracle RAC

When you create the database, Oracle Database creates an SPFILE in the file location that you specify. This location can be either an Oracle ASM disk group or a cluster file system.

All instances in the cluster database use the same SPFILE at startup. Because the SPFILE is a binary file, do not directly edit the SPFILE with an editor. Instead, change SPFILE parameter settings using Oracle Enterprise Manager or `ALTER SYSTEM` SQL statements.

See Also:

Oracle Real Application Clusters Administration and Deployment Guide for information about creating and modifying SPFILE

6.5 Multiple Oracle Home Directories on Windows

Install each Oracle product in its own Oracle home.

The value for `%ORACLE_BASE%` is stored in the registry (for example, in `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0`). The values for `ORACLE_HOME` and `ORACLE_SID` are also stored in the registry. Symbolic links for these directories, like those used on UNIX platforms, are not supported on Windows platforms

Oracle Universal Installer (OUI) records the values for environment variables such as `ORACLE_BASE`, `ORACLE_HOME`, and `ORACLE_SID` in the registry and also updates the value for the `PATH` environment variable for the user performing the installation. In Linux and UNIX systems, you must manually set these environment variables in the user session or user profile.

- Changing the Current Setting for Oracle Home
  Use Oracle Universal Installer (OUI) to change the current Oracle home.

Related Topics:

- Running Oracle Net Configuration Assistant Using Response Files
  You can run Oracle Net Configuration Assistant (NETCA) in silent mode to configure and start an Oracle Net listener on the system, configure naming methods, and configure Oracle Net service names.

> **See Also:**
>
> *Oracle Database Platform Guide for Microsoft Windows* for more information about the registry keys

6.5.1 Changing the Current Setting for Oracle Home

Use Oracle Universal Installer (OUI) to change the current Oracle home.

This procedure changes the value of the default `ORACLE_HOME` variable in the registry to the value you selected. It also ensures that the `%ORACLE_HOME%\bin` directories for each product are listed in the correct order in your `PATH` environment variable.

1. Start the Oracle Universal Installer.
2. Click the **Installed Products** button.
3. Click the **Environment** tab at the top of the window.
4. Move the Oracle home directory that you want as your default to the top of the list.
5. Apply the changes, and exit the installer.

Related Topics:

- Running Oracle Net Configuration Assistant Using Response Files
  You can run Oracle Net Configuration Assistant (NETCA) in silent mode to configure and start an Oracle Net listener on the system, configure naming methods, and configure Oracle Net service names.

6.6 About Pluggable Databases in Oracle RAC

A pluggable database (PDB) is a portable collection of schemas, schema objects, and nonschema objects that appears to an Oracle Net client as a non-CDB.

PDBs can be plugged into to CDBs. A CDB can contain multiple PDBs. Each PDB appears on the network as a separate database.

Starting in Oracle Database 12*c*, you must create a database as either a multitenant container database (CDB) or as an Oracle database that is non-CDB. This also applies to Oracle RAC databases. The only difference to the installation process is to choose whether to create the Oracle RAC database as a CDB or a non-CDB.

If you create an Oracle RAC database as a CDB and plug one or more PDBs into the CDB, then, by default, a PDB is not started automatically on any instance of the Oracle RAC CDB. With the first dynamic database service assigned to the PDB (other than the default database service which has the same name as the database name), the PDB is made available on those instances on which the service runs.

Whether a PDB is available on more than one instance of an Oracle RAC CDB is typically managed by the services running on the PDB. You can manually enable PDB access on each instance of an Oracle RAC CDB by starting the PDB manually on that instance.

See Also:

- *Oracle Database Concepts* for more information about PDBs
- *Oracle Database Administrator's Guide* for more information about managing PDBs
- *Oracle Real Application Clusters Administration and Deployment Guide* for information specific to the administration of Oracle RAC CDBs

6.7 Database Components Created Using Database Configuration Assistant

Database Configuration Assistant (DBCA) create various database components.

- About Tablespaces and Data Files
  For both single-instance and cluster database environments, Oracle Database is divided into smaller logical areas of space known as **tablespaces**.
- About Control Files
  The database is configured with two control files that must be stored on shared storage.
- About Online Redo Log Files
  Each database instance must have at least two online redo log files. The online redo log files for a database instance are called the **redo thread**.

6.7.1 About Tablespaces and Data Files

For both single-instance and cluster database environments, Oracle Database is divided into smaller logical areas of space known as **tablespaces**.

Each tablespace corresponds to one or more data files on the shared storage.

Table 6-1 Tablespace Names Used with Oracle Real Application Clusters Databases

Tablespace Name	Contents
`SYSAUX`	An auxiliary system tablespace that contains the `DRSYS` (contains data for Oracle Text), `CWMLITE` (contains the OLAP schemas), `XDB` (for XML features), `ODM` (for Oracle Data Mining), and `INDEX` schemas
`SYSTEM`	Consists of the data dictionary, including definitions of tables, views, and stored procedures needed by the database. Oracle Database automatically maintains information in this tablespace.
`TEMP`	Contains temporary tables and indexes created during SQL statement processing. You may need to expand this tablespace if you run 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*	Contains undo tablespaces for each instance that Oracle Database Configuration Assistant creates for automatic undo management
`USERS`	Consists of application data. As you create and enter data into tables, Oracle Database fills this space with your data.

You cannot alter these tablespace names when using the preconfigured database configuration option from Oracle Universal Installer (OUI). 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 on shared file systems. The data file names created by the preconfigured database configuration options vary by storage type such as Oracle ASM or a cluster file system.

See Also:

- *Oracle Database Administrator's Guide* for more information about the `SYSTEM`, `SYSAUX`, and other tablespaces
- "Selecting an Oracle Database Configuration Type" for more information about database configuration types.

6.7.2 About Control Files

The database is configured with two control files that must be stored on shared storage.

Every database must have one unique control file; any additional control files configured for the database are identical copies of the original control file. If a control file becomes unusable, then the database instance fails when it attempts to access the damaged control file. By multiplexing (creating multiple copies of) a control file on different disks, the database can achieve redundancy and thereby avoid a single point of failure.

See Also:

- "Overview of Control Files" in *Oracle Database Concepts*
- "Managing Control Files" in *Oracle Database Administrator's Guide*

6.7.3 About Online Redo Log Files

Each database instance must have at least two online redo log files. The online redo log files for a database instance are called the **redo thread**.

Each Oracle RAC database instance has its own redo thread to avoid contention for a single set of online redo log files. In case of instance failure, the online redo log files must be accessible by the surviving instances. Therefore, the online redo log files for an Oracle RAC database must be placed on shared storage or Oracle ASM. If you use a file system for storage, then the file system must be a shared or cluster file system.

The file names of the redo log files that are created with the preconfigured database configuration option vary by storage type.

See Also:

- *Oracle Database Concepts* for more information about the online redo log files
- *Oracle Real Application Clusters Administration and Deployment Guide* for more information about storage for online redo log files

6.8 About Managing Undo Tablespaces in Oracle RAC

Oracle Database 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 management mode for undo tablespaces that is easier to administer than manual undo management.

When Oracle ASM and Oracle Managed Files are used along with Automatic Undo Management, an instance that is started for the first time, and thus does not have an undo tablespace, has its undo tablespace created for it by another instance automatically. The same is also true for online redo logs.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about automatic undo management
> - *Oracle Real Application Clusters Administration and Deployment Guide* for more information about managing undo tablespaces

6.9 About Initialization Parameter Files

Oracle recommends using the server parameter file (SPFILE) for storing Oracle Database initialization parameters.

Oracle recommends that you store all SPFILEs on Oracle ASM, including the Oracle ASM SPFILE. SPFILEs must be located on shared storage; all instances in a cluster database can access this parameter file.

> **See Also:**
>
> *Oracle Real Application Clusters Administration and Deployment Guide* for more information about the creation and use of parameter files

6.10 Oracle Net Services Configuration for Oracle RAC Databases

When connecting to an Oracle Database, you can use a connect descriptor or a net service name.

For Oracle RAC databases, you can also use the single client access name (SCAN) to connect to any available instance of the Oracle RAC database. Users can access an Oracle RAC database using a client/server configuration or through one or more middle tiers, with or without connection pooling.

- Database Services for an Oracle RAC Database
  Each database is represented by one or more services. A service is identified by a service name, such as `sales.example.com`.
- Naming Methods and Connect Descriptors
  Each net service name is associated with a connect descriptor. A **connect descriptor** provides the location of the database and the name of the database service.
- Easy Connect Naming Method
  The Easy Connect naming method eliminates the need to look up service names in the tnsnames.ora file or other repository for TCP/IP environments.
- Understanding SCANs
  The SCAN is a domain name registered to at least one and up to three IP addresses, either in domain name service (DNS) or in Grid Naming Service (GNS).

- About Connecting to an Oracle RAC Database Using SCANs
  Oracle recommends that you configure Oracle RAC database clients to use the SCAN to connect to the database instead of configuring the `tnsnames.ora` file.
- About Listener Configuration for an Oracle RAC Database
  An Oracle RAC database uses multiple listeners to direct client requests to the available instances.
- About Service Registration for an Oracle RAC Database
  An Oracle Database 12*c* release 2 (12.2) database service automatically registers with the listeners specified in the database initialization parameters `LOCAL_LISTENER` and `REMOTE_LISTENER`.
- How Database Connections are Created When Using SCANs
  Based on the environment, the following actions occur when you use a SCAN to connect to an Oracle RAC database using a service name.

6.10.1 Database Services for an Oracle RAC Database

Each database is represented by one or more services. A service is identified by a service name, such as `sales.example.com`.

A client uses a service name to identify the database it must access. During installation, Oracle RAC databases are configured with a default database service that has the same name as the database. This service can be used for performing database management tasks. Additional services must be created for client and application connections to the database.

A service name can be associated with multiple database instances, and an instance can be associated with multiple services. The listener acts as a mediator between the client and database instances and routes the connection request to the appropriate instance. Clients connecting to a service do not have to specify which instance they want to connect to.

6.10.2 Naming Methods and Connect Descriptors

Each net service name is associated with a connect descriptor. A **connect descriptor** provides the location of the database and the name of the database service.

A connect descriptor includes one or more protocol addresses of the listener and the connect information for the destination service.

The information needed to use a service name to create a database connection can be stored in a repository, used by one or more naming methods. A **naming method** is a resolution method used by a client application to resolve a service name to a connect descriptor. Oracle Net Services offers several types of naming methods that support localized configuration on each client, or centralized configuration that can be accessed by all clients in the network.

6.10.3 Easy Connect Naming Method

The Easy Connect naming method eliminates the need to look up service names in the tnsnames.ora file or other repository for TCP/IP environments.

With Easy Connect, clients use a connect string for a simple TCP/IP address, which consists of a host name, and an optional port and service name. If you use this

method, then no naming or directory system is required. See "Example 6-1" for an example.

Networking elements for the Oracle Database server and clients are preconfigured for most environments. The Easy Connect naming method is enabled by default and does not require a repository. If you use a naming method other than Easy Connect, then additional configuration of Oracle Net Services may be required.

6.10.4 Understanding SCANs

The SCAN is a domain name registered to at least one and up to three IP addresses, either in domain name service (DNS) or in Grid Naming Service (GNS).

- About the SCAN
  During the installation of Oracle Grid Infrastructure, several Oracle Clusterware resources are created for the SCAN.
- About SCAN VIP Addresses
  SCAN virtual IP addresses (VIPs) function like node VIPs. However, unlike node VIPs, SCAN VIPs can run on any node in the cluster.
- About SCAN Listeners
  During Oracle Grid Infrastructure installation, SCAN listeners are created for as many IP addresses as there are SCAN VIP addresses assigned to resolve to the SCAN.

6.10.4.1 About the SCAN

During the installation of Oracle Grid Infrastructure, several Oracle Clusterware resources are created for the SCAN.

- A SCAN virtual IP (VIP) is created for each IP address that Oracle Single Client Access Name (SCAN) resolves to
- A SCAN listener is created for each SCAN VIP
- A dependency on the SCAN VIP is configured for the SCAN listener

SCANs are defined using one of two options:

- The SCAN is defined in DNS

  If you configure a SCAN manually, and use DNS for name resolution, then your network administrator should create a single name for the SCAN that resolves to three IP addresses on the same network as the public network for the cluster. The SCAN name must be resolvable without the domain suffix (for example, the address `sales1-scan.example.com` must be resolvable using `sales1-scan`). The SCAN must not be assigned to a network interface, because Oracle Clusterware resolves the SCAN.

  The default SCAN is `cluster_name-scan.domain_name`. For example, in a cluster that does not use GNS, if your cluster name is `sales1`, and your domain is `example.com`, then the default SCAN address is `sales1-scan.example.com:1521`.
- The SCAN is defined in GNS

  When using GNS and DHCP, Oracle Clusterware configures the VIP addresses for the SCAN name that is provided during cluster configuration. The node VIP and the three SCAN VIPs are obtained from the DHCP server when using GNS. If a new server joins the cluster, then Oracle Clusterware dynamically obtains the

required VIP address from the DHCP server, updates the cluster resource, and makes the server accessible through GNS.

Oracle recommends that you configure clients connecting to the cluster to use the SCAN name, rather than node VIPs used in releases before Oracle Grid Infrastructure 11*g* Release 2 (11.2). Clients connecting to Oracle RAC databases using SCANs do not have to be configured with addresses of each node that hosts a particular database or database instance. For example, if you configure policy-managed server pools for a cluster, then connecting to the database using a SCAN enables connections to server pools in that database, regardless of which nodes are allocated to the server pool. You can add or remove nodes from the database without having to reconfigure clients connecting to the database.

> **See Also:**
>
> *Oracle Grid Infrastructure Installation Guide* for your platform for more information about SCAN configuration and requirements

6.10.4.2 About SCAN VIP Addresses

SCAN virtual IP addresses (VIPs) function like node VIPs. However, unlike node VIPs, SCAN VIPs can run on any node in the cluster.

Clients (users or applications) that connect using a SCAN instead of a node VIP name or address do not have to update the list of node names or addresses in their local `tnsnames.ora` file when nodes are added to or removed from the cluster, or when a database instance runs on a different node.

> **Note:**
>
> Configuring three SCAN VIPs in DNS does not by itself ensure failover of connections. Instead, the Oracle Client uses the returned SCAN VIPs to failover the connection request to a different SCAN listener. If the connection attempt to a SCAN VIP fails, then the client uses the next returned SCAN VIP address to connect. For this reason, Oracle recommends that you use Oracle Client 11*g* Release 2 or later clients for connections that use the SCAN.

If you use GNS for name resolution, then you only provide the SCAN name during installation (for example, `sales1-scan`). GNS obtains DHCP address leases for three IP addresses and resolves these addresses to the SCAN. The GNS daemon listens for registrations. When a SCAN VIP starts on a node, it registers its addresses with GNS.

Service requests to the cluster domain that GNS manages are routed to the GNS VIP address, which routes these requests to the GNS daemon for the cluster. When GNS receives a request from a DNS for the SCAN, it returns the registered addresses of the SCAN listeners to the DNS. The DNS then returns the three SCAN VIP addresses to the client.

