

Oracle® Database

Installation Guide

10g Release 2 (10.2) for hp OpenVMS

B25414-02

November 2006

Oracle Database Installation Guide, 10g Release 2 (10.2) for hp OpenVMS

B25414-02

Copyright © 2006, Oracle. All rights reserved.

Primary Authors: Namrata Bhakthavatsalam, Michelle D'Souza, Sanjay Sharma

Contributing Authors: Ed DeRosier, Kass McMahon-Kelly

Contributors: Grant Hayden, Dave Hayter, Gary Hodson, Steve Holck, Thomas Montgomery, Rajesh Parasuraman, Sam Ravindhran, Regina Rohr-Haase, Chris Schuetz, Kannan Viswanathan, Eric Wittenberg

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

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

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

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

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

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

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

Contents

Preface	xiii
Intended Audience.....	xiii
Documentation Accessibility	xiii
Related Documentation.....	xiv
Conventions	xiv
Third-Party Software Notices	xiv
1 Installation Overview	
Overview of Oracle Database Installation	1-1
Changes and Enhancements	1-2
Turning Off the Image Dump	1-2
Oracle Database Dependency on ODS-5 Disk Structure.....	1-3
Oracle Database Installation Directory Structure.....	1-4
Unsupported Initialization Parameters	1-4
Deprecated Initialization Parameters.....	1-5
Oracle Database Installation Types	1-5
Overview of Database Configuration Assistant	1-6
Database Configuration Options	1-7
Database Options	1-7
General Purpose.....	1-8
OLTP	1-8
Customized.....	1-8
Software Only.....	1-8
Installation Choices that Affect Database Creation.....	1-8
Creating a Database After Installation.....	1-9
Available Installation Options	1-9
Database Storage Options	1-9
Database Backup and Recovery Options.....	1-9
Important Compatibility Issues	1-11
Installation Considerations	1-11
Hardware and Software Certification	1-11
Multiple Oracle Homes	1-11
Logical Names and Symbols	1-12
Oracle Cluster Synchronization Services Daemon.....	1-12
Upgrade Issues	1-12

Manually Upgrade a Database.....	1-12
AL24UTFFSS Character Set	1-13
CRS/RAC upgrade	1-13
Cluster File System	1-13

2 Preinstallation Tasks

Checking Hardware Requirements	2-1
Minimum Hardware Requirements	2-2
Images	2-2
Disk Space Requirements.....	2-2
Oracle Advanced Security	2-3
Checking Network Requirements for Oracle RAC	2-4
Network Hardware Requirements	2-4
IP Address Requirements	2-4
Checking Each Node for Network Requirements	2-5
Checking Software Requirements	2-6
Installation-Specific Issues and Restrictions	2-11
New ORA_ROOT	2-11
JDK	2-11
Devices	2-11
Creating a User Account.....	2-12
Creating an Oracle Database DBA Account.....	2-12
Adding a Record in the User Authorization File.....	2-13
Setting Account Privileges.....	2-14
Setting Account Quotas.....	2-14
Process Rights for Database Administrators.....	2-15
Process Rights for SGA Protection.....	2-16
Security Issues with Multiple Databases	2-16
Completing Account Setup.....	2-16
Verifying Privileges	2-16
Setting Permissions for File Creation	2-16
SYSGEN Parameters	2-17
Reserved Memory Registry	2-17
AUTOGEN	2-18
System Configuration.....	2-19
Performing Setup Tasks As the SYSTEM User	2-20
Creating an Oracle Database Account	2-20
Creating an Apache Server Account	2-20
Performing Setup Tasks As the oracle user.....	2-21
Setting the Display	2-21
Setting Up a Batch Queue for the EM Agent and HAS	2-21
Setup Tasks for Oracle Products.....	2-22
Precompilers and Tools.....	2-22
Network and System Management Products	2-22
Setup Tasks for Oracle Net Services.....	2-22
Identifying Required Software Directories	2-23
Oracle Inventory Directory	2-23

Oracle Clusterware Home Directory.....	2-23
Oracle Home Directory	2-23
Guidelines for Placing Oracle Recovery Files	2-24
Creating Directories for Oracle Database or Recovery Files	2-24
Stopping Existing Oracle Processes.....	2-25
Configuring the oracle User's Environment	2-26
The ORATAB File.....	2-27
3 Installation Tasks	
Installation Overview.....	3-1
Reviewing Product-Specific Installation Guidelines	3-2
Downloading Oracle Software from the Oracle Technology Network Web Site.....	3-2
Downloading the Installation Archive Files	3-2
Installing Oracle Database Software.....	3-3
Installing Oracle Database from the Hard Drive.....	3-3
Installing Oracle Database from Install Media	3-4
Mount the DVD.....	3-4
Installing Oracle Database.....	3-4
Completing Oracle Database Server Enterprise Edition Installation.....	3-6
Installing Oracle Database Client.....	3-6
Reviewing the Log of an Installation Session	3-7
Cleaning Up After a Failed Installation.....	3-8
4 Postinstallation Tasks	
Database Configuration Assistant	4-1
Running Database Configuration Assistant in Interactive Mode.....	4-1
Using Database Configuration Assistant Response Files.....	4-2
Database Configuration Assistant Unsupported Options	4-3
Required Postinstallation Tasks	4-3
Downloading and Installing Patches	4-3
Configuring Oracle Products.....	4-3
Setting the NLS_LANG Environment Variable.....	4-3
Configuring New or Upgraded Databases	4-4
Changing User Passwords	4-4
Performing Configuration Tasks as the SYSTEM User	4-4
Performing Configuration Tasks as the oracle User	4-5
Required Product-Specific Postinstallation Tasks.....	4-5
Configuring Oracle Net Services	4-5
Configuring Oracle Label Security	4-6
Installing Oracle Text Supplied Knowledge Bases	4-6
Configuring Oracle Messaging Gateway	4-7
Oracle Precompilers.....	4-8
5 Installing Oracle RAC	
Oracle RAC Documentation Overview.....	5-1
Oracle Real Application Clusters Administrator's Guide.....	5-2

Oracle Real Application Clusters Deployment and Performance Guide	5-2
Documentation Sources	5-2
Quota Considerations	5-3
Cluster Setup and Preinstallation Configuration Tasks	5-3
Preinstallation, Installation, and Postinstallation Overview	5-4
Preinstallation Overview	5-4
Installation Overview	5-4
Postinstallation Overview	5-4
Oracle Universal Installer and Oracle Real Application Clusters	5-5
Additional Considerations	5-5
Oracle Database and Oracle Real Application Clusters Components	5-6
Oracle Clusterware	5-6
Installed Oracle Real Application Clusters Components	5-6
Oracle Real Application Clusters Version Compatibility	5-6
Installing Oracle Clusterware	5-6
Installation Setup	5-7
Installing Oracle Clusterware	5-7
Installation Tasks for Oracle Real Application Clusters	5-10
Selecting a Database Configuration Type	5-10
Configuration Type Descriptions	5-10
Using the Standard Configuration Types	5-11
Using the Advanced Configuration Type	5-11
Action of Oracle Universal Installer, Database Configuration Assistant, and Other Assistants During Installation	5-12
Procedure for Installation Setup	5-12
Installing Oracle Database with Oracle Real Application Clusters	5-13
Postinstallation Procedures for Oracle Real Application Clusters	5-16
Backing Up the Voting Disk After Installation	5-16
Downloading and Installing Patches	5-16
Configuring Oracle Products	5-17
Verifying Enterprise Manager Operations	5-17
Setting Up Users Accounts After Installation	5-18
Uninstalling Oracle Real Application Clusters Software	5-18
Procedure for Removing Oracle Real Application Clusters Software	5-19
Uninstalling Oracle Clusterware	5-20

6 Configuring Oracle RAC

Creating Directories	6-2
Guidelines for Placing Oracle Clusterware Files on a File System	6-2
Guidelines for Placing Oracle Database Files on a File System	6-2
Guidelines for Placing Oracle Database Recovery Files on a File System	6-3
Creating Required Directories	6-3
Using Database Configuration Assistant for Oracle RAC	6-4
Benefits of Using Database Configuration Assistant	6-4
Oracle RAC High-Availability Services	6-5
Service Configuration and Instance Preferences	6-5
Transparent Application Failover Policies	6-5

Creating the Database After Installation	6-5
Creating the Oracle RAC Database	6-6
Deleting an Oracle RAC Database	6-9
Configuring Oracle RAC	6-10
Parameter Files and Oracle RAC	6-10
Using Server Parameter Files in Oracle RAC	6-11
Location of the Server Parameter File.....	6-11
Parameter File Search Order in Oracle RAC	6-11
Upgrading to the Server Parameter File in Oracle RAC Environments	6-11
Server Parameter File Placement in Oracle RAC.....	6-12
Procedures for Upgrading to the Server Parameter File.....	6-12
Server Parameter File Errors in Oracle RAC	6-12
Understanding the Oracle RAC Installed Configuration	6-12
Understanding the Configured Environment in Oracle RAC.....	6-13
Oracle Cluster Registry in Oracle RAC.....	6-13
OpenVMS oratab Configurations for Oracle RAC.....	6-13
Database Components Created Using Database Configuration Assistant	6-13
Tablespaces and Data Files.....	6-14
Control Files.....	6-14
Redo Log Files.....	6-14
Managing Undo Tablespaces in Oracle RAC	6-14
Configuring Service Registration-Related Parameters	6-15
Configuring the Listener File (LISTENER.ORA)	6-15
Local Listeners.....	6-16
Multiple Listeners.....	6-16
How Oracle Database Uses the Listener (LISTENER.ORA).....	6-16
Listener Registration and PMON Discovery.....	6-17
Directory Server Access (LDAP.ORA File)	6-17
Net Service Names (TNSNAMES.ORA File)	6-17
Profile (SQLNET.ORA File)	6-21

7 Getting Started with Oracle Database

Reviewing Accounts and Passwords	7-1
Unlocking and Resetting User Passwords	7-3
Using Grid Control to Unlock Accounts and Reset Passwords.....	7-3
Using SQL*Plus to Unlock Accounts and Reset Passwords.....	7-4
Identifying Databases	7-4
Locating the Parameter Files	7-5
Locating the INIT.ORA File.....	7-5
Locating the Server Parameter File.....	7-5
Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files	7-6
Identifying Tablespaces and Data Files.....	7-6
Locating Redo Log Files.....	7-7
Locating Control Files.....	7-7
Adding or Upgrading Products	7-8
Preparing to Add a Product.....	7-8
Adding a Product Summary.....	7-9

Relinking the Oracle Code	7-9
8 Removing Oracle Software	
Identifying All Instances	8-1
Removing an Oracle Database.....	8-2
Reconfiguring Oracle Cluster Synchronization Services.....	8-2
Identifying Oracle Database Oracle Homes.....	8-3
Reconfiguring the Oracle Cluster Synchronization Services Daemon.....	8-3
Deleting the Oracle Cluster Synchronization Services Daemon Configuration.....	8-4
Removing Oracle Software.....	8-4
A Oracle Database 64-Bit Feature	
Introduction to Oracle Database 64-Bit Feature	A-1
Benefits of Oracle Database 64-Bit Feature.....	A-1
Implementation of Oracle 64-Bit Feature.....	A-2
Suggested Parameter Settings.....	A-2
Other Recommendations	A-3
B Noninteractive Installation and Configuration	
Silent Mode Installation	B-1
What Is a Response File?	B-1
Why Perform a Silent Installation?.....	B-2
Modifying a Response File.....	B-2
List of Response File Names.....	B-3
Performing Silent Installation	B-3
Creating the ORAINST.LOC File.....	B-3
Preparing Response Files	B-4
Running Oracle Universal Installer Noninteractively	B-4
Running Oracle Universal Installer in Silent Mode or Suppressed Mode	B-5
Running Oracle Network Configuration Assistant in Silent Mode	B-5
Running Database Configuration Assistant in Noninteractive or Silent Mode	B-6
Using Database Configuration Assistant in Noninteractive Mode	B-6
Using Database Configuration Assistant in Silent Mode.....	B-7
Running Database Configuration Assistant in Noninteractive or Silent Mode.....	B-7
C Oracle RAC: Other Considerations	
Documentation Sources	C-1
Troubleshooting the Oracle RAC Installation Process	C-2
Preconversion Considerations	C-2
Prerequisites for Conversion	C-2
Administrative Considerations.....	C-2
Converting from Single-Instance to Oracle RAC	C-2
Single-Instance Database on a Noncluster Computer to Oracle RAC	C-3
Backing Up the Original Single-Instance Database	C-3
Performing the Preinstallation Steps.....	C-3
Setting Up the Cluster	C-3

Copying the Preconfigured Database Image	C-4
Installing Oracle Database with Oracle RAC.....	C-4
Single-Instance Database on a Cluster to Oracle RAC	C-4
Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home	C-5
Automated Conversion.....	C-5
Manual Conversion	C-5
Single-Instance Database on a Cluster Running from a Oracle RAC-Disabled Oracle Home	C-6
Single-Instance Database on a Cluster Running from a Noncluster-Installed Oracle Home	C-7
Postconversion Steps	C-7
OpenVMS Directory Structures for Oracle RAC.....	C-8
D Installing, Configuring, and Running EMAgent	
Introduction to EM Framework.....	D-1
Installation Requirements	D-2
Installation and Configuration	D-2
Management and Maintenance	D-3
Startup, Shutdown, and Status.....	D-3
Troubleshooting and Maintenance.....	D-3
EMAgent Fails to Start	D-4
Extended File Specification (EFS) Characteristics	D-4
TMP Directory	D-4
Monitoring the Batch Queue	D-5
Disk Space on EMAgent Install Area.....	D-5
Resetting EMAgent Environment	D-5
Supported Targets and Jobs.....	D-5
Known Limitations	D-6
E Managing Oracle Database Port Numbers	
About Managing Ports	E-1
Viewing Port Numbers and Access URLs	E-1
Port Numbers and Protocols of Oracle Components	E-2
Changing the Oracle Enterprise Management Agent Port.....	E-2
F Configuring Oracle Database Globalization Support	
About NLS_LANG Logical.....	F-1
G Troubleshooting	
Verifying Requirements.....	G-1
X Window Display Errors	G-1
Errors During Installation	G-2
Reviewing the Log of an Installation Session	G-2
Troubleshooting Configuration Assistants	G-3
Configuration Assistant Failure	G-3
Fatal Errors.....	G-3

Silent Response File Error Handling.....	G-4
Cleaning Up After a Failed Installation.....	G-4

H Apache Server Installation and Configuration

Postinstallation Checklist	H-1
Running AUTOGEN.....	H-1
Check Disk Quota	H-1
Checking for SET TERMINAL/INQUIRE	H-2
Test the Installation	H-2
Browser Test.....	H-2
Troubleshooting	H-2
Running Oracle HTTP Server on OpenVMS	H-2
Starting and Stopping the Server	H-3
Server Log File	H-3
Performance Considerations	H-3
Limits and Quotas for Light to Moderate Load	H-3
Server with Medium to High Load	H-5
Excessive File Build Up.....	H-5
Customizing the Server Environment.....	H-5
How to Configure Apache.....	H-5
Modules and Directives	H-6
Supported and Unsupported Features	H-6
Modules Not Included	H-6
Unsupported Directives.....	H-6
Command-Line Options	H-8
Virtual Host Support	H-8
Dynamic Shared Object Support	H-9
File Handlers.....	H-9
Content Negotiation	H-9
Apache API.....	H-9
suEXEC Support.....	H-9
File Formats.....	H-9
File Naming Conventions	H-10
File Transfer Process and Access Control List	H-10
Logical Names	H-10
OpenVMS Cluster Considerations	H-12
Individual System Versus Clusterwide Definition	H-12
Mixed-Architecture Cluster.....	H-13
CGI Programs	H-13
CGI Environment Logical.....	H-13
Referencing Input.....	H-14
Running CGI Images	H-14
Logical Names for Debugging CGI Scripts.....	H-14
Displaying Graphics with CGI Command Procedures.....	H-14
Security Information	H-15
Process Model.....	H-15
Privileged Images.....	H-15

Privileges Required to Start and Stop the Server	H-16
File Ownership and Protection	H-17
Server Extensions (CGI Scripts)	H-17
suEXEC Not Available for Protecting Script Execution	H-17
Open Source Licenses	H-17
Apache	H-17

I Identifying Listener Processes

Index

Preface

This guide and *Oracle Database Administrator's Reference* provide instructions for installing and configuring Oracle Database 10g on the following platforms:

- hp OpenVMS Alpha
- hp OpenVMS Itanium

Documentation for the products that are included with this software is in the Oracle Database generic documentation set.

For important release information related to this product, refer to *Oracle Database Release Notes for hp OpenVMS*.

Intended Audience

This document is intended for anyone responsible for installing Oracle Database 10g on hp OpenVMS Alpha or hp OpenVMS Itanium systems.

Documentation Accessibility

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

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

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

Related Documentation

Information about system administration and tuning for a production database system is provided in the following documents:

- *Oracle Database Administrator's Reference*
- *Oracle Database Administrator's Guide*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Database Net Services Reference*
- *Oracle Database Performance Tuning Guide*

Information about upgrading from a previous version of Oracle Database is provided in *Oracle Database Upgrade Guide*.

The platform-specific documentation for Oracle Database 10g products includes the following manuals:

- *Oracle Database Installation Guide*
- *Oracle Database Release Notes for hp OpenVMS*
- *Oracle Database Administration's Reference*

Refer to Oracle Database Release Notes for your platform for important information that was not available when this book was released. The Release Notes for Oracle Database 10g are updated regularly. You can get the most recent version from Oracle Technology Network at

<http://www.oracle.com/technology/documentation/index.html>

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.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.
<i>italics</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Third-Party Software Notices

This program contains third-party software from Hewlett-Packard. The Oracle program license that accompanied this product determines your right to use the Oracle program, including the Hewlett-Packard software. Notwithstanding anything to the contrary in the Oracle program license, the Hewlett-Packard software is

provided "as is" and without intellectual property indemnities, warranties, or support of any kind from Oracle or Hewlett-Packard.

This program contains third-party software from International Business Machines Corporation (IBM). The Oracle program license that accompanied this product determines your right to use the Oracle program, including the IBM software.

Notwithstanding anything to the contrary in the Oracle program license, the IBM software is provided "as is" and without intellectual property indemnities, warranties, or support of any kind from Oracle or IBM.