> **See Also:**
>
> *Oracle Clusterware Administration and Deployment Guide* for more information about SCAN names, listeners, and client service requests

6.10.4.3 About SCAN Listeners

During Oracle Grid Infrastructure installation, SCAN listeners are created for as many IP addresses as there are SCAN VIP addresses assigned to resolve to the SCAN.

Oracle recommends that the SCAN resolves to three VIP addresses, to provide high availability and scalability. If the SCAN resolves to three addresses, then three SCAN VIPs and three SCAN listeners are created.

Each SCAN listener depends on its corresponding SCAN VIP. The SCAN listeners cannot start until the SCAN VIP is available on a node.

The addresses for the SCAN listeners resolve either through an external domain name service (DNS), or through the Grid Naming Service (GNS) within the cluster. SCAN listeners and SCAN VIPs can run on any node in the cluster. If a node where a SCAN VIP is running fails, then the SCAN VIP and its associated listener fails over to another node in the cluster. If the number of available nodes within the cluster falls to less than three, then one server hosts two SCAN VIPs and SCAN listeners. The SCAN listener also supports HTTP protocol for communication with Oracle XML Database (XDB).

> **See Also:**
>
> *Oracle Clusterware Administration and Deployment Guide* for more information about SCAN listeners

6.10.5 About Connecting to an Oracle RAC Database Using SCANs

Oracle recommends that you configure Oracle RAC database clients to use the SCAN to connect to the database instead of configuring the `tnsnames.ora` file.

Clients configured to connect to the cluster using node VIP addresses for Oracle RAC releases earlier than Oracle Database 11*g* Release 2 can continue to use their existing connection addresses. Using the SCAN is not required. When an earlier release of Oracle Database is upgraded, the database is not only registered with the local listeners, but is also registered with the SCAN listeners, allowing clients to start using the SCAN to connect to that database.

If the SCAN is resolved by DNS, then DNS returns all three SCAN VIP addresses to the client. If the SCAN is resolved by GNS, then DNS zone delegation sends the lookup request to GNS, which then returns all three SCAN VIP addresses to the client.

Oracle Database 12*c* database clients use SCAN to connect to the database. Oracle recommends against using the easy connect method with SCAN because the easy connect method does not have the ability to specify timeouts and retries for connection establishment. Instead, applications must use an Oracle Net connect descriptor with the following format:

```
(DESCRIPTION =
      (CONNECT_TIMEOUT=90) (RETRY_COUNT=20)(RETRY_DELAY=3)
(TRANSPORT_CONNECT_TIMEOUT=3)
                   ( ADDRESS = (PROTOCOL = TCP)(HOST=scan)(PORT=1521))
                      (CONNECT_DATA=(SERVICE_NAME=service_name)))
```

scan represents the SCAN for your cluster. If you do not specify a port number, then the default value of 1521 is used for the TCP port identifier. The *service_name* is the name of a dynamic database service.

The client then uses one of the returned SCAN VIP addresses to contact a SCAN listener. When a SCAN listener receives a connection request from a client, the SCAN listener identifies the least loaded instance in the cluster that provides the requested service. It then redirects the connection request to the local listener on the node where the least loaded instance is running, and the client is given the local listener address. The local listener then creates the connection to the database instance.

Example 6-1 Connecting to Oracle RAC Using an Oracle Net Connect Descriptor

If the Oracle RAC database runs on a cluster for which the SCAN is `sales1-scan.mycluster.example.com`, then you can submit a connection request for the database service `oltp.example.com` by using a connect descriptor similar to the following:

```
(DESCRIPTION =
      (CONNECT_TIMEOUT=90) (RETRY_COUNT=20)(RETRY_DELAY=3)
(TRANSPORT_CONNECT_TIMEOUT=3)
                 ( ADDRESS = (PROTOCOL = TCP)(HOST=sales1-scan.mycluster.example.com)
(PORT=1521))
                           (CONNECT_DATA=(SERVICE_NAME=oltp.example.com)))
```

If the SCAN is resolved by DNS, then DNS returns all three SCAN VIP addresses to the client. If the SCAN is resolved by GNS, then DNS zone delegation sends the lookup request to GNS, which then returns all three SCAN VIP addresses to the client. The client then uses one of the returned SCAN VIP addresses to contact a SCAN listener.

When a SCAN listener receives a connection request from a client, the SCAN listener identifies the least loaded instance in the cluster that provides the requested service. It then redirects the connection request to the local listener on the node where the least loaded instance is running, and the client is given the local listener address. The local listener then creates the connection to the database instance.

6.10.6 About Listener Configuration for an Oracle RAC Database

An Oracle RAC database uses multiple listeners to direct client requests to the available instances.

An Oracle database receives connection requests through the local listener. The local listener brokers a client request, handing off the request to the server. The listener is configured with a protocol address, and clients configured with the same protocol address can send connection requests to the listener. When a connection is established, the client and Oracle database communicate directly with one another.

The local listener, or default listener, is located in the Grid home when you have Oracle Grid Infrastructure installed. Local listeners are configured to respond to database connection requests, and to non-database connection requests, such as

external procedures or Oracle XML Database (XDB) requests. When the database starts, the Database Agent process (`oraagent.exe`, previously known as `racgimon`) sets the `LOCAL_LISTENER` parameter to a connect descriptor that does not require an Oracle Net service name. The value for `LOCAL_LISTENER` is computed to be the endpoints of the Grid home listeners.

You can configure multiple Oracle Database listeners, each with a unique name, in one `listener.ora` file. Multiple listener configurations for database listeners are possible because each of the top-level configuration parameters has a suffix of the listener name or is the listener name itself. To configure a database to register with multiple local listeners, you must manually modify the `LOCAL_LISTENER` parameter.

> **Note:**
>
> Oracle recommends running only one listener for each node in most customer environments.

For an Oracle RAC database, the database parameter `REMOTE_LISTENER` identifies the SCAN listeners. The database registers with the local and SCAN listeners by using the connect description information contained in these parameters. Oracle Database 11*g* Release 2 and later instances only register with SCAN listeners as remote listeners. Upgraded databases register with SCAN listeners as remote listeners, and also continue to register with all node listeners.

The `REMOTE_LISTENER` parameter for an Oracle RAC database is always set to the SCAN address. For example, if the SCAN for the cluster is myscan, and the GNS subdomain for the cluster is `mycluster.example.com`, then the `REMOTE_LISTENER` parameter has the following value:

```
myscan.mycluster.example.com:1521
```

> **Note:**
>
> Do not set the `REMOTE_LISTENER` parameter for an Oracle RAC database to an Oracle Net alias that has a single address that uses the SCAN for the host name (`HOST=`*`scan`*).

6.10.7 About Service Registration for an Oracle RAC Database

An Oracle Database 12*c* release 2 (12.2) database service automatically registers with the listeners specified in the database initialization parameters `LOCAL_LISTENER` and `REMOTE_LISTENER`.

During registration, the listener registration (LREG) process sends information such as the service name, instance names, and workload information to the listeners. This feature is called **service registration**.

When a listener starts after the Oracle instance starts, and the listener is available for service registration, registration does not occur until the next time the Oracle Database LREG process starts its discovery routine. By default, the LREG discovery routine is started every 60 seconds. To override the 60-second delay, use the SQL statement `ALTER SYSTEM REGISTER`. This statement forces LREG to register the service immediately

> **Note:**
>
> Oracle recommends that you create a script to run the `ALTER SYSTEM REGISTER` statement immediately after starting the listener. If you run this statement when the instance is registered and all services are currently registered, or while the listener is down, then the statement has no effect.

> **See Also:**
>
> *Oracle Database Net Services Administrator's Guide* for more information about service registration

6.10.8 How Database Connections are Created When Using SCANs

Based on the environment, the following actions occur when you use a SCAN to connect to an Oracle RAC database using a service name.

The numbered actions correspond to the arrows shown in the figure displayed after the steps.

1. The LREG process of each instance registers the database services with the default listener on the local node and with each SCAN listener, which is specified by the `REMOTE_LISTENER` database parameter. The listeners are dynamically updated on the amount of work being handled by the instances and dispatchers.
2. The client issues a database connection request using a connect descriptor of the form:

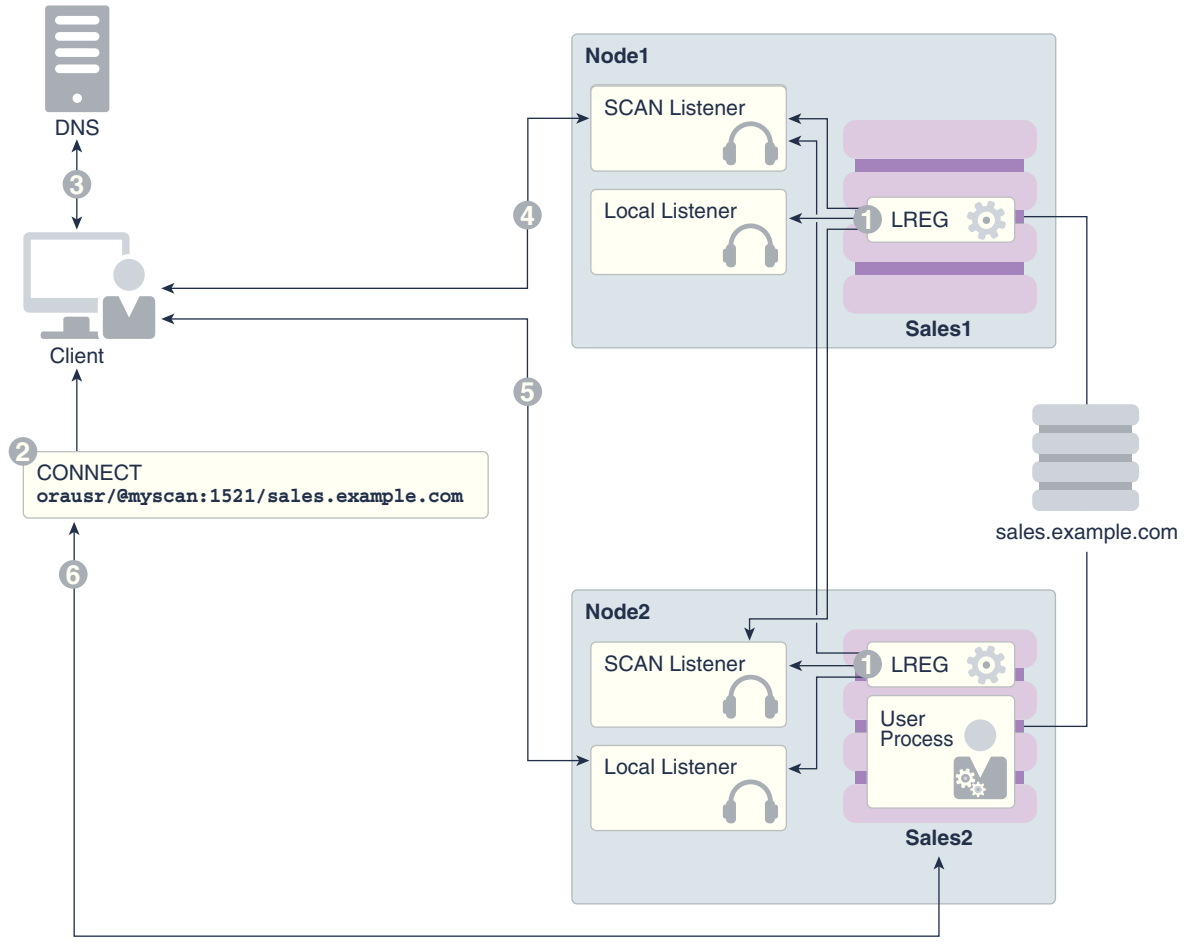
   ```
   orausr/@scan_name:1521/webapp
   ```

   > **Note:**
   >
   > If you use the Easy Connect naming method, then ensure that the `sqlnet.ora` file on the client contains `EZCONNECT` in the list of naming methods specified by the `NAMES.DIRECTORY_PATH` parameter.

3. The client uses DNS to resolve *scan_name*. After DNS returns the three addresses assigned to the SCAN, the client sends a connect request to the first IP address. If the connect request fails, then the client attempts to connect using the next IP address.
4. When the connect request is successful, the client connects to a SCAN listener for the cluster that hosts the `sales` database and has an instance offering the `webapp` service, which in this example is `sales1` and `sales2`. The SCAN listener compares the workload of the instances `sales1` and `sales2` and the workload of the nodes on which they run. If the SCAN listener determines that `node2` is less loaded than `node1`, then the SCAN listener selects `node2` and sends the address for the local listener on that node back to the client.
5. The client connects to the local listener on `node2`. The local listener starts a dedicated server process for the connection to the database.

6. The client connects directly to the dedicated server process on `node2` and accesses the `sales2` database instance.

Figure 6-1 Load Balancing Actions for Oracle RAC Connections That Use SCAN



6.11 Performance Features of Oracle Net Services and Oracle RAC

Oracle RAC databases provide the important benefits of connection load balancing and failover.

- Load Balancing of Connections to Oracle RAC Databases
  There are two types of load balancing that you can implement for an Oracle RAC database: client-side and server-side load balancing.
- Connection Failover for Oracle RAC Databases
  When a client issues a connection request using SCAN, the three SCAN addresses are returned to the client.
- Shared Server Configuration for an Oracle RAC Database
  Standalone Oracle databases perform load balancing by distributing connections among the shared server dispatcher processes.

6.11.1 Load Balancing of Connections to Oracle RAC Databases

There are two types of load balancing that you can implement for an Oracle RAC database: client-side and server-side load balancing.

Services coordinate their sessions by registering their **workload**, or the amount of work they are currently handling, with the local listener and the SCAN listeners. Clients are redirected by the SCAN listener to a local listener on the least-loaded node that is running the instance for a particular service. This feature is called **load balancing**. The local listener either directs the client to a dispatcher process (if you configured the database to use shared servers), or directs the client to a dedicated server process.

Client-side load balancing balances the connection requests across the listeners. With server-side load balancing, the SCAN listener directs a connection request to the best instance currently providing the service by using the load balancing advisory.

> **See Also:**
>
> - *Oracle Grid Infrastructure Installation Guide* for more information about SCAN and its configuration
> - *Oracle Real Application Clusters Administration and Deployment Guide* for more information about failover, load balancing, and the load balancing advisory

6.11.2 Connection Failover for Oracle RAC Databases

When a client issues a connection request using SCAN, the three SCAN addresses are returned to the client.

If the first address fails, then the connection request to the SCAN fails over to the next address. Using multiple addresses allows a client to connect to an instance of the database even if the initial instance has failed.

Oracle RAC provides failover with the node VIP addresses by configuring multiple listeners on multiple nodes to manage client connection requests for the same database service. If a node fails, then the service connecting to the VIP is relocated transparently to a surviving node, enabling fast notification of the failure to the clients connecting through the VIP. If the application and client are configured with transparent application failover options, then the client is reconnected to the surviving node.

6.11.3 Shared Server Configuration for an Oracle RAC Database

Standalone Oracle databases perform load balancing by distributing connections among the shared server dispatcher processes.