Installation Overview

This chapter describes the different types of Oracle Database installations that you can perform, as well as issues that you should consider before installing the software. It includes information about the following topics:

- [Overview of Oracle Database Installation](#)
- [Changes and Enhancements](#)
- [Oracle Database Installation Types](#)
- [Overview of Database Configuration Assistant](#)
- [Database Configuration Options](#)
- [Available Installation Options](#)
- [Important Compatibility Issues](#)
- [Installation Considerations](#)
- [Upgrade Issues](#)
- [Cluster File System](#)

Overview of Oracle Database Installation

The Oracle Database installation process consists of five phases:

1. **Read the release notes:** Read the Oracle Database 10g release 2 (10.2) release notes before you begin the installation. The release notes are available with the platform-specific documentation. The latest version of the release notes is available on Oracle Technology Network at:
<http://www.oracle.com/technology/documentation/index.html>
2. **Planning the installation:** This chapter describes the Oracle products that you can install and issues that you must consider before starting the installation.
3. **Completing preinstallation tasks:** [Chapter 2](#) describes preinstallation tasks that you must complete before installing the product.
4. **Installing software:**
 - [Chapter 3](#) describes how to use Oracle Universal Installer to install Oracle Database.

If you want to install client tools, applications, and client interfaces that are not included on the Oracle Database DVD, then refer to the documentation for those products.

-
- [Chapter 5](#) describes how to install Oracle RAC.
 - [Chapter 8](#) describes how to remove Oracle Database.
 - [Appendix B](#) provides information on performing noninteractive (silent) installations, which you may want to use if you need to perform multiple installations of Oracle Database.
 - [Appendix D](#) describes how to install, configure and run the EMAgent.
 - [Appendix G](#) provides troubleshooting advice in case you encounter problems with the installation.
 - [Appendix H](#) describes how to install and configure Apache Server.
 -
5. **Completing postinstallation tasks:**
- [Chapter 4](#) describes recommended and required postinstallation tasks.
 - [Chapter 6](#) explains how to configure Oracle RAC.

Changes and Enhancements

To make full use of the latest OpenVMS CPU and compiler optimizations in Oracle software products, Oracle Database 10g release 2 (10.2) and associated Oracle products that are based on Oracle Database 10g requires an OpenVMS Alpha system with an EV56 or later generation CPU. This restriction is not applicable to the VMS Itanium platform

Some older/slower hardware with EV56 chipset might not work.

Note: Oracle Database 10g Release 2 (10.2) is not certified on and is, therefore, not supported on Alpha chip sets earlier than EV56.

See Also: Note 181307.1 on *OracleMetaLink* for a list of affected models and for instructions about how to identify the CPU model number

The following changes and enhancements are unique to this release of Oracle Database:

- [Turning Off the Image Dump](#)
- [Oracle Database Dependency on ODS-5 Disk Structure](#)
- [Oracle Database Installation Directory Structure](#)
- [Unsupported Initialization Parameters](#)
- [Deprecated Initialization Parameters](#)

Turning Off the Image Dump

In earlier releases, when one of the background processes terminated abruptly with an access violation, no useful information was readily available regarding the cause for termination. For troubleshooting, certain actions needed to be taken, and the failure needed to be reproduced. Starting with this release, by default, an image dump is created when a background process terminates abruptly. If you do not want an Oracle

background process to generate image dump files, then create an OpenVMS system logical for that background process as follows:

```
$ DEFINE/SYSTEM ORA_sid_bg process_DMP FALSE
```

In the preceding command, the value of *bg process* can be *pmon*, *lmd0*, *lms0*, and so on. The value of *sid* should be set to the required system Identifier or instance name.

Start the instance, and open the database. For example:

Note: OpenVMS logicals must be defined prior to starting the instance.

```
$ DEFINE/SYSTEM ORA_PROD_LMD0_DMP FALSE
$ SQLPLUS/NOLOG
SQL> connect / as sysdba
SQL> startup
```

This change has no effect on the usual daily production activities. Oracle recommends that you do not define the *dump* logical.

Oracle Database Dependency on ODS-5 Disk Structure

On-Disk Structure Level 5 (ODS-5) format enables OpenVMS to support Microsoft Windows file name formats. Longer file names, additional character support, and the ability to have lowercase and mixed-case file names are some of the features supported on ODS-5 disks.

Caution: Oracle recommends that you read *OpenVMS Guide to Extended File Specifications* before changing the structure of any disk. This guide describes ODS-2 and ODS-5 formats in more detail and provides important information that you must follow.

The disks on which the Oracle code tree resides must be formatted in Files-11 ODS-5 format. This is the disk to which the OpenVMS logicals *ORA_ROOT*, *ORA_ROOTDIR*, and *ORACLE_HOME* point. The directory that is referenced by these logicals is the parent directory for all Oracle Database software, such as *Apache*, *CTX*, *INSTALL*, *NETCONFIG*, *RDBMS*, and *UTIL*. The disks on which Oracle Database files reside can be formatted as either ODS-2 or ODS-5.

If you downloaded Oracle 10g R2 from OTN or copy the kit from the DVD to disk, make sure that the location of the software kit is in ODS-5 format.

See Also: Oracle Note 377458.1 on Oracle MetaLink

Alternatively, use the search function and search for ODS-5 for more information

Precautions

You must observe the following precautions:

- Do not change the format of any OpenVMS disk without first making a backup of the disk.
- Conversion from ODS-2 to ODS-5 does not require you to restore files.

-
- Conversion from ODS-5 to ODS-2 requires you to restore files.

Note: Contact Hewlett-Packard for more information about reverting to ODS-2 format from ODS-5 format.

Checking Disk Format

To check the current disk format, run the following DCL command:

```
$ SHOW DEVICE/FULL device_name
```

The `VOLUME STATUS:` line of the output displays the current ODS level. The default when initializing disks is ODS-2.

Changing Disk Format

If you plan to reformat an existing (in use) disk from ODS-2 to ODS-5 format, then:

1. Create at least one OpenVMS backup of the disk.
2. Use the following command:

```
$ SET VOLUME /STRUCTURE_LEVEL=5 device_name
```

Example:

```
$ SET VOLUME/STRUCTURE_LEVEL=5 $2$DKC100:
```

Formatting a New Disk

If you plan to use a new (empty) disk, then use the following command to format the new disk to ODS-5 format:

```
$ INITIALIZE/STRUCTURE=5 device_name:volume_label
```

Example:

```
INITIALIZE/STRUCTURE=5 $2$DKC100: TESTVOL
```

Oracle Database Installation Directory Structure

The installation directory structure of Oracle Database closely resembles that of the installation directory structure of Oracle Database on UNIX systems. By making the OpenVMS installation directory structure similar to that of other platforms, a more consistent and predictable layout is followed across platforms. This enables database administrators (DBAs) from other platforms to quickly become familiar with Oracle Database for this platform.

Unsupported Initialization Parameters

Support for the following Oracle Database initialization parameters found in earlier releases has been removed starting with this release of Oracle Database.

- `ALWAYS_ANTI_JOIN`
- `ALWAYS_SEMI_JOIN`
- `JOB_QUEUE_INTERVAL`
- `OPTIMIZER_PERCENT_PARALLEL`
- `HASH_MULTIBLOCK_IO_COUNT`

-
- DB_BLOCK_LRU_LATCHES
 - DB_BLOCK_MAX_DIRTY_TARGET
 - SORT_MULTIBLOCK_READ_COUNT
 - DB_FILE_DIRECT_IO_COUNT
 - GC_DEFER_TIME
 - GC_RELEASABLE_LOCKS
 - GC_ROLLBACK_LOCKS
 - LM_LOCKS
 - LM_RESS
 - USE_INDIRECT_DATA_BUFFERS

Deprecated Initialization Parameters

The following initialization parameters are deprecated starting with this release of Oracle Database.

- ROLLBACK_SEGMENTS
- FAST_START_IO_TARGET
- TRANSACTIONS_PER_ROLLBACK_SEGMENT
- LOG_CHECKPOINT_INTERVAL
- DB_BLOCK_BUFFERS
- BUFFER_POOL_KEEP
- BUFFER_POOL_RECYCLE

Oracle Database Installation Types

You can choose one of the following installation types when installing Oracle Database:

- **Enterprise Edition:** Installs licensable Oracle Database options and database configuration and management tools. It also installs products most commonly used for data warehousing and transaction processing.
- **Custom:** Enables you to select the individual components that you want to install from a list of available components.

Note: Oracle Database Client is installed in a separate Oracle home. Do not install Oracle Database Client in the Oracle home used for Oracle Database software.

Oracle Enterprise Manager 10g Agent is available in a separate Enterprise Manager Grid Control release.

Standard Edition is not offered on VMS and does not appear as an installation type.

See Also:

- *Oracle Enterprise Manager Concepts and Oracle Enterprise Manager Grid Control Installation and Basic Configuration* for more information about Oracle Enterprise Manager 10g
- *Oracle Database Licensing Information* guide for more information about the features available with each Oracle Database edition and information about licensing

Overview of Database Configuration Assistant

Database Configuration Assistant enables you to copy a preconfigured Oracle Database, or to create a fully customized database to match the selected environment and database configuration. It starts automatically (unless you selected software only) after Oracle Database software has been installed during an installation session.

When installing Oracle Database software using any database configuration option other than Custom and Software Only, Oracle Universal Installer prompts for a global database name and system identifier (*SID*). After Oracle Database installation is complete, Database Configuration Assistant uses this information to create a database.

The following list provides descriptions of database configurations:

- **General Purpose**

If you select this option with an Enterprise Edition installation, then Database Configuration Assistant creates a preconfigured, ready-to-use, multipurpose database with the following features:

 - Default initialization parameters using server parameter file and Automatic Undo Management
 - Automatic installation and configuration of Oracle Options and *interMedia*
 - Advanced Replication capabilities
 - Database configuration of dedicated server mode
 - Archiving mode of NOARCHIVELOG
- **Transaction Processing**

If you select this option, then Database Configuration Assistant creates a preconfigured, ready-to-use OLTP database.
- **Data Warehouse**

If you select this option, then Database Configuration Assistant creates a database that is fully enabled for data warehousing applications.
- **Customized**

If you select this option, then Database Configuration Assistant guides you in creating a fully customized database.
- **Automatic or Manual**

Automatic or manual configuration options, Oracle Text components, and advanced replication.

Select this option only if you are experienced with advanced database creation procedures.

You will need to customize some or all of the following settings or parameters:

-
- Data, control, and redo log file settings
 - Tablespace and extent sizes
 - Database memory parameters
 - Archiving modes, formats, and destinations
 - Trace file destinations
 - Character set values
 - Software Only

If you select this option, then Oracle Universal Installer only installs Oracle Database software. Database Configuration Assistant does not start after installation, and no preconfigured database is installed.

If a choice other than the Software Only option is selected for database configuration, then Oracle Universal Installer starts Database Configuration Assistant during the installation session. If a "Software Only" installation is performed, then Database Configuration Assistant can be run later from the command line.

See Also:

- [Section , "Running Database Configuration Assistant in Noninteractive or Silent Mode"](#) for more information about running Database Configuration Assistant from the command line
- *Oracle Database Globalization Support Guide* for more information about database character sets

Database Configuration Options

During the installation, you can choose whether you want to create an Oracle Database as part of the installation. If you choose to create an Oracle Database, Oracle Universal Installer uses Database Configuration Assistant to create it. You can choose to create one of the preconfigured database types, which are designed for different types of applications, modify one of the preconfigured database types, or create a customized database to suit your requirements.

This section describes the following database configuration options:

- [Database Options](#)
- [Installation Choices that Affect Database Creation](#)
- [Creating a Database After Installation](#)

Database Options

Oracle provides the following preconfigured database types that you can create or customize during the installation:

- [General Purpose](#)
- [OLTP](#)
- [Customized](#)
- [Software Only](#)

General Purpose

Users of this database type perform a variety of database tasks, ranging from simple transactions to complex queries. Select this database environment for general purpose use.

OLTP

Users of this database type perform large numbers of concurrent transactions, in which each transaction is a relatively simple operation that is processing a small amount of data.

Billing databases, such as those commonly found on Internet commerce Web sites, are the most common example of this database configuration. These databases are also known as Online Transaction Processing (OLTP) databases.

Customized

This option enables you to create a customized database configuration of Oracle products that meets specialized requirements. Select this option only if you are prepared to provide detailed product and database environment choices. Processing this option involves a longer installation session than choosing a preconfigured database.

Software Only

This option enables you to install Oracle products. It does not configure the listeners or the network infrastructure, and it does not create a database.

Oracle recommends that you install at least one preconfigured database to serve as a template for database configuration.

Installation Choices that Affect Database Creation

Oracle Universal Installer runs Database Configuration Assistant in one of two modes, depending on the choices that you make during the installation:

- Noninteractive mode

If you choose the Enterprise Edition installation type, then choose a preconfigured database type. Oracle Universal Installer prompts you for the minimum amount of information required to create a database of the type you choose. It then runs Database Configuration Assistant in noninteractive mode to create the database after it installs the software.

Note: Oracle recommends that you use this method to create a database if you have not previously created one.

- Interactive mode

If you choose the Custom installation type, then Oracle Universal Installer does not prompt you for database information. Instead, it installs the software and then runs Database Configuration Assistant in interactive mode. Using the screens in Database Configuration Assistant, you can either modify one of the preconfigured database types or create a custom database and specify how you want to configure it.

Note: If you choose this method to create a database, then click **Help** on any of the Database Configuration Assistant screens for a description of the information that you must specify on that screen.

Creating a Database After Installation

If you decide not to create a database during the installation, then you can use Database Configuration Assistant to create one after you have installed the software.

See Also: *Oracle Database 2 Day DBA* for more information about using Database Configuration Assistant to create a database after installation

Available Installation Options

The current release of Oracle Database provides new options for simplifying database administration tasks. These include:

- [Database Storage Options](#)
- [Database Backup and Recovery Options](#)

Database Storage Options

If you choose to create a database during the installation, then you can specify the File System option as the storage option for database files.

File System

Automated Storage Management (ASM) is not supported as a storage option in this release. If you choose the File System option, then Database Configuration Assistant creates the database files in a directory on a device mounted on the system. Oracle recommends that the device that you select be separate from the devices that are used by the operating system or by Oracle products. The device that you select can be any of the following:

- A standalone device
- A device on a logical volume manager (LVM) volume or RAID device

If you are using multiple disks in an logical volume manager (LVM) or RAID configuration, then Oracle recommends that you use the stripe-and-mirror-everything (SAME) methodology to increase performance and reliability. Using this methodology, you do not need to specify more than one device for database storage.

See Also: *Oracle Database Administrator's Guide*

Database Backup and Recovery Options

If you choose to use Oracle Enterprise Manager Grid Control during the installation, then you can enable automated database backups that use the default backup strategy recommended by Oracle.

Note: You do not have to enable automated backups during the installation. If you prefer, you can use Oracle Enterprise Manager Grid Control to configure automated backups after you install the software and create a database.

This section contains the following sections:

- [Enabling Automated Backups](#)
- [Backup Job Default Settings](#)

Enabling Automated Backups

If you enable automated backups, then Oracle Enterprise Manager schedules a daily backup job that uses Oracle Recovery Manager (RMAN) to back up all the database files to a storage area on disk called the flash recovery area. The first time the backup job runs, it creates a full backup of the database. Subsequent backup jobs perform incremental backups, which enable you to recover the database to its state at any point during the preceding 24 hours.

To enable automated backup jobs during installation, you must have previously installed the Oracle Enterprise Manage Grid Control Agent. During database installation you must specify the following information:

- The location of the flash recovery area
You can choose a device directory or accept the default for the flash recovery area. The default disk quota configured for the flash recovery area is 2 GB.
- An operating system user name and password for the backup job
Oracle Enterprise Manager uses the operating system credentials that you specify when running the backup job. The user name that you specify must have the appropriate privileges and rights required for database administrators.

See Also: [Chapter 2](#) for information about choosing the location of the flash recovery area and determining the disk space requirements and about the requirements for the database administrator account

Backup Job Default Settings

If you enable automated backups after choosing one of the preconfigured databases during the installation, then automated backup is configured with the following default settings:

- The backup job is scheduled to run nightly at 2 a.m.
- The disk quota for the flash recovery area is 2 GB.

If you enable automated backups by using Database Configuration Assistant, either during or after the installation, you can specify a start time for the backup job and a disk quota for the flash recovery area.

See Also:

- *Oracle Database 2 Day DBA* for information about using Oracle Enterprise Manager Grid Control to configure or customize automated backups or to recover a backed up database
- *Oracle Database Backup and Recovery Basics* manual or *Oracle Database Backup and Recovery User's Guide* for more information about defining a backup strategy and about backing up and recovering Oracle Databases

Important Compatibility Issues

Caution: If you are installing Oracle Database 10g on a node where other versions of Oracle Database are running, then ensure that you have no Oracle symbols and logicals defined prior to installing Oracle Database 10g. Do not run any `orauser` scripts as part of the login sequence. Failure to heed this warning will result in numerous problems, including undefined symbols and overwriting of the previous code tree.

The following are compatibility issues that you must address:

- Oracle Database 10g Enterprise Edition must be installed in a location separate from other previous Oracle Database installations.
- Oracle products released for earlier versions of Oracle Database (such as release 9.2.0.x and release 8.1.7.x) must not be linked with an Oracle Database 10g installation.

Installation Considerations

This section contains information about the following considerations, which you must address before deciding how to install Oracle Database:

- [Hardware and Software Certification](#)
- [Multiple Oracle Homes](#)
- [Logical Names and Symbols](#)
- [Oracle Cluster Synchronization Services Daemon](#)

Hardware and Software Certification

The hardware and software requirements included in this installation guide are current at the time of publishing this guide. However, because new platforms and operating system software versions may be certified after this guide is published, review the certification matrix on the *OracleMetaLink* Web site for an updated list of certified hardware platforms and operating system versions. You can visit the *OracleMetaLink* Web site at

<https://metalink.oracle.com>

If you do not have a current Oracle Support Services contract, then you can access the same information from the Oracle Technology Network Web site at

<http://www.oracle.com/support/metalink/content.html>

Multiple Oracle Homes

This product supports multiple Oracle homes. This means that you can install this release or previous releases of the software more than once on the same system.

Installing the Software on a System with an Existing Oracle Installation

You must install this product into a new Oracle home directory. You cannot install products from one release of Oracle Database into the Oracle home directory of a different release. For example, you cannot install Oracle Database 10g software into an

existing Oracle9i Oracle home directory. If you attempt to install this release into an Oracle home directory that contains software from an earlier Oracle release, then the installation fails.