By default, Oracle Database Configuration Assistant (DBCA) configures your Oracle RAC database with dedicated servers, not shared servers. However, if you select the shared server option when using DBCA, then DBCA configures shared servers. Oracle RAC uses both dedicated and shared server processing when shared servers are configured.

> **See Also:**
>
> *Oracle Database Net Services Administrator's Guide* for more information about shared server configurations

6.12 Oracle Net Services Configuration Files and Parameters

Networking elements for the Oracle Database server and clients are preconfigured for most environments.

The Easy Connect naming method is enabled by default and does not require a repository. If you use a naming method other than Easy Connect, then additional configuration of Oracle Net Services may be required.

- Database Initialization Parameters for Database Service Registration
  An Oracle Database 12*c* release 2 (12.2) database service automatically registers with the listeners specified in the `LOCAL_LISTENER` and `REMOTE_LISTENER` parameters.
- Net Service Names and the tnsnames.ora File
  The installation process creates a `tnsnames.ora` file on each node.
- Net Service Names Created by DBCA
  Oracle Database Configuration Assistant creates net service names for connecting to the database instances.
- Listener Configuration and the listener.ora File
  In Oracle RAC environments, Oracle recommends that you let the Oracle Agent manage Oracle listeners for Oracle Databases.
- Net Services Profile File (sqlnet.ora)
  Oracle Universal Installer starts Oracle Net Configuration Assistant (NETCA) after the database software is installed. NETCA creates the Oracle Net Services profile, or the `sqlnet.ora` file.

6.12.1 Database Initialization Parameters for Database Service Registration

An Oracle Database 12*c* release 2 (12.2) database service automatically registers with the listeners specified in the `LOCAL_LISTENER` and `REMOTE_LISTENER` parameters.

During registration, the listener registration (LREG) process sends information such as the service name, instance names, and workload information to the listeners.

When a listener starts after the Oracle instance starts, and the listener is available for service registration, registration does not occur until the next time the Oracle Database LREG process starts its discovery routine. By default, the LREG discovery routine is started every 60 seconds. To override the 60-second delay, use the SQL statement `ALTER SYSTEM REGISTER`. This statement forces LREG to register the service immediately.

> **Note:**
>
> Oracle recommends that you create a script to run the `ALTER SYSTEM REGISTER` statement immediately after starting the listener. If you run this statement when the instance is registered and all services are currently registered, or while the listener is down, then the statement has no effect.

> **See Also:**
>
> *Oracle Database Net Services Administrator's Guide* for more information about service registration

6.12.2 Net Service Names and the tnsnames.ora File

The installation process creates a `tnsnames.ora` file on each node.

The `tnsnames.ora` file acts as a repository of net service names. Each net service name is associated with a connect identifier. A **connect identifier** is an identifier that maps a user-defined name 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` parameter, with the value set to the name of a database service

> **Note:**
>
> The `SERVICE_NAME` parameter you use in the `tnsnames.ora` file is singular, because you can specify only one service name. The `SERVICE_NAME` parameter is different from the `service_names` database initialization parameter. The `service_names` database parameter defaults to the global database name, a name comprising the `db_name` and `db_domain` parameters in the initialization parameter file. When you add service names using SRVCTL or Oracle Enterprise Manager Cloud Control, it lists additional cluster-managed services for the database.

The `tnsnames.ora` file is located in both the *`Grid_home`*`\network\admin` and *`Oracle_home`*`\network\admin` directories. By default, the `tnsnames.ora` file is read from the Grid home when Oracle Grid Infrastructure is installed.

With Oracle Clusterware 11*g* Release 2 and later, the listener association no longer requires `tnsnames.ora` file entries. The listener associations are configured as follows:

- Oracle Database Configuration Assistant (DBCA) no longer sets the `LOCAL_LISTENER` parameter. The Oracle Clusterware agent that starts the database sets the `LOCAL_LISTENER` parameter dynamically, and it sets it to the actual value, not an alias. So `listener_`*`alias`* entries are no longer needed in the `tnsnames.ora` file.

- The `REMOTE_LISTENER` parameter is configured by DBCA to reference the SCAN and SCAN port, without any need for a `tnsnames.ora` entry. Oracle Clusterware uses the Easy Connect naming method with *`scanname:scanport`*, so no listener associations for the `REMOTE_LISTENER` parameter are needed in the `tnsnames.ora` file.

Example 6-2 Adding a Second Listener to an Oracle RAC Database

If you created a database named `orcl1`, to add a second listener, listening on port 2012, use a command similar to the following command to have the database register with both listeners on startup:

```
SQL> alter system set local_listener='(DESCRIPTION=(
ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.0.61)(PORT=1521))
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.0.61)(PORT=2012))))'
scope=BOTH SID='OCRL1';
```

See Also:

- *Oracle Database Administrator's Guide*
- *Oracle Database Net Services Administrator's Guide* for more information about the `tnsnames.ora` file

6.12.3 Net Service Names Created by DBCA

Oracle Database Configuration Assistant creates net service names for connecting to the database instances.

- Net Service Names for Database Connections
  Clients that connect to any instance of Oracle RAC use the SCAN in the connect descriptor.
- Net Service Names for Instance Connections
  Clients that connect to a particular instance of the database use the net service name for the instance.

6.12.3.1 Net Service Names for Database Connections

Clients that connect to any instance of Oracle RAC use the SCAN in the connect descriptor.

You can also use net service names to connect to Oracle RAC. The default database service created by Oracle Database Configuration Assistant (DBCA) enables Oracle Enterprise Manager to discover an Oracle RAC database, and must not be used for client connections.

If you use DBCA to create an Oracle RAC database that is a multitenant container database (CDB), then DBCA creates a database service that has the same name as the database. Clients that use this database service can connect to any database instance for the Oracle RAC CDB. However, if you use DBCA to add a pluggable database (PDB) to an existing CDB, then DBCA does not create a database service for the new PDB.

The net service name does not require a fully qualified domain name for the server on which a database, database instance, or listener runs. SCANs are resolved by the

DNS or GNS, which returns three addresses to the client. The client then submits connection requests to each address in succession until a connection is made.

Example 6-3 Net Service Name Entry for a Database Connection

This example shows a connect descriptor that is used in a `tnsnames.ora` file. The connect identifier in this case is the same as the cluster domain, `mycluster.example.com`. Instead of specifying the address for an individual server, a virtual Internet Protocol (VIP) address, or a cluster node name, the connect descriptor uses the SCAN, which is `myscan.mycluster.example.com`.

```
mycluster.example.com =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = host=myscan.mycluster.example.com)
      (PORT = 1522))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = myApp)
    )
  )
```

Oracle Clusterware resolves connection requests that use the net service name `mycluster.example.com` to any of the database instances of the `mycluster` database that run the `myApp` database service. The specific cluster node on which the instance is running is invisible to the client.

6.12.3.2 Net Service Names for Instance Connections

Clients that connect to a particular instance of the database use the net service name for the instance.

Example 6-4 Example Net Service Name Entry for an Instance Connection

In this example, the connect identifier is the same as the instance name, `mycluster1.example.com`. The connect descriptor uses the SCAN to locate the instance within the cluster. Clients connecting to the net service name `mycluster1.example.com` are connected to the `mycluster1` database instance of the `mycluster` database. Oracle Clusterware resolves that connection to the cluster node on which the instance is running. The specific cluster node on which the instance is running is invisible to the client.

```
mycluster1.example.com=
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=TCP)(HOST=myscan.mycluster.example.com)(PORT=1521))
    (CONNECT_DATA=
      (SERVICE_NAME=mycluster.example.com)
      (INSTANCE_NAME=mycluster1)
    )
  )
```

6.12.4 Listener Configuration and the listener.ora File

In Oracle RAC environments, Oracle recommends that you let the Oracle Agent manage Oracle listeners for Oracle Databases.

> **Note:**
>
> If you enable GNS, then you do not have to manually configure the listener.

- Local Listener for an Oracle RAC Database
  The local listener, or default listener, is located in the Grid home when you have Oracle Grid Infrastructure installed.
- Remote Listeners for an Oracle RAC Database
  A **remote listener** is a listener residing on one computer that redirects connections to a database instance on another computer.
- Managing Multiple Listeners for an Oracle RAC Database
  In Oracle RAC environments, Oracle recommends that you let the Oracle Agent manage the Oracle listeners for the databases.
- How Oracle Database Uses the Listener File (listener.ora)
  The `listener.ora` file is the configuration file for a listener.

6.12.4.1 Local Listener for an Oracle RAC Database

The local listener, or default listener, is located in the Grid home when you have Oracle Grid Infrastructure installed.

The `listener.ora` file is located in the *`Grid_home`*`\network\admin` directory. If needed, you can edit the `listener.ora` file for the Grid home listeners to define listener parameters for node and SCAN listeners. Do not modify the endpoints because these are automatically managed by the listener agent.

During Oracle Database creation, the `LOCAL_LISTENER` parameter is automatically configured to point to the local listener for the database. You can set a value manually for `LOCAL_LISTENER`. If you modify the value of the `LOCAL_LISTENER` parameter, then the Database Agent process does not automatically update this value. Oracle recommends that you leave the parameter unset so that the Database Agent process can maintain it automatically. If you do not set `LOCAL_LISTENER`, then the Database Agent process automatically updates the database associated with the local listener in the Grid home, even when the ports or IP address of that listener are changed.

> **See Also:**
>
> - "Net Service Names and the tnsnames.ora File" for more information about listener associations defined in the `tnsnames.ora` file
> - *Oracle Database Net Services Reference* for more information about the `listener.ora` file
> - *Oracle Database Net Services Administrator's Guide* for information about registering information with a local listener

6.12.4.2 Remote Listeners for an Oracle RAC Database

A **remote listener** is a listener residing on one computer that redirects connections to a database instance on another computer.

For example, SCAN listeners are remote listeners. In Oracle RAC environments, Oracle recommends that you let the Oracle Agent manage the Oracle listeners for the databases.

See Also:

- "Net Service Names and the tnsnames.ora File" for more information about listener associations defined in the `tnsnames.ora` file
- *Oracle Database Net Services Reference* for more information about the `listener.ora` file
- *Oracle Database Net Services Administrator's Guide* for information about registering information with a remote listener

6.12.4.3 Managing Multiple Listeners for an Oracle RAC Database

In Oracle RAC environments, Oracle recommends that you let the Oracle Agent manage the Oracle listeners for the databases.

- Use the `lsnrctl` executable located in the Grid home to administer Oracle Database 12*c* release 2 (12.2) local and SCAN listeners.

  Do not attempt to use the `lsnrctl` executables from Oracle home locations for earlier releases, because they cannot be used with Oracle Database 12*c* release 2 (12.2).
- Use SRVCTL and the `setenv` command to modify the value of `TNS_ADMIN` for each listener.

  For listeners not managed by Oracle Clusterware, you can use a nondefault location for the `listener.ora` file by setting the `TNS_ADMIN` environment variable or registry value to point to the directory that contains the Oracle Net Services configuration files.

6.12.4.4 How Oracle Database Uses the Listener File (listener.ora)

The `listener.ora` file is the configuration file for a listener.

The `listener.ora` file can include the protocol addresses it is accepting connection requests on, a list of the database and other services it is listening for, and control parameters used by the listener. You can modify the configuration of the listeners used by Oracle Clusterware and Oracle Real Application Clusters (Oracle RAC) with Server Control Utility (SRVCTL) commands, or by using Oracle Net Configuration Assistant (NETCA). Manual editing of the `listener.ora` file is not required.

Each listener is configured with one or more protocol addresses that specify its listening endpoints. The listener agent dynamically updates endpoints with the listener. Starting with Oracle Database 11*g* Release 2, the `listener.ora` file now only contains an IPC key and the following information:

```
(ADDRESS = (PROTOCOL=TCP)(HOST=)(PORT=1521))
```

In the previous example, the protocol `ADDRESS` refers implicitly to the HOST endpoint of the local node. The listener.ora file is the same on every node for an Oracle RAC

database. Listening endpoints, such as the port numbers, are dynamically registered with the listener.

Before you install Oracle RAC, during the Oracle Grid Infrastructure installation, NETCA creates and starts a default listener in the Grid home called `LISTENER`. The listener is configured with default protocol listening addresses. The listener is configured to respond to connection requests that are directed to one protocol address specified during installation.

During the Oracle RAC installation, the Oracle RAC database uses the listener in the Grid home, and configures service information about the Oracle RAC database. The database services automatically register their information with the listener, such as the service name, instance names, and load information. Dynamic service registration eliminates the need for static configuration of database services. However, static service configuration is required if you plan to use Oracle Enterprise Manager.

Example 6-5 Example listener.ora File for an Oracle RAC Node

The following is an example `listener.ora` file for the `mycluster` cluster as it appears after installation, with an entry for a node named `node1` and a SCAN listener.

```
LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_
SCAN1))))               # line added by Agent
LISTENER_NODE1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER))))
          # line added by Agent
# listener.ora.mycluster Network Configuration File:
C:\app\oracle\product\12.2.0\dbhome_1\network\admin\listener.ora.mycluster
# Generated by Oracle configuration tools.

LISTENER_NODE1 =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
  )

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_NODE1=ON  # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2=ON  # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1=ON  # line added by Agent
```

6.12.5 Net Services Profile File (sqlnet.ora)

Oracle Universal Installer starts Oracle Net Configuration Assistant (NETCA) after the database software is installed. NETCA creates the Oracle Net Services profile, or the `sqlnet.ora` file.

In an Oracle Grid Infrastructure installation, the `sqlnet.ora` file is located in the *`Grid_home`*`\network\admin` directory by default.

For the local listener for the Oracle RAC database instance, the default location of the `sqlnet.ora` file is `%ORACLE_HOME%\network\admin` directory. In this directory there is a default `sqlnet.ora` file. Also, you can find a sample `sqlnet.ora` file in the subdirectory `sample`.

During installation of the Oracle RAC software, NETCA creates the following entries in the `sqlnet.ora` file, where `%ORACLE_BASE%` is the path to the Oracle base directory for the Oracle RAC installation:

```
NAMES.DIRECTORY_PATH=(TNSNAMES, EZCONNECT)
ADR_BASE =%ORACLE_BASE%
```

The `AUTHENTICATION_SERVICES` parameter (not shown in the above example) specifies the method by which users are authenticated for database access. The value `NTS` indicates that Microsoft Windows native authentication should be used to authorize access to the database. Oracle Databases that use Oracle Automatic Storage Management (Oracle ASM) and the databases that are managed by Oracle Grid infrastructure must use Windows native authentication, which is enabled by default.

The parameter `NAMES.DIRECTORY_PATH` specifies the priority order of the naming methods to use to resolve connect identifiers to connect descriptors. The `ADR_BASE` parameter specifies the base directory into which tracing and logging incidents are stored when Automatic Diagnostic Repository (ADR) is enabled for the database.

See Also:

- *Oracle Database Net Services Administrator's Guide* for more information about the `sqlnet.ora` file
- *Oracle Database Concepts* for more information about authentication
- *Oracle Database Client Installation Guide for Microsoft Windows* for more information about configuring clients for database connectivity
- *Oracle Database Administrator's Guide* for more information about ADR

7

Removing Oracle Real Application Clusters Software

If necessary, you can completely remove all Oracle databases, instances, and software from an Oracle Real Application Clusters (Oracle RAC) Oracle home directory.