You can install this release more than once on the same system if each installation is installed in a separate Oracle home directory.

Logical Names and Symbols

Note: Before installing Oracle software, you must log out of any existing Oracle sessions. If you do not do this, then the new installation includes the logicals and symbols of the active Oracle sessions.

Ensure that the `LOGIN.COM` file of the account that is used to install Oracle Database does not define any of the Oracle logicals or symbols, and does not run any command file that may define them. Ensure that none of the general Oracle specific logicals (typically beginning with `ORA_`) are defined in the system table, except for some of the logicals related to mailbox devices and shared libraries. Oracle Database may not run correctly if these logicals are defined.

Oracle Cluster Synchronization Services Daemon

For Oracle RAC installations, the Oracle Cluster Synchronization Services (CSS) daemon is installed with Oracle Clusterware in a separate Oracle home directory.

See Also: [Section , "Reconfiguring Oracle Cluster Synchronization Services"](#) for more information about deleting and reconfiguring the Oracle Cluster Synchronization Services daemon

Upgrade Issues

For more information about upgrading a previous release of Oracle Database to Oracle Database 10g, refer to *Oracle Database Upgrade Guide*. The following sections includes additional upgrade information that you should review before upgrading an existing database.

Manually Upgrade a Database

To manually upgrade an existing database to Oracle Database 10gR2 on OpenVMS, follow the steps in the *Oracle Database Upgrade Guide*. In Chapter 3 of the guide, use the following steps for OpenVMS.

In the "Prepare the New Oracle Home" subsection of the "Upgrade the Database Manually" section, substitute `ORACLE_HOME/dbs` with `ORA_ROOT:[DBS]`. Also substitute `ORACLE_HOME/dbs/orapwsid` with `ORA_ROOT:[DBS]orapwsid`.

- In the "Upgrade the Database" subsection, ignore step 1 (for Windows operating system).
- Set your default directory to the new Oracle home and execute the script `@ORAUSER.COM` (without arguments). Then `SET DEF ORA_ROOT:[RDBMS]` and execute the migrate script:
 - To upgrade 8.1.7 or 9.0.1 to 10gR2, enter `@MIGRATE89.COM`

-
- To upgrade from 9.2 to 10gR2, enter @MIGRATE92.COM
 - To upgrade from 10gR1 to 10gR2, enter @MIGRATE101.COM

Enter the details prompted by the upgrade assistant.

After you complete this step, proceed with Step 4 in this subsection, which involves changing to the `ORA_ROOT: [RDBMS.ADMIN]` directory.

If you want to use an existing OpenVMS Alpha database on an OpenVMS Itanium system (Integrity Server), you must upgrade the database to 10.2 on Alpha before transferring it to Itanium.

AL24UTFFSS Character Set

To upgrade an existing database that uses the AL24UTFFSS character set, upgrade the database character set to UTF8 before upgrading to Oracle Database10g. Oracle recommends that you use the Character Set Scanner (CSSCAN) utility for data analysis before attempting to upgrade the existing database character set.

The Character Set Scanner utility checks all character data in the database and tests for the effects of, and problems with, changing the character set encoding.

CRS/RAC upgrade

Please refer to file `readme_vms_10grac.txt` that is shipped in CRS shiphome for complete details about CRS/RAC upgrade from 10gR1 to 10gR2.

The file location in the CRS software distribution will be at:

```
DISK$MOUNT [CLUSTERWARE] README_VMS_10GRAC.TXT
```

Cluster File System

In this document, there are references to the term `cluster file system`. This term refers to the configuration in which all disks are equally accessible from all nodes on the cluster. This is the most commonly used configuration on an OpenVMS cluster. All references to this term should be considered as references to this configuration on an OpenVMS cluster.



Preinstallation Tasks

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

- [Section , "Checking Hardware Requirements"](#)
- [Section , "Checking Network Requirements for Oracle RAC"](#)
- [Section , "Checking Software Requirements"](#)
- [Section , "Installation-Specific Issues and Restrictions"](#)
- [Section , "Creating a User Account"](#)
- [Section , "SYSGEN Parameters"](#)
- [Section , "System Configuration"](#)
- [Section , "Performing Setup Tasks As the SYSTEM User"](#)
- [Section , "Performing Setup Tasks As the oracle user"](#)
- [Section , "Setup Tasks for Oracle Products"](#)
- [Section , "Identifying Required Software Directories"](#)
- [Section , "Guidelines for Placing Oracle Recovery Files"](#)
- [Section , "Creating Directories for Oracle Database or Recovery Files"](#)
- [Section , "Stopping Existing Oracle Processes"](#)
- [Section , "Configuring the oracle User's Environment"](#)
- [Section , "The ORATAB File"](#)

Checking Hardware Requirements

Note: Oracle Database 10g must be installed on an ODS-5 formatted disk. Refer to [Section , "Changes and Enhancements"](#) for information about this.

The following sections list the minimum hardware requirements for installing Oracle Database products on an OpenVMS system.

- [Section , "Minimum Hardware Requirements"](#)
- [Section , "Images"](#)
- [Section , "Disk Space Requirements"](#)

- [Section , "Oracle Advanced Security"](#)

Minimum Hardware Requirements

The following are the minimum hardware requirements for installing Oracle Database 10g.

RAM

A minimum of one GB RAM is required to install Oracle Database products. Oracle Database Client products require 512 MB of RAM.

To determine the amount of RAM installed on the system as well as the amount of paging currently configured on the system, enter the following command:

```
$ SHOW MEMORY
```

To review memory reservations on the node, run the following command:

```
$ SHOW MEMORY/RESERVED
```

```
System Memory Resources on 13-JUL-2002 09:57:11.72
```

```
Memory Reservations (pages):  Group  Reserved  In Use  Type
ORA_PROD_SGA                  SYSGBL  5120      0      Allocated
ORA_PROD_SGA                  SYSGBL   5         0      Page Table
Total (40.04 Mbytes reserved)  5125     0
```

See Also:

- [Section , "Reserved Memory Registry"](#)
- Hewlett-Packard documentation for more information about reserving memory

Chip Set

Hp OpenVMS AlphaServer systems must have an EV56 or later generation CPU. Refer to [Section , "Changes and Enhancements"](#) for more information.

DVD Drive

A DVD drive that is supported by OpenVMS is required. Oracle uses ODS-5 format installation media.

Images

Three images are installed (using the install utility) by default by Oracle Database Enterprise Edition.

LIBCLNTSH.SO is linked with Oracle Database Client images, reduces their size by removing direct references to Oracle Net Services and other common routines.

ORACLE.EXE is the Oracle server image and will not be present in a Client-only installation.

Disk Space Requirements

Oracle Universal Installer enables you to select an installation category and type. Your choices determine how much disk space is required. The disk space requirements do not include the size of the database. A production Oracle Database that supports many users requires significant disk space and memory.

Table 2–1 lists the disk space requirements for Oracle Database.

Table 2–1 Disk Space Requirements for Oracle Database Server

Installation Type	Required Disk Space
Enterprise Edition	10 GB
Custom	Depends on the components selected, but a minimum of 5 GB
Staging area (copying dvd contents to disk)	5 GB

Table 2–2 lists the disk space requirements for Oracle Database Client.

Table 2–2 Disk Space Requirements for Oracle Database Client

Installation Type	Required Disk Space
Administrator	4.5 GB
Run time	4.5 GB
Custom	9 GB

Oracle Advanced Security

Oracle Advanced Security is an add-on product that is added to the standard Oracle Net Services Server or Oracle Net Services Client. If you purchase it, then install it on both the server and the client.

The SSL version compatibility is SSL3.0 or later. Kerberos and Secure Sockets Layer (SSL) have requirements for authentication protocols that are supported by Oracle Advanced Security. No additional authentication protocol software is required to relink Oracle products. The requirements for Kerberos and SSL are as follows:

- Kerberos
Requires MIT Kerberos version 5 release 1.1. The Kerberos authentication server must be installed on a physically secure system.
- SSL
Requires a wallet that is compatible with the Oracle Wallet Manager version 10.2. Wallets that were created by using earlier releases of the Oracle Wallet Manager are not upward-compatible. Oracle Advanced Security provides and installs SSL.

See Also: For more information about Oracle Advanced Security and system management products, refer to *Oracle Database Advanced Security Administrator's Guide*

Note: No additional authentication protocol software is required to relink Oracle products. However, Oracle does not provide third-party authentication servers (Kerberos, for example). The appropriate authentication server for these protocols must be installed and configured separately.

SSL is provided and is always installed with Oracle Advanced Security.

Checking Network Requirements for Oracle RAC

Check that you have the networking hardware and IP addresses required for a Oracle RAC installation.

See Also: For the latest information about supported network protocols and hardware for Oracle RAC installations, refer to the Certify screens on the Oracle *MetaLink* Web site at

<https://metalink.oracle.com>

This section contains the following topics:

- [Section , "Network Hardware Requirements"](#)
- [Section , "IP Address Requirements"](#)
- [Section , "Checking Each Node for Network Requirements"](#)

Network Hardware Requirements

Each node in the cluster must meet the following requirements:

- The node must have at least two network adapters, one for the public network interface and one for the private network interface (the interconnect).
- The interface names that are associated with the network adapters for each network must be the same on all nodes.
- For increased reliability, you can configure redundant public and private network adapters for each node.
- Each network adapter must support TCP/IP.

The private network between the nodes in the clusters is expected to act as a high-bandwidth network for communication between inter-Oracle RAC instances.

The public network is expected to provide connections between Oracle Database clients and Oracle RAC instances.

IP Address Requirements

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

- An IP address and an associated host name registered in the domain name service (DNS) for each public network interface
- One private IP address for each node to serve as the private interconnect

Oracle recommends that you use nonroutable IP addresses for the private interfaces, for example, 10.*.* or 192.168.*.*. You can use the local TCP/IP database on each node to associate private host names with private IP addresses.

This IP address must be separate from the public network, and it must have the same interface name on every node that is part of the cluster.

During a Oracle RAC installation, Oracle Universal Installer displays two screens on which you specify the IP addresses of the private interconnect. The private interconnect is used for internode communication by both Oracle Clusterware and Oracle RAC.

On the Cluster Configuration screen, you must specify a private node name to be associated with each public node name. The public node name is the host name of

each node, which is specified as either an alias or an IP address. The private node name is the private interconnect address that will be used by Oracle Clusterware. In the Private Node Name field, enter the name of the private interconnect if it is available from a network name server or system hosts file, or enter the private IP address, unique to each node.

The information about the Private Interconnect Enforcement screen is used to determine which private interconnect will be used by Oracle RAC database instances. Oracle RAC uses all interconnects identified as private on this screen. These interconnects must be in the running state, as in the case if their IP addresses were specified in the `CLUSTER_INTERCONNECTS` logical. Oracle RAC does not fail over between cluster interconnects. So, if one cluster is down, then the instances using it will not start.

- One public IP address for each node to serve as the virtual IP address for client connections and for connection failover

This is in addition to the public host IP address that is managed and assigned to the node by the operating system. This public virtual IP must be associated with the same interface name on every node that is part of the cluster. In addition, the IP addresses that you use for all the nodes that are part of a cluster must be from the same subnet. The host names for the virtual IP must be registered with the domain name server (DNS).

Note: The virtual IP address must not be in use at the time of the installation because this is a virtual IP address that Oracle Database manages.

The reason the virtual IP address must be unused is that when VIPCA tries to register the vip on the system, it first pings if the IP is alive and, if so, assumes that VIPCA has already been run and fails to complete. After installation, you can configure clients to use the virtual host name or IP address. If a node fails, then its virtual IP address fails over to another node.

For example, if each node has two public and two private interfaces, you may have the host names and IP addresses on one of the nodes (`rac1`) and similar host names and IP addresses on the other nodes as shown in [Table 2-3](#).

Table 2-3 Sample IP Addresses for Each Node

Host Name	Type	IP Address	Registered In
<code>rac1.mydomain.com</code>	Public	143.47.43.100	DNS
<code>rac1-2.mydomain.com</code>	Public	143.46.51.101	DNS
<code>rac1-vip.mydomain.com</code>	Virtual	143.47.43.104	DNS
<code>rac1-priv1</code>	Private	10.0.0.1	A local TCP/IP database
<code>rac1-priv2</code>	Private	10.0.0.2	A local TCP/IP database

Checking Each Node for Network Requirements

To verify that each node meets the network requirements:

1. If necessary, install the network adapters for the public and private networks, and configure them with either public or private IP addresses.

2. Register the host names and IP addresses for the public network interfaces in DNS.
3. For each node, register one virtual host name and IP address in DNS.
4. For each private interface on every node, include the following in the TCP/IP configuration on all nodes, specifying the private IP address and associated private host name as follows:

```
10.0.0.1    rac1-priv1
```

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

```
@SYS$MANAGER:TCPIP$DEFINE_COMMANDS
```

Then run:

```
$ IFCONFIG -A
```

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

Checking Software Requirements

This section provides information about checking the software requirements.

Checking for Required Software

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

Item	Requirement
Operating system	<p>For OpenVMS Alpha: OpenVMS version 8.2</p> <p>For OpenVMS Itanium: OpenVMS version 8.2-1</p>
Oracle Universal Installer	X Window and X/Motif software:
Oracle Messaging Gateways	<p>Download the SupportPac MA88 from http://www-1.ibm.com/support/docview.wss?rs=203&uid=swg24000173&loc=en_US&cs=utf-8&lang=en</p>
Pro*C/C++, Oracle XML Developer's Kit (XDK)	<p>For OpenVMS Alpha:</p> <ul style="list-style-type: none"> ■ Compaq C 6.5 ■ Compaq C ++ 6.5 <p>For OpenVMS Itanium:</p> <ul style="list-style-type: none"> ■ HP C S7.1 ■ HP C++ 7.1

Item	Requirement
Oracle Net Services Requirements	<p>Oracle Net Services on OpenVMS is developed and certified using TCP/IP Services for OpenVMS (UCX). If you wish to use the TCP/IP protocol adapter for Oracle Net Services, then you should have Version 5.5 ECO 1 or later of TCP/IP Services for OpenVMS installed.</p> <p>Caution: TCP/IP protocol stacks from other vendors may work with Oracle, but this is not guaranteed by Oracle. Any TCP/IP problems that cannot be reproduced using TCP/IP Services for OpenVMS should be referred to the TCP/IP vendor.</p> <p>Vendor-provided protocol services are usually upward-compatible, so existing applications will continue to work without modification. Therefore, later releases of TCP/IP are upward-compatible with Oracle Net Services, as long as the vendor-specified application programming interface (API) does not change with new releases.</p>
Oracle JDBC/OCI Drivers	<p>You can use the following JDK versions with the Oracle JDBC/OCI drivers:</p> <p>For OpenVMS Alpha:</p> <ul style="list-style-type: none"> ■ HP JDK v 1.3.1 ■ HP JDK v 1.2 ■ HP JDK 1.4.2 <p>However, they are not required for the installation.</p> <p>For OpenVMS Itanium:</p> <ul style="list-style-type: none"> ■ HP JDK 1.4.2 <p>Note: HP JDK v 1.4.2 is installed as part of a standard Oracle software installation.</p>
Pro*COBOL	<p>For OpenVMS Alpha: Compaq COBOL 2.8</p> <p>For OpenVMS Itanium: HP COBOL 2.8</p>
Pro*FORTRAN	<p>For OpenVMS Alpha: Compaq FORTRAN release 7.5-2630-48C8L</p> <p>For OpenVMS Itanium: HP FORTRAN release 8.0-48071-50EAE</p>

To ensure that the system meets these requirements:

1. To determine which version of OpenVMS is installed, enter the following command:

For OpenVMS Alpha:

```
$ WRITE SYS$OUTPUT F$GETSYI("VERSION")
V8.2
```

For OpenVMS Itanium:

```
$ WRITE SYS$OUTPUT F$GETSYI("VERSION")
V8.2-1
```

If necessary, refer to the operating system documentation for information about upgrading the operating system.

-
2. To display additional information about the operating system, enter the following command:

```
$ SHOW SYSTEM /NOPROCESS /FULL
```

To view information about all nodes in a cluster, add the /CLUSTER qualifier to this command.

3. If you intend to use Oracle Messaging Gateway and require MQSeries classes for Java and MQSeries classes for Java Message Service (SupportPac MA88), download it from the following IBM Web page:

<http://www-1.ibm.com/support/docview.wss?rs=203&uid=swg24000173>

4. If you require a Corrective Services Deliverable (CSD) for MQSeries, refer to the following Web site for download and installation information:

<http://www.ibm.com/software/integration/mqfamily/support/summary/dig.html>

5. To determine the version of TCP/IP installed, enter the following command:

For OpenVMS Alpha:

```
$ tcpip show version
HP TCP/IP Services for OpenVMS Alpha Version V5.5 - ECO 1
on a AlphaServer GS160 6/731 running OpenVMS V8.2
```

For OpenVMS Itanium:

```
$ tcpip show version
HP TCP/IP Services for OpenVMS Industry Standard 64
Version V5.5 - ECO 1
on an HP rx4640 (1.50GHz/6.0MB) running OpenVMS V8.2-1
```

Oracle Net Services OpenVMS Mailbox Driver

The Oracle Net Services OpenVMS Mailbox driver (protocol IPC) is included in NETWORK. You do not need an Oracle Net Services license to use the OpenVMS Mailbox driver.

Checking for Required Patches

You must ensure that the patches described in the following table are installed on the system.