Topics:

- Overview of Deinstallation Procedures
  There are several tasks to perform when completely removing all Oracle databases, instances, and software from an Oracle home directory.
- Identifying All Instances On a Cluster
  You can identify the database instances on your cluster using either SRVCTL or the Windows Services control interface.
- Removing Oracle RAC Software
  The deinstallation tool removes Oracle Clusterware and Oracle Automatic Storage Management (Oracle ASM) from your server, and Oracle Database installations, for both Oracle RAC and single-instance databases.
- Cleaning Up After a Failed Installation
  If an installation fails, then you must remove the Oracle home directory and remove all files that Oracle Universal Installer (OUI) created during the attempted installation.

> **See Also:**
>
> The product-specific documentation, for example *Oracle Database Client Installation Guide for Microsoft Windows*, for information on removing a single product. Review the requirements and restrictions that apply for that product.

7.1 Overview of Deinstallation Procedures

There are several tasks to perform when completely removing all Oracle databases, instances, and software from an Oracle home directory.

- Identify all instances associated with the Oracle home
- Shut down processes
- Remove listeners installed in the Oracle Database home
- Remove database instances
- Remove Oracle Automatic Storage Management (Oracle ASM) release 11.1 or earlier
- Remove Oracle Clusterware and Oracle ASM (Oracle Grid Infrastructure)

Note:

- For information on removing Oracle Database Vault, see *Oracle Database Vault Administrator's Guide*.
- For information on removing Oracle Configuration Manager, see *Oracle Configuration Manager Installation and Administration Guide*.
- With Oracle Grid Infrastructure 11*g* release 2 (11.2) and later, Oracle ASM and Oracle Clusterware comprise the Oracle Grid Infrastructure installation. These components are installed and removed together.

Caution:

If any cluster member node has more than one database with the same global database name (GDN) on a server, then you cannot use the deinstallation tool to remove only one database.

For example, if you have a standalone database on one of your cluster nodes with the GDN `mydb.example.com`, and your Oracle RAC database GDN is also `mydb.example.com`, then both databases on that node are removed by the deinstallation tool.

7.2 Identifying All Instances On a Cluster

You can identify the database instances on your cluster using either SRVCTL or the Windows Services control interface.

- Identifying All Instances On a Cluster Using SRVCTL
  You can use SRVCTL to identify all database instances associated with an Oracle home.
- Identifying All Instances On a Cluster Using the Windows Services Control Manager
  You can use Windows Services Control Manager to identify all database instances and services associated with an Oracle home.

7.2.1 Identifying All Instances On a Cluster Using SRVCTL

You can use SRVCTL to identify all database instances associated with an Oracle home.

- Enter the following command, where *dbname* is the name of the database:

```
C:\..> srvctl status database -db dbname
```

7.2.2 Identifying All Instances On a Cluster Using the Windows Services Control Manager

You can use Windows Services Control Manager to identify all database instances and services associated with an Oracle home.

1. Log in to a cluster node as a user with Administrator privileges.
2. Use the Windows Services Control Manager to locate the Oracle services related to the Oracle home.

   Look for any Oracle services (their names begin with `Ora`) that access the Oracle home you are removing and have the status `Started`.
   To determine which Oracle home a service is associated with, check "Path to Executable" for a service to see the directory where the executable for the service is located.

7.3 Removing Oracle RAC Software

The deinstallation tool removes Oracle Clusterware and Oracle Automatic Storage Management (Oracle ASM) from your server, and Oracle Database installations, for both Oracle RAC and single-instance databases.

- About the Deinstallation Tool
  The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system.
- Deinstallation Tool Command Reference
  You can use the deinstallation tool to remove Oracle software. You can run this command in standalone mode, from an Oracle home directory, or through the installer.
- Using the Deinstallation Tool to Remove Oracle RAC
  You can run the deinstallation tool in multiple ways.

7.3.1 About the Deinstallation Tool

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system.

Starting with Oracle Database 12*c*, the deinstallation tool is integrated with the database installation media. You can run the deinstallation tool using the `setup.exe` command with the `-deinstall` and `-home` options from the base directory of the Oracle Database or Oracle Database Client installation media.

The deinstallation tool is also available as a separate command (`deinstall.bat`) in Oracle home directories after installation. It is located in the `%ORACLE_HOME%\deinstall` directory.

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system for a specific Oracle home. If you run the deinstallation tool to remove an Oracle Grid Infrastructure for Windows installation, then the deinstaller automatically runs the appropriate scripts to deconfigure Oracle Grid Infrastructure or Oracle Grid Infrastructure for standalone server.

The deinstallation tool uses the information you provide, plus information gathered from the software home to create a response file. You can alternatively supply a response file generated previously by the `deinstall.bat` command using the `-checkonly` option and `-o` option. You can also edit a response file template to create a response file.

> **Note:**
>
> You must run the deinstallation tool from the same release to remove Oracle software. Do not run the deinstallation tool from a later release to remove Oracle software from an earlier release. For example, do not run the deinstallation tool from the Oracle Database 12.2 installation media to remove Oracle software from an existing 11.2.0.4 Oracle home.

If the software in the Oracle home is not running (for example, after an unsuccessful installation), then the deinstallation tool cannot determine the configuration and you must provide all the configuration details either interactively or in a response file.

7.3.2 Deinstallation Tool Command Reference

You can use the deinstallation tool to remove Oracle software. You can run this command in standalone mode, from an Oracle home directory, or through the installer.

Purpose

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system.

File Path

```
%ORACLE_HOME%\deinstall\deinstall
```

Prerequisites

Before you run the deinstallation tool for Oracle Grid Infrastructure installations:

- Dismount Oracle Automatic Storage Management Cluster File System (Oracle ACFS) and disable Oracle Automatic Storage Management Dynamic Volume Manager (Oracle ADVM).
- If Grid Naming Service (GNS) is in use, then notify your DNS administrator to delete the subdomain entry from the DNS.

Syntax When Using the deinstall.bat Program

```
deinstall.bat [-silent] [-checkonly] [-local]
 [-paramfile complete path of input response file]
 [-params name1=value [name2=value . . .]]
 [-o complete path of directory for saving files]
 [-tmpdir complete path of temporary directory to use]
 [-logdir complete path of log directory to use] [-skipLocalHomeDeletion] [-
skipRemoteHomeDeletion] [-help]
```

Options

Table 7-1 Options for the Deinstallation Tool

Command Option	Description
`home` *`complete path of Oracle home`*	Specify this option to indicate the home path of the Oracle home to check or deinstall. To deinstall Oracle software using the `deinstall.bat` command located within the Oracle home being removed, provide a response file in a location outside the Oracle home, and do not use the `-home` option. If you run `deinstall.bat` from the `%ORACLE_HOME%\deinstall` path, then the `-home` option is not required because the tool knows from which home it is being run.
`silent`	Specify this option to run the deinstallation tool in noninteractive mode. This option requires one of the following: • A working system that it can access to determine the installation and configuration information; the `-silent` option does not work with failed installations. • A response file that contains the configuration values for the Oracle home that is being deinstalled or deconfigured.
`checkonly`	Specify this option to check the status of the Oracle software home configuration. Running the `deinstall` command with the `-checkonly` option does not remove the Oracle configuration. This option generates a response file that you can use with the `deinstall.bat` command. When you use the `-checkonly` option to generate a response file, you are prompted to provide information about your system. You can accept the default value the tool has obtained from your Oracle installation, indicated inside brackets (`[]`), or you can provide different values. To accept the defaults, press `Enter` at each prompt.
`local`	Specify this option on a multinode environment to deconfigure Oracle software in a cluster. When you run `deinstall.bat` with this option, it deconfigures and deinstalls the Oracle software only on the local node (the node on which you run `deinstall.bat`) for non-shared Oracle home directories. The deinstallation tool does not deinstall or deconfigure Oracle software on remote nodes.

Table 7-1 (Cont.) Options for the Deinstallation Tool

Command Option	Description
`paramfile` *`complete path of input response file`*	(Optional) You can specify this option to run `deinstall.bat` with a response file in a location other than the default. When you use this option, provide the complete path where the response file is located. If you run the `deinstall.bat` command from the Oracle home that you plan to deinstall, then you do not need to specify the `-paramfile` option. The default location of the response file depends on the location of the deinstallation tool: • From the installation media or stage location: *`X:\staging_location`*`\deinstall\response` • After installation, from the installed Oracle home: `%ORACLE_HOME%\deinstall\response`.
`params` *`name1=value`*`[` *`name2=value name3=value...`*`]`	Use this option with a response file to override one or more values in a response file that you created.
`o` *`complete path of directory for saving response files`*	Use this option to provide a path other than the default location where the response file (`deinstall.rsp.tmpl`) is saved. The default location of the response file depends on the invocation method of the deinstallation tool: • From the installation media or stage location: *`stagelocation`*`\response` • After installation, from the installed Oracle home: `%ORACLE_HOME%\deinstall\response`.
`tmpdir` *`complete path of temporary directory to use`*	Specifies a non-default location where the deinstallation tool writes the temporary files for the deinstallation.
`logdir` *`complete path of log directory to use`*	Specifies a non-default location where the deinstallation tool writes the log files for the deinstallation.
`skipLocalHomeDeletion`	Specify this option in Oracle Grid Infrastructure installations on a multinode environment to deconfigure a local Grid home without deleting the Grid home.
`skipRemoteHomeDeletion`	Specify this option in Oracle Grid Infrastructure installations on a multinode environment to deconfigure a remote Grid home without deleting the Grid home.
`help`	Use the `-help` option to obtain additional information about the deinstallation tool options.

Location of Log Files for the Deinstallation Tool

If you use the `deinstall.bat` command located in an Oracle home, then the deinstallation tool writes log files in the `C:\Program Files\Oracle\Inventory\logs` directory.

If you are using the `deinstall.bat` command to remove the last Oracle home installed on the server, then the log files are written to the current user's home directory. For example, if you are logged in as the domain user `RACDBA\dba1`, then the log files are stored in the directory `C:\Users\dba1.RACDBA\logs`.

7.3.3 Using the Deinstallation Tool to Remove Oracle RAC

You can run the deinstallation tool in multiple ways.

- Running the Deinstallation Tool From an Oracle Home
  You can run the deinstallation tool from an Oracle home or from the software installation media.
- Running the Deinstallation Tool Interactively From the Installer
  You can run the deinstallation tool from an Oracle home or from the software installation media.
- Running the Deinstallation Tool with a Response File
  When running the deinstallation tool, you can use a response file instead of responding to each prompt individually.
- Generating a Response File For Use With the Deinstallation Tool
  To use a response file with the deinstallation tool you must first create the response file.

7.3.3.1 Running the Deinstallation Tool From an Oracle Home

You can run the deinstallation tool from an Oracle home or from the software installation media.

1. The default method for running the deinstallation tool is from the `deinstall` directory in the Oracle home as the Oracle Installation user:

   ```
   C:\> %ORACLE_HOME%\deinstall\deinstall.bat
   ```

2. Provide information about your servers as prompted or accept the defaults.

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system.

Example 7-1 Running `deinstall.bat` From Within the Oracle Home

The most common method of running the deinstallation tool is to use the version installed in the Oracle home being removed. The deinstallation tool determines the software configuration for the local Oracle home, and then provides default values at each prompt. You can either accept the default value, or override it with a different value. If the software in the Oracle home is not running (for example, after an unsuccessful installation), then the deinstallation tool cannot determine the configuration, and you must provide all the configuration details either interactively or in a response file. To use the deinstallation tool located in the current Oracle home directory, issue the following commands while logged in as a member of the Administrators group:

```
C:\> app\oracle\product\12.2.0\dbhome_1\deinstall\deinstall.bat
```

Provide additional information as prompted.

> **Note:**
>
> When using the deinstallation tool from a location other than within the Oracle home being removed, you must specify the `-home` option on the command line.

7.3.3.2 Running the Deinstallation Tool Interactively From the Installer

You can run the deinstallation tool from an Oracle home or from the software installation media.

1. Use the `setup.exe` command with the `-deinstall` option, followed by the `-home` option to specify the path of the Oracle home you want to remove.

   For example:

   ```
   setup.exe -deinstall -home C:\app\12.2.0\grid
   ```

2. Provide information about your servers as prompted or accept the defaults.

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system.

Example 7-2 Running the Deinstallation Tool from the Software Installation Media

If you run the deinstallation tool from the installer in the installation media, then when the `deinstall.bat` command runs, it uses the information you provide to determine the system configuration and then provides default values at each prompt. You can either accept the default value, or override it with a different value. If the software in the specified Oracle home is not running (for example, after an unsuccessful install attempt), then the deinstallation tool cannot determine the configuration, and you must provide all the configuration details either interactively or in a response file.

For example, to remove the Oracle Database home `dbhome_1` using the deinstallation tool started from the installer, use commands similar to the following while logged in as a member of the Administrator group, where *directory_path* is the path to the database directory on the installation media:

```
C:\> cd directory_path
C:\..database> setup.exe -deinstall -home C:\app\oracle\product\12.2.0\db_home1\
```

Provide additional information as prompted.

> **Note:**
>
> When using the deinstallation tool from a location other than within the Oracle home being removed, you must specify the `-home` option on the command line.

7.3.3.3 Running the Deinstallation Tool with a Response File

When running the deinstallation tool, you can use a response file instead of responding to each prompt individually.

The deinstallation tool uses the information you provide, plus information gathered from the software home to create a response file. You can alternatively supply a response file generated previously by the `deinstall.bat` command using the `-checkonly` option and `-o` option. You can also edit a response file template located at *`Oracle_Home`*`\deinstall\response\deinstall.rsp.tmpl` to create a response file.

- To run the `setup.exe` command located in the path */directory_path*, where *directory_path* is the path to the database directory on the installation media, and use a response file in the software owner location `C:\user\oracle`, enter the following commands while logged in as a member of the Administrators group:

  ```
  C:\> cd directory_path\
  C:\..database> setup.exe -deinstall -paramfile C:\user\oracle\my_db_parmfile.tmpl
  ```

7.3.3.4 Generating a Response File For Use With the Deinstallation Tool

To use a response file with the deinstallation tool you must first create the response file.

You can generate the a response file by running the `deinstall.bat` command with the `-checkonly` and `-o` options before you run the command to deinstall the Oracle home, or you can use the response file template and manually edit it to create the response file.

Alternatively, you can use the response file template located at `%ORACLE_HOME%\deinstall\response\deinstall.rsp.tmpl`.

- To generate the response file `deinstall_dbhome_1.rsp` using the `deinstall.bat` command located in the Oracle home and the `-checkonly` option, enter a command similar to the following, where `C:\app\oracle\product\12.2.0\dbhome_1` is the location of the Oracle home and `C:\Users\oracle` is the directory in which the generated response file is created:

  ```
  C:\> app\oracle\product\12.2.0\dbhome_1\deinstall\deinstall.bat -checkonly -o C:\Users\oracle\
  ```

7.4 Cleaning Up After a Failed Installation

If an installation fails, then you must remove the Oracle home directory and remove all files that Oracle Universal Installer (OUI) created during the attempted installation.

1. Run OUI to deinstall Oracle RAC.
2. Manually remove the directory that was used as the Oracle home directory during the installation.
3. Remove the following Windows Registry keys created by OUI during the previous installation attempt:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBCINST.INI
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Oracle12\Performance
 \KEY_ORACLE_HOME_NAME
```

After you have completed these steps, you can start the installation again.

> **See Also:**
>
> Removing Oracle Real Application Clusters Software for more information about removing Oracle RAC software.

A

Using Scripts or Response Files to Create Oracle RAC Databases

During noninteractive installations you can create Oracle Real Application Clusters (Oracle RAC) databases using scripts.

> **Note:**
>
> The scripts generated by Database Configuration Assistant (DBCA) are for reference only. Oracle strongly recommends that you use DBCA to create a database.

- Using DBCA to Generate Installation Scripts for Oracle RAC
  You can generate scripts that create an Oracle RAC database and prepare the database for use.
- About DBCA Noninteractive (Silent) Configuration for Oracle RAC
  You can perform a noninteractive, or *silent* configuration of Oracle RAC using Oracle Database Configuration Assistant (DBCA).
- Using DBCA Commands for Noninteractive (Silent) Configuration of Oracle RAC
  You can use Oracle Database Configuration Assistant (DBCA) in non-interactive mode to create an Oracle RAC database.
- How Response Files Work
  Response files can assist you with installing an Oracle product multiple times on multiple computers.
- Preparing Response Files
  There are two methods you can use to prepare response files for silent mode or response file mode installations.
- Running Oracle Universal Installer Using a Response File
  After creating the response file, run Oracle Universal Installer at the command line, specifying the response file you created, to perform the installation.
- Running Configuration Assistants Using Response Files
  You can run configuration assistants in response file mode to configure and start Oracle software after it is installed on your system.
- Postinstallation Configuration Using Response File Created During Installation
  Use response files to configure Oracle software after installation. You can use the same response file created during installation to also complete postinstallation configuration.
- Postinstallation Configuration Using the ConfigToolAllCommands Script
  You can create and run a response file configuration after installing Oracle software.

> **See Also:**
>
> *Oracle Universal Installer (OUI) User's Guide* for detailed information about generating scripts

A.1 Using DBCA to Generate Installation Scripts for Oracle RAC

You can generate scripts that create an Oracle RAC database and prepare the database for use.

1. Start Oracle Database Configuration Assistant (DBCA) and select your preferred options to build the Oracle RAC database.
   a. On the Creation Options page of your DBCA session, deselect Create Database.
   b. Select **Generate Database Creation Scripts**.
   c. Click **Finish**.

   You can accept the default destination directory for the scripts, or browse for a different location. In either case, note the path name for use in the next step.
2. Go to the directory 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 the scripts do not contain the statements for the specific database characteristics you need, 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.bat`, 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 entering an `ALTER SYSTEM` statement in SQL*Plus, or by uncommenting the parameter in the PFILE for each instance.
5. Configure Oracle Net Services to support your new database and instances.
6. Set the `REMOTE_LISTENER` parameter to the SCAN (using the Easy Connect Naming syntax *scanname*:*scanport*) in your SPFILE by entering an `ALTER SYSTEM` statement in SQL*Plus, or by uncommenting the parameter in the PFILE for each instance.
7. Run the Server Control Utility (SRVCTL) to configure and start database and instance applications as described in *Oracle Real Application Clusters Administration and Deployment Guide*.

See Also:

- "Understanding the Oracle RAC Installed Configuration"
- "Selecting DBCA Options to Create an Oracle RAC or Oracle RAC One Node Database" for additional information about running a DBCA session.
- *Oracle Universal Installer (OUI) User's Guide* for additional information about creating and using scripts to install Oracle software with Oracle Universal Installer (OUI).

A.2 About DBCA Noninteractive (Silent) Configuration for Oracle RAC

You can perform a noninteractive, or *silent* configuration of Oracle RAC using Oracle Database Configuration Assistant (DBCA).

To perform a silent configuration, you must have completed an Oracle Grid Infrastructure (Oracle Clusterware and Oracle Automatic Storage Management (Oracle ASM)) installation.

You can use DBCA to create a database from templates supplied by Oracle, or from templates that you create. The templates contain settings optimized for a particular type of workload.

Oracle provides templates for the following two workload types:

- General purpose or transaction processing
- Data warehouse

For more complex environments, you can select the **Custom Database** option. This option does not use templates and results in a more extensive installation interview, which means that it takes longer to create your database.

A.3 Using DBCA Commands for Noninteractive (Silent) Configuration of Oracle RAC

You can use Oracle Database Configuration Assistant (DBCA) in non-interactive mode to create an Oracle RAC database.

Example A-1 Using DBCA in Silent Mode to Create an Oracle RAC Database

You can use the following command syntax to create an Oracle RAC database using the general purpose template, placing the data files in an existing Oracle ASM disk group. Nodes `node1` and `node2` are the cluster nodes on which Oracle RAC database instances are created. The disk group name is `+ASMgrp1`. An Oracle Home User was specified for this installation, and is indicated with the `-serviceUserPassword` option:

```
%ORACLE_HOME%\bin\dbca -silent -createDatabase -templateName General_Purpose.dbc
-gdbName %DBNAME% -sid %ORACLE_SID% -sysPassword -systemPassword
-sysmanPassword -dbsnmpPassword  -serviceUserPassword
-emConfiguration LOCAL -storageType ASM -diskGroupName +ASMgrp1
```

```
-datafileJarLocation %ORACLE_HOME%\assistants\dbca\templates
-nodeinfo node1,node2 -characterset WE8MSWIN1252
-obfuscatedPasswords false -sampleSchema false -asmSysPassword
```

After you run this command, if you did not include the passwords as values in the above command, then DBCA prompts you for the passwords for the SYS, SYSTEM, SYSMAN, DBSNMP, Oracle Home (or Oracle Service) and SYSASM users, for example:

```
Enter SYS user password:
password
Enter SYSTEM user password:
password
...
```

> **See Also:**
>
> *Oracle Database Administrator's Guide* for a complete description of the `dbca` commands and options

A.4 How Response Files Work

Response files can assist you with installing an Oracle product multiple times on multiple computers.

When you start the installer, you can use a response file to automate the installation and configuration of Oracle software, either fully or partially. The installer uses the values contained in the response file to provide answers to some or all installation prompts.

Typically, the installer runs in interactive mode, which means that it prompts you to provide information in graphical user interface (GUI) screens. When you use response files to provide this information, you run the installer from a command prompt using either of the following modes:

- Silent mode

  If you include responses for all of the prompts in the response file and specify the `-silent` option when starting the installer, then it runs in silent mode. During a silent mode installation, the installer does not display any screens. Instead, it displays progress information in the terminal that you used to start it.

- Response file mode

  If you include responses for some or all of the prompts in the response file and omit the `-silent` option, then the installer runs in response file mode. During a response file mode installation, the installer displays all the screens, screens for which you specify information in the response file, and also screens for which you did not specify the required information in the response file.

You define the settings for a silent or response file installation by entering values for the variables listed in the response file. For example, to specify the Oracle home name, supply the appropriate value for the `ORACLE_HOME` variable:

```
ORACLE_HOME=C:\app\oracle\product\12.2.0\dbhome_1
```

Another way of specifying the response file variable settings is to pass them as command line arguments when you run the installer. For example:

```
-silent directory_path
```

In this command, *directory_path* is the path of the database directory on the DVD, or the path of the directory on the hard drive.

Ensure that you enclose the variable and its setting in double-quotes.

- Deciding to Use Silent Mode or Response File Mode
  There are several reasons for running the installer in silent mode or response file mode.
- Creating a Database Using Oracle ASM for Database Files in Silent Mode
  Creating an Oracle Real Application Clusters (Oracle RAC) database that uses Oracle Automatic Storage Management (Oracle ASM) for storage is a multi-step process.
- Using Response Files
  Use these general steps for installing and configuring Oracle products using the installer in silent or response file mode.

> **See Also:**
>
> *Oracle Universal Installer (OUI) User's Guide* for more information about response files

A.4.1 Deciding to Use Silent Mode or Response File Mode

There are several reasons for running the installer in silent mode or response file mode.

Table A-1 Reasons for Using Silent Mode or Response File Mode

Mode	Reasons to Use
Silent	Use silent mode for the following installations: • To complete an unattended installation, which you schedule using operating system utilities • To complete several similar installations on multiple systems without user interaction • Install the software on a system that cannot display the Oracle Universal Installer (OUI) graphical user interface OUI displays progress information on the terminal that you used to start it, but it does not display any of the installer screens.

Table A-1 (Cont.) Reasons for Using Silent Mode or Response File Mode

Mode	Reasons to Use
Response file	Use response file mode to complete similar Oracle software installations on more than one system, providing default answers to some, but not all the installer prompts. If you do not specify information required for a particular OUI screen in the response file, then the installer displays that screen. OUI suppresses screens for which you have provided all of the required information.

A.4.2 Creating a Database Using Oracle ASM for Database Files in Silent Mode

Creating an Oracle Real Application Clusters (Oracle RAC) database that uses Oracle Automatic Storage Management (Oracle ASM) for storage is a multi-step process.

Before you create a database that uses Oracle ASM, you must run the `root.bat` script. For this reason, you cannot create a database using Oracle ASM as the storage option for database files during a silent-mode installation.

> **Note:**
>
> This limitation applies only to databases that use Oracle Automatic Storage Management as the storage option for database files. You can create a database that uses the file system option during a silent-mode installation.

1. Complete a software-only installation of Oracle RAC using silent-mode.
2. Optional: Run Oracle Net Configuration Assistant in silent mode.
3. Run Oracle Database Configuration Assistant (DBCA) in silent mode.

> **See Also:**
>
> "Perform a Software-Only Installation of Oracle Database"

A.4.3 Using Response Files

Use these general steps for installing and configuring Oracle products using the installer in silent or response file mode.

> **Note:**
>
> You must complete all required preinstallation tasks on a system before running the installer in silent or response file mode.

1. Verify the Windows Registry key `HKEY_LOCAL_MACHINE\Software\Oracle` exists and that the value for `inst_loc` is the location of the Oracle Inventory directory on the local node.

   To install Oracle RAC, you must first have installed Oracle Grid Infrastructure on your cluster nodes. The Oracle Inventory directory was created and added to the Windows registry during the installation of Oracle Grid Infrastructure. If the `inst_loc` key does not exist in the Windows registry, then install Oracle Grid Infrastructure for a cluster before continuing.

   > **Note:**
   >
   > Changing the value for `inst_loc` in the Windows registry is not supported after the installation of Oracle software

2. Prepare a response file.
3. Run the installer in silent or response file mode.
4. If you completed a software-only installation, then run Oracle Net Configuration Assistant (NETCA) and Oracle Database Configuration Assistant (DBCA) in silent or response file mode.

A.5 Preparing Response Files

There are two methods you can use to prepare response files for silent mode or response file mode installations.

- About Response File Templates
  Oracle provides response file templates for each product and installation type and for each configuration tool.
- Editing a Response File Template
  You can copy and modify a response file template for each product and installation type and for each configuration tool.
- Recording Response Files
  You can use the installer in interactive mode to record response files, which you can then edit and use to complete silent mode or response file mode installations.

A.5.1 About Response File Templates

Oracle provides response file templates for each product and installation type and for each configuration tool.

For Oracle Database, the response file templates are located in the `database\response` directory on the installation media and in the *`Oracle_home`*`\inventory\response` directory. For Oracle Grid Infrastructure, the response file templates are located in the *`Grid_home`*`\install\response` directory after the software is installed.

> **Note:**
>
> If you copied the installation media to a directory on a local disk (referred to as the *staging_dir* directory), then the response files are located in the directory *`staging_dir`*`\database\response`.

All response file templates contain comment entries, sample formats, examples, and other useful instructions. Read the response file instructions to understand how to specify values for the response file variables, so that you can customize your installation.

The following response files are provided with this software:

Table A-2 Response Files for Oracle Database and Oracle Grid Infrastructure

Response File	Used For
`db_install.rsp`	Silent installation of Oracle Database
`dbca.rsp`	Silent creation and configuration of an Oracle Database using Oracle Database Configuration Assistant (DBCA)
`netca.rsp`	Silent configuration of Oracle Net using NETCA
`grid_install.rsp`	Silent configuration of Oracle Grid Infrastructure installations

> **Caution:**
>
> When you modify a response file template and save a file for use, the response file may contain plain text passwords. Ownership of the response file must be given to the Oracle software installation owner only, and access restricted to the response file. Oracle strongly recommends that database administrators or other administrators delete or secure response files when they are not in use.

A.5.2 Editing a Response File Template

You can copy and modify a response file template for each product and installation type and for each configuration tool.

To copy and modify a response file, perform the following steps:

1. Copy the response file from the response file directory to a directory on your system.

   ```
   copy Oracle_home\install\response\product_timestamp.rsp local_directory
   ```

2. Open the response file in a text editor.
3. Follow the instructions in the file to edit it.

> **Note:**
>
> The installer or configuration assistant fails if you do not correctly configure the response file. Also, ensure that your response file name has the `.rsp` suffix.

4. Secure the response file.

   Ensure that only the user that installed the Oracle software can view or modify response files. Consider deleting the modified response file after the installation succeeds.

> **Note:**
>
> A fully specified response file for an Oracle Grid Infrastructure installation or an Oracle Database installation can contain the passwords for:
>
> - Oracle Automatic Storage Management (Oracle ASM) administrative accounts
> - Database administrative accounts
> - A user who is a member of the operating system group `ORA_DBA` (required for automated backups)

> **See Also:**
>
> *Oracle Universal Installer (OUI) User's Guide* for detailed information about creating response files

A.5.3 Recording Response Files

You can use the installer in interactive mode to record response files, which you can then edit and use to complete silent mode or response file mode installations.

This method is useful for Advanced or software-only installations. You can save all the installation steps into a response file during installation by clicking **Save Response**

File on the Summary page. You can use the generated response file for a silent installation later.

When you record the response file, you can either complete the installation, or you can exit from the installer on the Summary page, before the installer starts to copy the software to the local disk.

If you use record mode during a response file mode installation, then the installer records the variable values that were specified in the original source response file into the new response file.

> **Note:**
>
> You cannot save passwords while recording the response file.

1. Complete preinstallation tasks as for a standard installation.

   When you use run the installer to record a response file, it checks the system to verify that it meets the requirements to install the software. For this reason, Oracle recommends that you complete all of the required preinstallation tasks and record the response file while completing an installation.
2. Log in as the Oracle Installation User. Ensure that the Oracle Installation User has permissions to create or write to the Oracle home path that you specify during installation.
3. Start the installer. On each installation screen, specify the required information.
4. When the installer displays the Summary screen, perform the following steps:
   a. Click **Save Response File**. In the pop up window, specify a file name and location to save the values for the response file, then click **Save**.
   b. Click **Finish** to continue with the installation.

      Click **Cancel** if you do not want to continue with the installation. The installation stops, but the recorded response file is retained.

> **Note:**
>
> Your response file name must end with the `.rsp` suffix.

5. If you do not complete the installation, then delete the Oracle home directory that the installer created using the path you specified in the Specify File Locations screen.
6. Before you use the saved response file on another system, edit the file and make any required changes. Use the instructions in the file as a guide when editing it.

A.6 Running Oracle Universal Installer Using a Response File

After creating the response file, run Oracle Universal Installer at the command line, specifying the response file you created, to perform the installation.

Run Oracle Universal Installer at the command line, specifying the response file you created. The Oracle Universal Installer executables, `setup.exe` and `gridSetup.bat`, provide several options. For help information on the full set of these options, run the `gridSetup.bat` or `setup.exe` command with the `-help` option. For example:

- For Oracle Database:

  ```
  C:\..\bin> setup.exe -help
  ```

- For Oracle Grid Infrastructure:

  ```
  C:\..\bin> gridSetup.bat -help
  ```

The help information appears in your session window after a short period of time.

To run the installer using a response file, perform the following steps:

1. Complete the preinstallation tasks for a normal installation.
2. Log in as an Administrator user or the user that installed the software.
3. To start the installer in silent or response file mode, enter a command similar to the following:

   - For Oracle Database:

     ```
     C:\> directory_path\setup.exe [-silent] [-noconfig] \
          -responseFile response_filename
     ```

   - For Oracle Grid Infrastructure:

     ```
     C:\> directory_path\gridSetup.bat [-silent] [-noconfig] \
          -responseFile response_filename
     ```

> **Note:**
>
> Do not specify a relative path to the response file. If you specify a relative path, then the installer fails.

In this example:

- *`directory_path`* is the path of the DVD or the path of the directory on the hard drive where you have copied the installation software.
- `-silent` runs the installer in silent mode.
- `-noconfig` suppresses running the configuration assistants during installation, and a software-only installation is performed instead.
- *`response_filename`* is the full path and file name of the installation response file that you configured.

A.7 Running Configuration Assistants Using Response Files

You can run configuration assistants in response file mode to configure and start Oracle software after it is installed on your system.

To run configuration assistants in response file mode, you must first copy and edit a response file template.

> **Note:**
>
> If you copied the software to a hard disk, then the response file template is located in the directory `X:\staging_dir\database\response`.

- Silent Mode of Database Configuration Assistant
  Use the `-silent` option in combination with the `-responseFile` option to set the mode to silent.
- Running Database Configuration Assistant in Response File Mode
  You can run Database Configuration Assistant (DBCA) in response file mode to configure and start an Oracle database on the system.
- Running Oracle Net Configuration Assistant Using Response Files
  You can run Oracle Net Configuration Assistant (NETCA) in silent mode to configure and start an Oracle Net listener on the system, configure naming methods, and configure Oracle Net service names.

A.7.1 Silent Mode of Database Configuration Assistant

Use the `-silent` option in combination with the `-responseFile` option to set the mode to silent.

In the silent mode, Database Configuration Assistant uses values that you specify, in the response file or as command-line options, to create a database. No window or user interface is displayed in the silent mode.

A.7.2 Running Database Configuration Assistant in Response File Mode

You can run Database Configuration Assistant (DBCA) in response file mode to configure and start an Oracle database on the system.

To run DBCA in response file mode, you must copy and edit a response file template.

1. Oracle provides a response file template named `dbca.rsp` in the response directory on the installation media. Copy the `dbca.rsp` response file template from the response file directory to a directory on your system.

   If you have copied the software to a hard drive, then you can edit the file in the `X:\staging_dir\database\response` directory if you prefer.

> **Note:**
>
> As an alternative to editing the response file template, you can also create a database by specifying all required information as command-line options when you run DBCA. For information about the list of options supported, enter the following command:
>
> ```
> C:\> %ORACLE_HOME%\bin\dbca -help
> ```

2. Log in as the Oracle Home user. Set the `%ORACLE_HOME%` environment variable to the correct Oracle home directory.
3. Open the response file in a text editor.
4. Edit the file, following the instructions in the file.

   In response file mode, DBCA uses values that you specify in the response file or as command-line options, to create a database.

> **Note:**
>
> DBCA fails if you do not correctly configure the response file.

5. Open a command-line window. Change directories to the location of the Oracle home directory.
6. Use a command similar to the following example to run DBCA in silent or response file mode using a response file:

   ```
   C:\> %ORACLE_HOME%\bin\dbca {-silent} -responseFile \
   local_dir\dbca.rsp
   ```

   In this example:

   - The `-silent` option runs DBCA in silent mode, suppressing user prompts.
   - *local_dir* is the full path of the directory where the `dbca.rsp` response file is located.

As DBCA configures and starts the database, it displays a window that contains status messages and a progress bar. The window that DBCA displays is the same window that is displayed when you choose to create a preconfigured database during an Oracle Database Enterprise Edition, Standard Edition, or Standard Edition 2 (SE2) installation.

> **See Also:**
>
> - "Multiple Oracle Home Directories on Windows" for more information about changing the current setting for Oracle Home.
> - *Oracle Database Administrator's Guide* for information about using DBCA in noninteractive, or silent mode, to create a database
> - *Oracle Automatic Storage Management Administrator's Guide* for information about running Oracle Automatic Storage Management Configuration Assistant (ASMCA) in noninteractive mode

A.7.3 Running Oracle Net Configuration Assistant Using Response Files

You can run Oracle Net Configuration Assistant (NETCA) in silent mode to configure and start an Oracle Net listener on the system, configure naming methods, and configure Oracle Net service names.

Oracle provides a response file template named `netca.rsp` in the `response` subdirectory of:

- The *`Oracle_home`*`\database\inventory\response` directory after a software-only installation
- The `database\response` directory on the installation media or staging area

To run NETCA in silent mode, you must copy and edit a response file template.

1. Copy the `netca.rsp` response file template from the response file directory to a directory on your system.

   If the software is staged on a hard drive, or has already been installed, then you can edit the file in the `response` directory located on the local disk instead.
2. Open the response file in a text editor.
3. Follow the instructions in the file to edit it.

> **Note:**
>
> NETCA fails if you do not correctly configure the response file.

4. Log in as the Oracle Home user. Set the `%ORACLE_HOME%` environment variable to the correct Oracle home directory.
5. Enter a command similar to the following to run NETCA in silent mode:

   ```
   C:\> Oracle_home\bin\netca -silent -responsefile X:\local_dir\netca.rsp
   ```

   In this command:

   - The `-silent` option runs NETCA in silent mode.

- *X:\local_dir* is the full path of the directory where you copied the `netca.rsp` response file template, where *X* represents the drive on which the file is located, and *local_dir* the path on that drive.

Related Topics:

- Multiple Oracle Home Directories on Windows
  Install each Oracle product in its own Oracle home.
- Changing the Current Setting for Oracle Home
  Use Oracle Universal Installer (OUI) to change the current Oracle home.

A.8 Postinstallation Configuration Using Response File Created During Installation

Use response files to configure Oracle software after installation. You can use the same response file created during installation to also complete postinstallation configuration.

- Using the Installation Response File for Postinstallation Configuration
  Starting with Oracle Database 12*c* release 2 (12.2), you can use the response file created during installation to also complete postinstallation configuration.
- Running Postinstallation Configuration Using Response File
  Complete this procedure to run configuration assistants configuration with the `-executeConfigTools` command.

A.8.1 Using the Installation Response File for Postinstallation Configuration

Starting with Oracle Database 12*c* release 2 (12.2), you can use the response file created during installation to also complete postinstallation configuration.

Run the installer with the `-executeConfigTools` option to configure configuration assistants after installing Oracle Grid Infrastructure or Oracle Database. You can use the response file located at *Oracle_home*`\install\response\`*product_timestamp*`.rsp` to obtain the passwords required to run the configuration tools. You must update the response file with the required passwords before running the `-executeConfigTools` command.

Oracle strongly recommends that you maintain security with a password response file. The owner of the response file must be the installation owner user.

Example A-2 Response File Passwords for Oracle Grid Infrastructure

```
oracle.install.crs.config.ipmi.bmcPassword=password
oracle.install.asm.SYSASMPassword=password
oracle.install.asm.monitorPassword=password
oracle.install.config.emAdminPassword=password
oracle.install.OracleHomeUserPassword=password
```

If you do not have a BMC card, or you do not want to enable IPMI, then leave the `ipmi.bmcPassword` input field blank.

If you do not want to enable Oracle Enterprise Manager for management, then leave the `emAdminPassword` password field blank.

If you did not specify an Oracle Home user for the Oracle Grid Infrastructure installation, then leave the `OracleHomeUserPassword` field blank.

Example A-3 Response File Passwords for Oracle Grid Infrastructure for a Standalone Server (Oracle Restart)

```
oracle.install.asm.SYSASMPassword=password
oracle.install.asm.monitorPassword=password
oracle.install.config.emAdminPassword=password
oracle.install.OracleHomeUserPassword=password
```

If you do not want to enable Oracle Enterprise Manager for management, then leave the `emAdminPassword` password field blank.

If you did not specify an Oracle Home user for the Oracle Grid Infrastructure for a Standalone Server (Oracle Restart) installation, then leave the `OracleHomeUserPassword` field blank.

Example A-4 Response File Passwords for Oracle Database

This example illustrates the passwords to specify for use with the database configuration assistants.

```
oracle.install.db.config.starterdb.password.SYS=password
oracle.install.db.config.starterdb.password.SYSTEM=password
oracle.install.db.config.starterdb.password.DBSNMP=password
oracle.install.db.config.starterdb.password.PDBADMIN=password
oracle.install.db.config.starterdb.emAdminPassword=password
oracle.install.db.config.asm.ASMSNMPPassword=password
oracle.install.OracleHomeUserPassword=password
```

You can also specify `oracle.install.db.config.starterdb.password.ALL=password` to use the same password for all database users.

Oracle Database configuration assistants require the SYS, SYSTEM, and DBSNMP passwords for use with Oracle Database Configuration Assistant (DBCA). Specify the following passwords, depending on your system configuration:

- If the database uses Oracle ASM for storage, then you must specify a password for the `ASMSNMPPassword` variable. If you are not using Oracle ASM, then leave the value for this password variable blank.
- If you create a multitenant container database (CDB) with one or more pluggable databases (PDBs), then you must specify a password for the `PDBADMIN` variable. If you are not using Oracle ASM, then leave the value for this password variable blank.
- If you did not specify an Oracle Home user for the Oracle Database installation, then leave the `OracleHomeUserPassword` field blank.

A.8.2 Running Postinstallation Configuration Using Response File

Complete this procedure to run configuration assistants configuration with the `-executeConfigTools` command.

1. Edit the response file and specify the required passwords for your configuration. You can use the response file created during installation, located at *`Oracle_home`*`\install\response\`*`product_timestamp`*`.rsp`. For example, for Oracle Grid Infrastructure:

```
oracle.install.asm.SYSASMPassword=password
oracle.install.config.emAdminPassword=password
```

2. Change directory to the Oracle home containing the installation software. For example, for Oracle Grid Infrastructure:

   ```
   cd Grid_home
   ```

3. Run the configuration script using the following syntax:

   For Oracle Grid Infrastructure:

   ```
   gridSetup.bat -executeConfigTools -responsfile Grid_home\install\response
   \product_timestamp.rsp
   ```

   For Oracle Database:

   ```
   setup.exe -executeConfigTools -responseFile Oracle_home\install\response
   \product_timestamp.rsp
   ```

   For Oracle Database, you can also edit and use the response file located in the directory `Oracle_home\inventory\response\`:

   ```
   setup.exe -executeConfigTools -responseFile Oracle_home\inventory\response
   \db_install.rsp
   ```

   The postinstallation configuration tool runs the installer in the graphical user interface mode, displaying the progress of the postinstallation configuration.

   Specify the `[-silent]` option to run the postinstallation configuration in the silent mode.

   For example, for Oracle Grid Infrastructure:

   ```
   gridSetup.bat -executeConfigTools -responseFile Grid_home\install\response
   \grid_2016-09-09_01-03-36PM.rsp [-silent]
   ```

   For Oracle Database:

   ```
   setup.exe -executeConfigTools -responseFile Oracle_home\inventory\response
   \db_2016-09-09_01-03-36PM.rsp [-silent]
   ```

A.9 Postinstallation Configuration Using the ConfigToolAllCommands Script

You can create and run a response file configuration after installing Oracle software.

The `configToolAllCommands` script requires users to create a second response file, of a different format than the one used for installing the product. Starting with Oracle Database 12*c* Release 2 (12.2), the `configToolAllCommands` script is deprecated and may be desupported in a future release.

- About the Postinstallation Configuration File
  The configuration assistants are started with a script called `configToolAllCommands`.
- Creating a Password Response File
  Use these steps to create a password response file for use with the configuration assistants.

- Performing Postinstallation Configuration Using a Response File
  To run configuration assistants with the `configToolAllCommands` script in silent mode or response file mode, perform these steps.

> **See Also:**
>
> "Postinstallation Configuration Using Response File Created During Installation" for an alternate method of postinstallation configuration of Oracle software using the same response file created at the time of installation.

A.9.1 About the Postinstallation Configuration File

The configuration assistants are started with a script called `configToolAllCommands`.

When you perform an installation using silent mode or response file mode, you provide information about your servers in a response file that you otherwise provide manually using a graphical user interface. However, the response file does not contain passwords for user accounts that configuration assistants require after software installation is complete. To run the configuration assistants after the installation completes in silent mode, you must run the `configToolAllCommands` script and provide the passwords used by the assistants in a password file.

You can run the `configToolAllCommands` script in silent mode by using a password response file. The script uses the passwords in the file to run the configuration tools in succession to complete the software configuration. If you keep the password file to use when cloning installations, then Oracle strongly recommends that you store the password file in a secure location.

You can also use the password file to restart a failed installation. If you stop an installation to fix an error, then you can rerun the configuration assistants using `configToolAllCommands` and a password response file.

The `configToolAllCommands` password response file has the following options:

- `oracle.crs` for Oracle Grid Infrastructure components or `oracle.server` for Oracle Database components that the configuration assistants configure
- *variable_name* is the name of the configuration file variable.
- *value* is the desired value to use for configuration.

The command syntax is as follows:

```
internal_component_name|variable_name=value
```

For example, to set the password for the SYS user of Oracle ASM:

```
oracle.crs|S_ASMPASSWORD=myPassWord
```

A.9.2 Creating a Password Response File

Use these steps to create a password response file for use with the configuration assistants.

1. Create a response file that has a name of the format `filename.properties`.

2. Open the file with a text editor, and cut and paste the sample password file contents, as shown in the example below, modifying as needed.
3. If the file is stored on a volume formatted for Windows New Technology File System (NTFS), then modify the security permissions to secure the file.

Example A-5 Sample Password Response File for Oracle RAC Databases

This example provides a template for a password response file to use with the database configuration assistants.

```
oracle.install.db.config.starterdb.password.SYS=password
oracle.install.db.config.starterdb.password.SYSTEM=password
oracle.install.db.config.starterdb.password.DBSNMP=password
oracle.install.db.config.starterdb.password.PDBADMIN=password
oracle.install.db.config.starterdb.emAdminPassword=password
oracle.install.db.config.asm.ASMSNMPPassword=password
oracle.install.db.config.OracleHomeUserPassword=password
```

If you do not want to enable access for Oracle Enterprise Manager or Oracle ASM, then leave those password fields blank.

A.9.3 Performing Postinstallation Configuration Using a Response File

To run configuration assistants with the `configToolAllCommands` script in silent mode or response file mode, perform these steps.

1. Change directory to `%ORACLE_HOME%\cfgtoollogs`.
2. Run the configuration script using the following syntax:

   ```
   configToolAllCommands RESPONSE_FILE=\path\name.properties
   ```

Example A-6 Running Configuration Assistants in Response File Mode

Assume you created a password response file in the `C:\users\oracle\db` directory with a name of `cfg_db.properties`. To run the configuration assistants in response file mode to configure the Oracle software after installation, enter commands similar to the following:

```
C:\> cd %ORACLE_HOME%\cfgtoollogs
C:\..\cfgtoollogs> configToolAllCommands RESPONSE_FILE=C:\users\oracle\db\cfg
_db.properties
```

B

Directory Structure for Oracle RAC Environments

After you install the software, there are several directory structures for Oracle Real Application Clusters (Oracle RAC).

- Understanding the Oracle RAC Directory Structure
  When you install Oracle Database 12*c* Release 2 (12.2) with Oracle RAC, several directory structures are created.
- Directory Structures for Oracle RAC
  The sample Optimal Flexible Architecture (OFA)-compliant database has a hierarchical directory tree of folders.

B.1 Understanding the Oracle RAC Directory Structure

When you install Oracle Database 12*c* Release 2 (12.2) with Oracle RAC, several directory structures are created.

All subdirectories except for the Oracle Inventory directory, the Oracle Automatic Storage Management (Oracle ASM) home (if applicable), and the Oracle Clusterware home, are under a top-level Oracle base directory. The Oracle home and `admin` directories are also located under the Oracle base directory.

> **See Also:**
>
> *Oracle Database Installation Guide for Microsoft Windows* for more information about the Oracle home and `admin` directories

B.2 Directory Structures for Oracle RAC

The sample Optimal Flexible Architecture (OFA)-compliant database has a hierarchical directory tree of folders.

Table B-1 Directory Structure for a Sample OFA-Compliant Environment

Directory	Description
`%ORACLE_BASE%`	`C:\app\oracle` The default `ORACLE_BASE` directory, where the software was installed by the Oracle Installation user
`%ORACLE_BASE%\`*`installation_type`*	`C:\app\oracle\product\12.2.0` The type of installation under the Oracle base directory. For example, when installing Oracle Database 12*c* release 2 (12.2), the value for installation type is `product\12.2.0\db`.

Table B-1 (Cont.) Directory Structure for a Sample OFA-Compliant Environment

Directory	Description
%ORACLE_HOME% (%ORACLE_BASE%*installation_type**Home_name*)	The location in which the Oracle Database software is installed. You can also choose to add a counter, if you have multiple installations of the software. For example, if you are creating a second Oracle home for the Oracle Database 12*c* release 2 (12.2) software, then the path is as follows: `C:\app\oracle\product\12.2.0\dbhome_2` Note that the Oracle database home is under the Oracle base directory path. An Oracle Clusterware directory must *not* be under the Oracle base directory path where the database executable files are located.
%ORACLE_HOME%\database	The directory in which the local initialization parameter file is stored for the database.
%ORACLE_BASE%\admin	`C:\app\oracle\admin` The administrative directory. Note that with Oracle Database 11*g* and later releases, `bdump`, `cdump`, and `udump` files are relocated to the directory associated with `%ADR_BASE%`.
%ORACLE_BASE%\admin*db_unique_name*	The database unique name; this is the same as `dbname` when the database name is 8 or fewer characters in length. For example, if your database name is `sales`, the directory path is as follows: `C:\app\oracle\admin\sales`
%ORACLE_BASE%\admin*db_unique_name*\hdump %ORACLE_BASE%\admin*db_unique_name*\pfile	The dump destinations for the database server.
%ADR_BASE%	This directory path is set by the initialization parameter `DIAGNOSTIC_DEST`, and the path for the Automatic Diagnostic Repository must be located on the same location available to all the nodes. By default, this path is a subset of the Oracle base directory, in the following path: `%ORACLE_BASE%\diag\`
%ADR_BASE%\bdump %ADR_BASE%\cdump %ADR_BASE%\udump	Automatic Diagnostic Repository dump destination trace files.
Oracle Grid Infrastructure for a cluster home (Grid home)	An OFA-compliant path for the Oracle Clusterware home. The default value is`C:\app\12.2.0\grid`. During the Oracle Grid Infrastructure for a cluster installation, Oracle Clusterware and Oracle Automatic Storage Management (Oracle ASM) software is installed.
Grid_home\bin	The subtree for Oracle Clusterware and Oracle ASM executable files.
Grid_home\network	The subtree for Oracle Net Services configuration files and utilities.

C

Preparing to Upgrade an Existing Oracle RAC Database

Understand how you can prepare an Oracle Real Application Clusters (Oracle RAC) database for patch updates or upgrade.

- Backing Up the Oracle RAC Database
  Make a backup of the Oracle software installation before modifying the installed software.
- Using CVU to Validate Readiness for Oracle RAC Upgrades
  Review the contents in this section to validate that your Oracle RAC cluster is ready for upgrades.

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

C.1 Backing Up the Oracle RAC Database

Make a backup of the Oracle software installation before modifying the installed software.

- Before you make any changes to the Oracle software, Oracle recommends that you create a backup of the Oracle Database installation.

> **See Also:**
>
> - *Oracle Database Upgrade Guide* for information about creating a backup strategy
> - *Oracle Database Backup and Recovery User's Guide* for information about backing a database using RMAN
> - *Oracle Database Platform Guide for Microsoft Windows* for information about backing up a database using VSS

C.2 Using CVU to Validate Readiness for Oracle RAC Upgrades

Review the contents in this section to validate that your Oracle RAC cluster is ready for upgrades.

- Using the CVU Database Upgrade Validation Command Options
  Use the Cluster Verification Utility (CVU) to check the readiness of your Oracle RAC installation for upgrades.
- Example of Verifying System Upgrade Readiness for Oracle RAC Infrastructure
  You can verify that the permissions required for installing Oracle RAC have been configured on the nodes `node1` and `node2` using `cluvfy`.
- Verifying System Readiness for Oracle Database Upgrades
  To avoid interruptions during the upgrade process, you can use Cluster Verification Utility to ensure your system is ready.

C.2.1 Using the CVU Database Upgrade Validation Command Options

Use the Cluster Verification Utility (CVU) to check the readiness of your Oracle RAC installation for upgrades.

Purpose

Running `cluvfy` with the `-pre dbinst` and `-upgrade` options performs system checks to confirm if the cluster is in a correct state for upgrading from an existing Oracle RAC installation.

Command Syntax

```
cluvfy stage -pre dbinst -upgrade -src_dbhome src_RAChome [-dbname
db_names_list] -dest_dbhome dest_RAChome -dest_version dest_version
 [-dest_serviceuser username [-dest_servicepasswd]][-verbose]
```

Command Options

Table C-1 Command Options for CLUVFY Pre-upgrade Check

Command Option	Description
`-src_dbhome` *`src_RAChome`*	The location of the source Oracle RAC home that you are upgrading, where *src_RAChome* is the path to the home that you want to upgrade.
`-dbname` *`db_names_list`*	Optional: List of unique names of the databases being upgraded.
`-dest_dbhome` *`dest_RAChome`*	The location of the upgraded Oracle RAC home, where *dest_RAChome* is the path to the Oracle RAC home.

Table C-1 (Cont.) Command Options for CLUVFY Pre-upgrade Check

Command Option	Description
`-dest_version` *`dest_version`*	Use the `dest_version` option to indicate the release number of the upgrade, including any patchset. The release number must include the five digits designating the release to the level of the platform-specific patch, for example: 12.2.0.1.0.
`-dest_serviceuser` *`username`*	Optional: The Oracle Home user for the destination Oracle home.
`-dest_servicepasswd`	Optional: Prompt for the Oracle Home user password
`-verbose`	Use the `-verbose` option to produce detailed output of individual checks.

> **See Also:**
>
> *Oracle Database Administrator's Guide* for information about release number format.

C.2.2 Example of Verifying System Upgrade Readiness for Oracle RAC Infrastructure

You can verify that the permissions required for installing Oracle RAC have been configured on the nodes `node1` and `node2` using `cluvfy`.

```
C:\..bin> cluvfy stage -pre dbinst -upgrade
-src_dbhome C:\app\oracle\product\12.1.0\dbhome_1
-dest_dbhome C:\app\oracle\product\12.2.0\dbhome_1
-dest_version 12.2.0.0.0 -verbose
```

C.2.3 Verifying System Readiness for Oracle Database Upgrades

To avoid interruptions during the upgrade process, you can use Cluster Verification Utility to ensure your system is ready.

- Use Cluster Verification Utility to assist you with system checks in preparation for starting a database upgrade.

  The installer runs the appropriate CVU checks automatically, and prompts you to fix problems before proceeding with the upgrade.

D

Managing Oracle Database Port Numbers

Review the default port numbers. If needed, use these steps to change assigned ports after installation.

- About Managing Ports
  During installation, Oracle Universal Installer (OUI) assigns port numbers to components from a set of default port numbers. Many Oracle Real Application Clusters (Oracle RAC) components and services use ports.
- About Viewing Port Numbers and Access URLS
  In most cases, the Oracle Database component's port number is listed in the tool used to configure the port.
- Setting UDP and TCP Dynamic Port Range for Oracle RAC Installations
  For certain configurations of Oracle RAC in high load environments it is possible for the system to exhaust the available number of sockets. To avoid this problem, expand the dynamic port range for both UDP and TCP.
- Port Numbers and Protocols of Oracle Components
  Review this information for port numbers and protocols used by components that are configured during the installation. By default, the first port in the range is assigned to the component, if it is available.
- Changing the Oracle Services for Microsoft Transaction Server Port
  In most cases, you are not required to reconfigure the port number for the Oracle Services for Microsoft Transaction Server.

D.1 About Managing Ports

During installation, Oracle Universal Installer (OUI) assigns port numbers to components from a set of default port numbers. Many Oracle Real Application Clusters (Oracle RAC) components and services use ports.

As an administrator, it is important to know the port numbers used by these services, and to ensure that the same port number is not used by two services on your system.

Most port numbers are assigned during installation. Every component and service has an allotted port range, which is the set of port numbers Oracle RAC attempts to use when assigning a port. Oracle RAC starts with the lowest number in the range and performs the following checks:

- Is the port used by another Oracle Database installation on the system?

  The installation can be either active or inactive at the time; Oracle Database can still detect if the port is used.
- Is the port used by a process that is currently running?

  This could be any process on the host, including processes other than Oracle Database processes.

If the answer to any of the preceding questions is yes, then Oracle RAC moves to the next highest port in the allotted port range and continues checking until it finds a free port.

D.2 About Viewing Port Numbers and Access URLS

In most cases, the Oracle Database component's port number is listed in the tool used to configure the port.

- Ports for some Oracle Database applications are listed in the `portlist.ini` file. This file is located in the directory `%ORACLE_HOME%\install`.
- If you change a port number after installation, then it is not updated in the `portlist.ini` file, so you can rely on this file only immediately after installation.
- To find or change a port number, use the methods described in this appendix.

D.3 Setting UDP and TCP Dynamic Port Range for Oracle RAC Installations

For certain configurations of Oracle RAC in high load environments it is possible for the system to exhaust the available number of sockets. To avoid this problem, expand the dynamic port range for both UDP and TCP.

1. Open a command line window as an Administrator user.
2. Run the following commands to set the dynamic port range:

   ```
   netsh int ipv4 set dynamicport udp start=9000 num=56000
   netsh int ipv4 set dynamicport tcp start=9000 num=56000
   ```

3. Run the following commands to verify that the dynamic port range was set:

   ```
   netsh int ipv4 show dynamicport udp
   netsh int ipv4 show dynamicport tcp
   ```

   For IPv6 network, replace IPv4 with IPv6 in the above examples.

D.4 Port Numbers and Protocols of Oracle Components

Review this information for port numbers and protocols used by components that are configured during the installation. By default, the first port in the range is assigned to the component, if it is available.

Table D-1 Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol	Used Only On Interconnect
Cluster Manager The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	TCP	Yes

Table D-1 (Cont.) Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol	Used Only On Interconnect
Cluster Synchronization Service daemon (CSSD) The Cluster Synchronization Service (CSS) daemon uses a fixed port for node restart advisory messages. This port is used on all interfaces that have broadcast capability. Broadcast occurs only when a node eviction restart is imminent.	42424	Dynamic	TCP	Yes
Grid Plug and Play (GPNPD) GPNPD provides access to the Grid Plug and Play profile, and coordinates updates to the profile among the nodes of the cluster to ensure that all of the nodes have the most recent profile.	Dynamic	Dynamic	TCP	No
Multicast Domain Name Service (MDNSD) The mDNS process is a background process on Linux and UNIX, and a service on Windows, and is necessary for Grid Plug and Play and GNS.	5353	Dynamic	TCP	No
Oracle Cluster Registry The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	TCP	Yes
Oracle Clusterware Daemon (CRSD) Oracle Clusterware daemon internode connection. The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	TCP	Yes
Oracle Connection Manager Listening port for Oracle client connections to Oracle Connection Manager. You can configure Oracle Connection Manager after installation using NETCA.	1630	1630	TCP	No
Quality of Management Service (QOMS) Server The CRS Agent uses port 8888 locally to manage the lifecycle of the container.	8888	8888	HTTP	Not applicable
Quality of Management Service (QOMS) Server Port for the Quality of Management Service server.	23792	23792	JMX/RMI	No

Table D-1 (Cont.) Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol	Used Only On Interconnect
Oracle Data Guard Shares the Oracle Net listener port and is configured during installation. To reconfigure this port, use Oracle Net Configuration Assistant (NETCA) to reconfigure the listener.	1521 (same value as the listener)	modifiable manually to any available port	TCP	No
Oracle Event Manager (EVM) Generates events for Oracle Clusterware. The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	TCP	Yes
Oracle Grid Interprocess Communication (GIPCD) A support daemon that enables Redundant Interconnect Usage.	42424	Dynamic	TCP	Yes
Oracle Grid Naming Service (GNSD) The Oracle Grid Naming Service daemon performs name resolution for the cluster.	53	53	UDP	No
Oracle Grid Naming Service (GNSD) The Oracle Grid Naming Service daemon performs name resolution for the cluster.	Dynamic	Dynamic	TCP	No
Oracle HA Services daemon (OHASD) The Oracle High Availability Services (OHAS) daemon starts the Oracle Clusterware stack.	42424	Dynamic	TCP	Yes
Oracle Net Listener Allows Oracle clients to connect to the database by using Oracle Net Services. You can configure this port during installation. To reconfigure this port, use NETCA.	1521	Port number changes to the next available port. Modifiable manually to any available port.	TCP	No
Oracle Notification Services (ONS) Port for ONS, used for the publish and subscribe service for communicating information about Fast Application Notification (FAN) events. The FAN notification process uses system events that Oracle Database publishes when cluster servers become unreachable or if network interfaces fail. Use srvctl to modify ONS ports.	6100 (local) 6200 (remote)	Configured manually	TCP	No

Table D-1 (Cont.) Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol	Used Only On Interconnect
Oracle Real Application Clusters The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	9000 to 64999	UDP	Yes
Oracle Services for Microsoft Transaction Server The port number for Microsoft Transaction Server is configured automatically by Oracle Universal Installer (OUI) the first time you install the software on a particular server. If you install the software in multiple Oracle homes on the same server, then OUI uses the same port number for all installations. In most cases, you do not have to reconfigure the port number.	Dynamic	49152 to 65535	TCP	No
Oracle XML DB - FTP The Oracle XML DB FTP port is used when applications must access an Oracle database from an FTP listener. The port is configured during installation and you *cannot* view it afterward.	0	Configured manually	FTP	No
Oracle XML DB - HTTP The Oracle XML DB HTTP port is used if web-based applications must access an Oracle database from an HTTP listener. The port is configured during installation, and you *cannot* view it afterward.	0	Configured manually	HTTP	No

See Also:

- *Oracle XML DB Developer's Guide* for information about changing Oracle XML DB FTP port number.
- *Oracle XML DB Developer's Guide* for information about changing Oracle XML DB HTTP port number.
- *Oracle Enterprise Manager Cloud Control Advanced Installation and Configuration Guide* for information on Oracle Management Agent ports.
- "Changing the Oracle Services for Microsoft Transaction Server Port" for information about changing Oracle Services for Microsoft Transaction Server port number.

D.5 Changing the Oracle Services for Microsoft Transaction Server Port

In most cases, you are not required to reconfigure the port number for the Oracle Services for Microsoft Transaction Server.

- If you must change the port number, then you can use the Registry Editor to edit its value in the `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OracleMTSRecoveryService\Protid_0` Windows Registry key to any available port within the range 1024 to 65535.

  During installation, Oracle Universal Installer takes the value for the port from the key, if it exists. Otherwise, a free port ranging from 49152 to 65535 is chosen automatically.

Index

A

Administrators group, *1-2*
Advanced
    database configuration type, *2-7*, *2-9*, *2-10*
    installation type, *2-8*, *2-9*
architecture
    Optimal Flexible Architecture (OFA), *B-1*
architectures
    Optimal Flexible Architecture (OFA), *2-4*, *3-11*
ASMADMIN group, *2-15*
ASMDBA group, *2-15*
AUTHENTICATION_SERVICES, *6-24*
Automatic Diagnostic Repository (ADR), *6-24*
Automatic Memory Management, *2-12*
    about, *1-4*
automatic undo management, *6-7*

B

best practices, *4-5*

C

candidate disks, *3-9*
CDBs, *2-2*, *6-5*
    character sets, *2-12*
certification matrix, *1-4*
changing
    product languages, *2-8*
changing host names, *6-2*
character sets, *2-12*
CLEANUP_ORACLE_BASE, *xx*
cluster databases
    installed configuration, *6-6*
    server parameter files (SPFILE), *6-8*
Cluster Manager
    ports, ranges, and protocols, *D-2*
Cluster Verification Utility
    download location, *1-4*
    incorporated into OUI, *1-4*
    overview, *1-4*
CLUSTER_DATABASE_INSTANCES, *3-9*
cluvfy comp healthcheck, *4-5*
commands
    executeConfigTools, *A-16*
    gridSetup.bat, *A-11*
components
    created when using DBCA, *6-5*
configToolAllCommands, *A-19*
configToolAllCommands script, *A-18*
configuration types
    Advanced, *2-10*
connect descriptors, *6-9*
connection load balancing, *6-16*
Connection Manager
    See Oracle Connection Manager
control files
    described, *6-6*
    installed configuration, *6-6*
converting
    to Oracle Real Application Clusters from single-instance Oracle databases, *A-1*
creating Oracle RAC databases
    using DBCA, *3-7*
    using scripts, *A-2*
cron jobs, *1-4*

D

data files
    and DBCA, *6-6*
    described, *6-6*
Data Warehouse
    database configuration type, *2-10*
database
    components created by DBCA, *6-5*
    configuration types, *2-8*, *2-10*
    creating, *A-2*
    services, *6-9*
Database Agent
    and listeners, *6-22*
    process, *6-22*
Database Agent process, *6-13*
Database Configuration Assistant (DBCA), *6-19*
    components created by, *6-5*
    control files, *6-6*

Database Configuration Assistant (DBCA) *(continued)*
creating Oracle Real Application Clusters database
after installation, *3-7*
data files, *6-6*
deleting databases, *3-11*
initialization parameter files, *6-8*
Initialization Parameters page, *3-9*
List of Cluster Databases page, *3-11*
no longer sets LOCAL_LISTENER and REMOTE_LISTENER, *6-19*
Operations page, *3-11*
Oracle ASM Disk Groups page, *3-9*
redo log files, *6-7*
response file, *A-8*
rollback segments, *6-7*
running in silent mode, *A-12*
Summary dialog, *3-11*
tablespaces, *6-6*
using, *3-1*
daylight savings time
and TIMESTAMP WITH TIME ZONE data, *2-3*
DB_NAME, *3-9*
DB_UNIQUE_NAME, *3-9*
DBCA
configuring Automatic memory Management, *2-12*
dbca.rsp file, *A-8*
DBSNMP user
password requirements, *2-12*
dedicated servers, *6-17*
deinstallation
tasks, *7-1*
deinstallation tool
location of log files, *7-4*
syntax, *7-4*
Deinstallation Tool, *7-3*
deleting databases
using DBCA, *3-11*
deprecated features
service management, *2-13*
directory objects, *4-10*
directory structures, *B-1*
domain user
used in installation, *1-2*

E

Easy Connect Naming, *6-24*
enterprise.rsp file, *A-8*
executeConfigTools, *A-15*
external tables, *4-10*

F

failover
and service registration, *6-17*
files
dbca.rsp, *A-8*
enterprise.rsp, *A-8*
netca.rsp, *A-8*
Free server pool, *5-3*
described, *5-3*

G

General Purpose/Transaction Processing
database configuration type, *2-10*
generic server pool
described, *5-3*
Generic server pool, *5-3*
global database names
selecting, *2-10*
globalization, *1-4*
Group Managed service account, *1-2*
groups
Administrators group, *1-2*
ORA_DBA, *A-9*

H

healthchecks, *4-5*
high availability
SCAN listeners, *6-12*
host names
changing, *6-2*

I

initialization parameter files, *6-8*
listeners, *6-22*
parameters, *6-13*
initialization parameters
CLUSTER_DATABASE_INSTANCES, *3-9*
DB_NAME, *3-9*
DB_UNIQUE_NAME, *3-9*
DISPATCHERS, *6-17*
REMOTE_LISTENER, *6-13*
installation
directory structure, *B-1*
listener.ora, *6-13*
of additional products after installation is completed, *3-9*
response files
preparing, *A-7*, *A-9*
silent mode, *A-11*
tnsnames.ora files, *6-19*

installation option
Automatic Memory Management, 2-12
invalid objects
recompiling, 4-3

L

licensing, 1-4
listener
automatic migration from 10.1 to 10.2, 3-3
service registration, 6-23
listeners
configuring
listener.ora file, 6-21
default configuration of listener.ora, 6-13
files
listener.ora, 6-13, 6-22, 6-23
sqlnet.ora, 6-24
LOCAL_LISTENER, 6-19
managing, 6-23
registration, 6-14, 6-18
remote, 6-22
listeners:parameters
REMOTE_LISTENER, 6-19
load balancing
and service registration, 6-17
local listeners, 6-13, 6-22
log file
how to see the log file during installation, 2-1
LREG process
and listener registration, 6-14, 6-18
discovery routine, 6-14, 6-18

M

managing tablespaces, 6-7
MSA, 1-2
multiple Oracle homes
and Oracle Clusterware, 1-8
multiple Oracle RAC databases
Oracle Clusterware requirements for, 1-8
multitenant container database
character sets, 2-12
multitenant container databases
See CDBs
Multitenant databases, 3-1
My Oracle Support website
about, 1-4
accessing, 1-4

N

naming methods, 6-9
Easy Connect Naming, 6-24
Net Configuration Assistant
See Oracle Net Configuration Assistant (NETCA)
net service names, 6-20
netca.rsp file, A-8
networks
configuration files, 6-13
sqlnet.ora, 6-24
tnsnames.ora, 6-19
directories, B-1
noninteractive mode
See response file mode
NTS operating system authentication, 6-24

O

operating system
Administrators group, 1-2
Optimal Flexible Architecture (OFA), 2-4, 3-11, B-1
ORA_DBA group, A-9
ORAchk
and Upgrade Readiness Assessment, 1-4
Oracle ASM
and candidate disks, 3-9
Change disk discovery path, 3-9
response files, A-6
Oracle Cluster Registry port, D-2
Oracle Clusterware
and multiple database releases, 1-8
ports, ranges, and protocol, D-2
Oracle Connection Manager, D-2
ports, ranges, and protocols, D-2
Oracle Data Guard
ports, ranges, and protocols, D-2
Oracle Database
upgrades of, 2-7
Oracle Database Configuration Assistant
See Database Configuration Assistant (DBCA)
Oracle Database Upgrade Assistant, 3-3
Oracle Event Manager
ports, ranges, and protocols, D-2
Oracle home
ASCII path for, 2-5
multiple, 6-4
Oracle Home name, 2-5
Oracle Net Configuration Assistant (NETCA), A-14
response file, A-8
response files, A-14
running at command prompt, A-14
Oracle Net Services listener
ports, ranges, and protocols, D-2
Oracle Notification Services (ONS)
ports, ranges, and protocols, D-2
Oracle RAC deployment, 1-1
Oracle RAC One Node

Oracle RAC One Node *(continued)*
and server pools, *5-6*
Oracle Real Application Clusters, *6-1*
databases, deleting, *3-11*
overview, *3-1*, *6-2*
ports, ranges, and protocols, *D-2*
Oracle Real Application Clusters One Node
databases, creating, *3-10*
Oracle Restart
password file, *A-15*
Oracle Services for Microsoft Transaction Server
ports, ranges, and protocols, *D-2*
Oracle Universal Installer (OUI), *A-8*
response files, *A-8*
Oracle Upgrade Companion, *2-7*
Oracle XML DB
ports, ranges, and protocols, *D-2*
ORACLE_BASE
default value, *2-4*
OUI
See Oracle Universal Installer (OUI)

P

passwords, *2-12*
PDBs, *2-2*, *3-1*, *6-5*
PGA, *2-12*
and memory management, *1-4*
pluggable databases
See PDBs
Pluggable Databases, *3-1*
policy-based management, *5-2*
ports, *4-2*
access URLs, *D-2*
Cluster Manager, *D-2*
configured for applications, *D-2*
default ranges, *D-1*, *D-2*
Oracle Cluster Registry, *D-2*
Oracle Clusterware, *D-2*
Oracle Connection Manager, *D-2*
Oracle Data Guard, *D-2*
Oracle Event Manager, *D-2*
Oracle Net Services listener, *D-2*
Oracle Notification Services (ONS), *D-2*
Oracle Real Application Clusters, *D-2*
Oracle Services for Microsoft Transaction Server, *D-2*
Oracle XML DB, *D-2*
portlist.ini, *D-2*
postinstallation
configuration of Oracle software, *A-15*, *A-18*
preconfigured database installation types, *2-9*
proxy realm, *1-4*

R

RAC
See Oracle Real Application Clusters
readme.txt file, *D-2*
recommendations
secure response files after modification, *A-9*
recompiling invalid objects
on a non-CDB, *4-3*
redo log files
described, *6-7*
installed configuration, *6-7*
remote listeners, *6-22*
removing software, *7-1*
response file mode, *A-4*
about, *A-4*
installation, *A-11*
preparing, *A-7*
reasons for using, *A-5*
See also response files
response files, *A-4*
about, *A-4*
creating with template, *A-8*, *A-9*
dbca.rsp, *A-8*
enterprise.rsp, *A-8*
for Oracle ASM, *A-6*
general procedure, *A-7*
netca.rsp, *A-8*
Oracle Net Configuration Assistant (NETCA), *A-14*
passing values at command line, *A-4*
reasons for using, *A-5*
specifying with the installer, *A-11*
rollback segments
described, *6-7*

S

SCAN VIP, *6-11*
scripts to create an Oracle Real Application Clusters database, *A-2*
security
selecting passwords, *2-12*
server categorization, *5-2*
server hardware, *1-2*
server parameter files (SPFILE)
about, *6-8*
server pools, *5-1*
and Oracle RAC One Node, *5-6*
described, *5-3*
Free, *5-3*
Generic, *5-3*
service registration, *6-14*, *6-18*
about, *6-23*
configuring, *6-16*

services
singleton and uniform, 5-3
setup.exe, A-11
SGA, 2-12
and memory management, 1-4
shared servers, 6-17
SID
See system identifiers (SID)
silent mode
See response file mode
singleton services, 5-3
software
removing, 7-3
uninstalling, 7-1, 7-3
sqlnet.ora files, 6-24
SYS user
password requirements, 2-12
sysasm privilege
storage tasks requiring, 3-9
SYSAUX tablespace
description, 6-6
system identifier (SID)
selecting, 2-10
SYSTEM tablespace
description, 6-6
SYSTEM user
password requirements, 2-12

T

tablespaces
and DBCA, 6-6
SYSAUX, 6-6
SYSTEM, 6-6
TEMP, 6-6
UNDOTBS, 6-6
USERS, 6-6
TEMP tablespace
description, 6-6
time zone settings, 6-3
time zone version files, 2-3
TIMESTAMP WITH TIME ZONE (TSTZ) data, 2-3
TNS_ADMIN
and the listener.ora file, 6-13, 6-22
configuring, 6-23
tnsnames.ora files
and VIP addresses, 6-11
tnsnames.ora files *(continued)*
default configuration, 6-19
Transaction Processing
database configuration type, 2-10
troubleshooting
cron jobs and installation, 1-4
log file, 2-1
TSTZ
See time zone version files
Typical installation type, 2-8

U

undo management, 6-7
UNDOTBS tablespace
description, 6-6
uniform services, 5-3
uninstalling software, 7-1
upgrade
of existing Oracle Databases, 3-3
of listener, 3-3
upgrades
and SCANs, 6-12
and using the same Oracle home, 1-8
of existing Oracle databases, 2-7
of TIMESTAMP WITH TIME ZONE data, 2-3
Oracle Clusterware release requirement for, 1-8
upgrading
and ORAchk Upgrade Readiness Assessment, 1-4
user
domain, 1-2
Oracle Home User, 1-2
user authentication, 6-24
USERS tablespace
description, 6-6
utlrp.sql, 4-3
on a non-CDB, 4-3

V

VIP addresses, 6-11

W

Windows Firewall, 4-2
Windows Group Managed service account, 1-2