Installation Type or Product	Minimum Requirement
All installations	<p>The following patches, or their later versions, are required:</p> <p>For OpenVMS Alpha:</p> <ul style="list-style-type: none"> ▪ DEC AXPVMS VMS82A_ACRTL V1.0 ▪ DEC AXPVMS VMS82A_BASRTL V1.0 ▪ DEC AXPVMS VMS82A_FIBRE_SCSI V2.0 ▪ DEC AXPVMS VMS82A_KITTING V1.0 ▪ DEC AXPVMS VMS82A_MONTOR V2.0 ▪ DEC AXPVMS VMS82A_PTHREAD V1.0 ▪ DEC AXPVMS VMS82A_SYS V3.0 ▪ HP VMS AVAIL_MAN_COL V2.5-BIAV ▪ HP VMS AVAIL_MAN_COL V2.5-AIAV ▪ DEC AXPVMS VMS82A_F11X V2.0 ▪ DEC AXPVMS VMS82A_INSTAL V1.0 ▪ DEC AXPVMS VMS82A_LMF V1.0 ▪ DEC AXPVMS VMS82A_LOADSS V1.0 ▪ DEC AXPVMS VMS82A_TDF V1.0 ▪ DEC AXPVMS VMS82A_PCSI V1.0 ▪ DEC AXPVMS VMS82A_UPDATE V2.0 ▪ HP VMS AVAIL_MAN_COL V2.5-AIAV ▪ HP AXPVMS AVAIL_MAN_ANA V2.5-A ▪ HP VMS AVAIL_MAN_COL V2.5-AIAV ▪ HP VMS AVAIL_MAN_COL V2.4-1 ▪ HP AXPVMS AVAIL_MAN_ANA V2.5-A ▪ HP AXPVMS AVAIL_MAN_ANA V2.4-1 ▪ DEC AXPVMS JAVA142 V1.4-24P5 ▪ DEC AXPVMS TCPIP V5.5-11ECO1 ▪ IBM AXPVMS WMQCLIENT V5.30 ▪ DEC AXPVMS FASTVM122 V1.2-21 ▪ HP VMS AVAIL_MAN_COL V2.4-1 ▪ HP AXPVMS AVAIL_MAN_ANA V2.4-1 ▪ CPQ AXPVMS CDSA V2.1-331 ▪ DEC AXPVMS DWMOTIF V1.5 ▪ DEC AXPVMS OPENVMS V8.2 ▪ HP AXPVMS KERBEROS V2.1-72

Installation Type or Product	Minimum Requirement
All installations	<p>For OpenVMS Itanium:</p> <ul style="list-style-type: none"> ■ VMS821I_BASRTL V1.0 ■ VMS821I_DEBUG V2.0 ■ VMS821I_FIBRE_SCSI V2.0 ■ VMS821I_LOADSS V2.0 ■ VMS821I_MONTOR V1.0 ■ VMS821I_PTHREAD V2.0 ■ VMS821I_SYS V2.0 ■ HP VMS AVAIL_MAN_COL V2.5-BIAV ■ HP VMS AVAIL_MAN_COL V2.5-AIAV ■ SEA V4.4-4 ■ WEBES V4.4-4 ■ WCCPROXY V1.3-4 ■ VMS821I_F11X V2.0 ■ VMS821I_INSTAL V1.0 ■ VMS821I_LMF V1.0 ■ VMS821I_LOADSS V1.0 ■ VMS821I_UPDATE V2.0 ■ VMS821I_TDF V2.0 ■ VMS821I_ACRTL V1.0 ■ VMS821I_UPDATE V1.0 ■ VMS821I_PCSI V1.0 ■ JAVA142 V1.4-23 ■ AVAIL_MAN_ANA V2.5-I ■ TCPIP V5.5-11ECO1
Oracle Messaging Gateway	Corrective service diskette (CSD) for MQSeries: CSD09 or later for MQSeries V5.3

In some cases, later patch sets would cumulatively include earlier patches and therefore, the preceding command would not explicitly indicate the required patch sets. In such cases, check the Patch Distribution Web site of Hewlett-Packard for the contents of the patches that are displayed to verify if the required patches are included or not.

To check if you have the required patches installed on the system, enter the following command:

```
$ PRODUCT SHOW HISTORY
```

The output of this command is a list of all software patches or products (or both) that are installed on the system. Verify that you meet the minimum requirements to install and run Oracle Database.

Installation-Specific Issues and Restrictions

There are factors that can affect the installation or use of Oracle Database. This section discusses these factors:

- [Section , "New ORA_ROOT"](#)
- [Section , "JDK"](#)
- [Section , "Devices"](#)

New ORA_ROOT

Do not install Oracle Database into the root directory of an existing Oracle installation containing any Oracle Software. Oracle recommends that you install Oracle Database products into a new ORA_ROOT.

Logical Names

Oracle Database 10g release 2 has two separate logical names that specify the top-level directory of the installation. ORA_ROOT is a concealed logical name used in earlier releases, and ORACLE_HOME is a nonconcealed logical name. You can use both logical names with the following restriction.

If the top-level directory is DISK\$DISK1 : [ORACLE10102] , then the logical names are defined as follows:

```
$ SHOW LOGICAL ORA_ROOT
DISK$DISK1 : [ORACLE10102.]
```

```
$ show logical ORACLE_HOME
DISK$DISK1 : [ORACLE10102]
```

Therefore, the following command is invalid:

```
$ SET DEFAULT ORA_ROOT
```

The following commands are valid:

```
$ SET DEFAULT ORA_ROOT : [000000]
$ SET DEFAULT ORACLE_HOME
```

JDK

The JDK release that is shipped with Oracle Database is used by Oracle Java applications such as Oracle Universal Installer and is the only JDK that is supported to run with these applications. You must not modify this JDK, unless it is done through a patch provided by Oracle Support Services.

Devices

Oracle Database software must be installed on an On-Disk Structure Level 5 (ODS-5) formatted disk. The data files and redo log files, however, can be placed on ODS-2 formatted disks as well.

Oracle Database server must be able to verify that files have been written to disk. Devices that do not support this verification are not supported for use with Oracle Databases, although Oracle software can be installed on them. Datafiles always must reside on volumes with write through cache enabled

Note: Because of this requirement for verification, third-party software that creates one or more virtual disks in memory must not be used for data files.

Creating a User Account

This section provides installation procedure requirements. It discusses the following topics:

- [Section , "Creating an Oracle Database DBA Account"](#)
- [Section , "Adding a Record in the User Authorization File"](#)
- [Section , "Setting Account Privileges"](#)
- [Section , "Setting Account Quotas"](#)
- [Section , "Process Rights for Database Administrators"](#)
- [Section , "Process Rights for SGA Protection"](#)
- [Section , "Security Issues with Multiple Databases"](#)
- [Section , "Completing Account Setup"](#)
- [Section , "Verifying Privileges"](#)
- [Section , "Setting Permissions for File Creation"](#)

Creating an Oracle Database DBA Account

Create an OpenVMS user account to administer and maintain the Oracle installation. Setting up an Oracle Database account is the same as setting up any other OpenVMS user account.

To set up an Oracle Database account:

1. Determine the values that you want to supply to create the account.
2. Run the `AUTHORIZE` utility to add a record in the User Authorization File (UAF).
3. Use `AUTHORIZE` to set the following account requirements:
 - Account quotas
 - Account privileges
 - Process rights identifiers
4. Exit `AUTHORIZE` to complete the account setup.

See Also: The Hewlett-Packard documentation for more information about running the `AUTHORIZE` utility

Determining Account Information

To add an account, you must supply the following information:

- User name
- User password
- `USER` and `GROUP` numbers for the UIC (octal values)
- Login device and directory

- Owner

Although this guide refers to this account as the Oracle Database account or Oracle account, you can assign any name or number to the account after taking into account UIC restrictions. The Oracle Database account owns the run-time libraries and executable images for every Oracle product. Therefore, the database administrator (DBA) must manage this account and install all Oracle products from it.

Note: In *Oracle Database Administrator's Guide*, the Oracle Database account is sometimes referred to as the DBA account. The Oracle Database account is not the same as the SYS or SYSTEM database user names that are created for every database. It is an OpenVMS account name.

The UIC group number of the Oracle Database account must be greater than the system parameter MAXSYSGROUP (which defaults to octal 10). Group numbers 1 through the value of MAXSYSGROUP are reserved for use by the OpenVMS operating system.

If the UIC group number is not greater than the value of MAXSYSGROUP, then all Oracle Database account commands are not supported. For example:

- STARTUP
- CREATE DATABASE or CREATE TABLESPACE
- ALTER DATABASE or ALTER TABLESPACE

The following command gives the value of MAXSYSGROUP (in decimal format):

```
$ WRITE SYS$OUTPUT F$GETSYI("MAXSYSGROUP")
```

Adding a Record in the User Authorization File

Use the AUTHORIZE utility to create or modify records in the UAF as follows:

1. To run the AUTHORIZE utility, enter the following commands:

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
```

2. At the UAF prompt, enter the ADD command to create the user account from which you will install the product software:

```
UAF> ADD /ACCOUNT=ORACLE10G /PASSWORD=ORACLE/UIC=[277,100]-
/DEVICE=device/DIRECTORY=[ORACLE10G]/OWNER="ORACLE DBA"
```

In this example, the account name is ORACLE10g. Note that the UIC group number is 277. The UIC group number must be greater than MAXSYSGROUP.

After adding the account, you must alter the account privileges and quotas. This can be done in any order. However, in most cases, the account privileges are set before the account quotas.

See Also: The Hewlett-Packard documentation for more information about using the AUTHORIZE utility

Setting Account Privileges

The following privileges are required as both authorized and default privileges for the Oracle Database DBA account:

Note: An exception is explained in [Section , "Security Issues with Multiple Databases"](#).

- CMKRNL
- GROUP
- GRPNAM
- IMPERSONATE
- LOG_IO
- NETMBX
- OPER
- PFNMAP
- PRMGBL
- PRMMBX
- SHARE
- SYSGBL
- SYSLCK
- SYSNAM
- SYSPRV
- TMPMBX
- WORLD

Use the AUTHORIZE utility to set the account privileges for the Oracle Database DBA account. At the UAF prompt, use the MODIFY command to add the required default and authorized privileges as follows:

```
UAF> MODIFY ORACLE10G -  
/PRIVILEGE= (CMKRNL, NETMBX, OPER, PFNMAP, PRMGBL, PRMMBX, SHARE, SYSGBL, -  
  SYSNAM, TMPMBX, GROUP, GRPNAM, IMPERSONATE, LOG_IO, WORLD, SYSLCK, SYSPRV) -  
/DEFPRIVILEGE= (CMKRNL, NETMBX, OPER, PFNMAP, PRMGBL, PRMMBX, SHARE, SYSGBL, -  
  SYSNAM, TMPMBX, GROUP, GRPNAM, IMPERSONATE, LOG_IO, WORLD, SYSLCK, SYSPRV)
```

Setting Account Quotas

Use the AUTHORIZE utility to change account quotas to meet the requirements of Oracle Database installation.

Account Quotas

[Table 2–4](#) lists the Oracle Database account quotas, their minimum recommended values, and their equivalent OpenVMS quota names as displayed by the SHOW PROCESS/QUOTA and SHOW WORKING_SET DCL commands.

Note: These quotas depend on the number of log files, databases, network connections, and other logicals on the system. You may need to customize these values.

Table 2–4 Oracle Database Account Quotas

Account Quota	Minimum Value	Quota Name
ASTLM	250 (the default)	Asynchronous system trap limit
BYTLM	750000	Buffered I/O limit
ENQLM	2000 (the default)	Enqueue quota
FILLM	2048	Open file quota
JTQUOTA	8192	Job table quota
MAXDETACH	0 (the default)	Max detached processes
MAXJOBS	0 (the default)	Max active jobs
PGFLQUOTA	2 million (recommended)	Paging file quota
WSDEFAULT	2048	Working set default
WSEXTENT	8192	Working set extent
WSQUOTA	4096	Working set quota

Modifying the Default Quotas

After adding a record with the default quotas, use the `MODIFY` command to alter the default values. The following example changes the enqueue quota (`ENQLM`) from the default value to a value of 150:

```
UAF> MODIFY ORACLE10G/ENQLM=150
```

Process Rights for Database Administrators

The Oracle Database DBA account must be granted one or more process rights identifiers. These identifiers enable you to run the `CONNECT /AS SYSDBA` command that is required to perform database administration functions. [Table 2–5](#) displays the combinations of adding and granting rights identifiers to this account:

Table 2–5 Adding and Granting Process Rights Identifiers

Adding . . .	And Granting . . .	Enables Control of . . .
<code>ORA_DBA</code>	<code>ORA_DBA</code>	Any database instance, provided it does not also have an <code>ORA_sid_DBA</code> identifier defined for it.
<code>ORA_sid_DBA</code>	<code>ORA_sid_DBA</code>	Database instance <code>sid</code> only
<code>ORA_sid_DBA</code>	<code>ORA_DBA</code>	All database instances except <code>sid</code>

For example, to grant the `ORA_sid_DBA` rights identifier to the Oracle Database account for an instance called `TEST`, enter the following command:

```
UAF> ADD/IDENTIFIER ORA_TEST_DBA
```

Then, grant the rights identifier to the Oracle Database account as follows:

```
UAF> GRANT/IDENTIFIER ORA_TEST_DBA ORACLE10g
```

If you add the `ORA_TEST_DBA` rights identifier, but grant only `ORA_DBA` to the `ORACLE10g` account, then the account would have insufficient privileges to administer the `TEST` instance.

The third row of the table shows that you can add and grant different identifiers, which restricts control of a particular instance while it grants control to other instances.

You can also grant database maintenance privileges to accounts (for example, privileges to start up and shut down the database) other than the Oracle Database account. However, with the Oracle Database account, the user UIC group number must be greater than `MAXSYSGROUP`. By granting the `ORA_sid_DBA` identifier, you can similarly restrict user privileges to an instance named `sid`.

Process Rights for SGA Protection

The SGA is protected by an access control list (ACL) that grants access to the identifier `ORA_SGA`. Before you start a database, add this rights identifier to the UAF, if it does not already exist. To do this, the system manager must enter the following command

```
UAF > ADD/IDENTIFIER/ATTRIBUTES=SUBSYSTEM ORA_SGA
```

Caution: Do not grant the `ORA_SGA` rights identifier to any user. Oracle software manages the assignment of the rights identifier when it starts the database.

Security Issues with Multiple Databases

If a site has several databases managed by different DBAs, then you may not want to grant the `CMKRNL` privilege to every DBA Oracle Database account. This privilege enables a DBA to activate any process rights identifiers and to install shared images.

For security reasons, Oracle recommends that the OpenVMS system administrator handle these tasks. This person should be responsible for starting Oracle Database instances from a controlled account with a UIC group number that is greater than the value of `MAXSYSGROUP`. If this is not possible, then at least one Oracle Database account must have the `CMKRNL` privilege.

Completing Account Setup

Exit the `AUTHORIZE` utility. If the user account that you modified is logged on at the time you exit this utility, then this user must log out and log in again before the changes take effect.

Verifying Privileges

Enter the following command to verify that your account has the correct privileges and rights identifiers:

```
$ SHOW PROCESS/PRIVILEGE
```

Setting Permissions for File Creation

Set default file protections for the `ORACLE10g` account to ensure that `GROUP` and `WORLD` have `READ` and `EXECUTE` permissions, but not `WRITE` permission, on installed files.

1. Enter the `$ SHOW PROTECTION` command to check the current setting.
2. If the `$ SHOW PROTECTION` command does not show the expected protection, then set protection in the `LOGIN.COM` of the `oracle10g` account as follows:

```
$ SET PROTECTION=(s:RWE, o:RWED, g:RE, w:RE)/default
```

SYSGEN Parameters

See Also: `READMEVMS.DOC` in the `RDBMS` directory for instructions on calculating `MIN_GBLPAGES`, and `MIN_GBLSECTIONS`

You can create and use reserved memory on an OpenVMS system. The following sections describe the utilities that you can use to set the values of the various SYSGEN parameters:

- [Section , "Reserved Memory Registry"](#)
- [Section , "AUTOGEN"](#)

Reserved Memory Registry

Through its interface within the `SYSMAN` utility, the reserved memory registry enables you to configure an OpenVMS system with large amounts of memory set aside for use within memory-resident global sections. The `AUTOGEN` utility considers the preallocated reserved memory when it tunes the system.

Oracle recommends that you do *not* use reserved memory until the instance SGA size is stable. In most cases, this is a few weeks after the database is performing well.

The advantages to reserving memory for an SGA are as follows:

- You can be certain that the memory is available and that the system is correctly tuned.
- The memory is reserved at startup time as contiguous aligned physical pages. This enables the system to optimize the mapping of the SGA.
- The memory is preallocated and zeroed. This results in faster SGA mapping and faster instance startup.

To reserve memory for an SGA, use the `SYSMAN` utility. The size qualifier is specified in megabytes. For example, to reserve memory for a 6 GB SGA for the `SID` named `TEST`, use the following commands:

```
$ MCR SYSMAN
SYSMAN> RESERVED_MEMORY ADD ORA_TEST_SGA/SIZE=6144/ALLOCATE/ZERO/PAGE
SYSMAN> EXIT
```

Next, run `AUTOGEN` and restart the system to allow `AUTOGEN` to adjust other system parameters for the reduced amount of memory that is available to the rest of the system.

Memory for multiple SGAs may be reserved. Any change to the name or size of a piece of reserved memory may require restarting the system. During instance startup, Oracle Database compares the size of the reserved memory, if any, with the size of the SGA. If appropriate, one of the following messages is included in the alert log:

```
** Reserved memory size = size greater than created SGA size = size**
** Please reduce reserved memory size to avoid wasting memory. **
```

```
** Memory was not reserved for the SGA. SGA size = size **
** There might be performance advantages to allocating memory for
    the SGA in the VMS reserved memory registry. **
```

If the memory reserved for an SGA is insufficient, then the instance startup fails. In this case, you must adjust the amount of reserved memory, run `AUTOGEN`, and restart the system. For example, to expand the SGA to 6.5 GB for the `TEST` instance, enter the following commands:

```
$ MCR SYSMAN
SYSMAN> RESERVED_MEMORY REMOVE ORA_TEST_SGA
SYSMAN> RESERVED_MEMORY ADD ORA_TEST_SGA/SIZE=6656/ALLOCATE/ZERO/PAGE
SYSMAN> EXIT
```

Next, run `AUTOGEN` and save it permanently using the `SAVEPARAMS` parameter. This saves the change in the `AGEN$FEEDBACK.DAT` file.

If you do not wish to save permanently then, instead of running `AUTOGEN`, use `SYSMAN` to free the memory reservation and proceed with starting the Oracle instance without reserved memory. The risk involved with this is that the system may not have enough fluid pages to create the SGA. In addition, the performance advantages of using reserved memory are not available.

See Also: The Hewlett-Packard documentation for more information about the reserved memory registry

To review memory reservations on the node, run the following command:

```
$ SHOW MEMORY/RESERVED

System Memory Resources on 13-JUL-2001 09:57:11.72
Memory Reservations (Pages)   Group   Reserved   In Use   Type
ORA_PROD_SGA                  SYSGBL   5120        0       Allocated
ORA_PROD_SGA                  SYSGBL    5           0       Page Table
Total (40.04 Mbytes reserved) 5120      0
```

AUTOGEN

Because `SYSGEN` parameters affect the entire operating system, the system administrator is the only person who should modify them. At some sites, the DBA and system administrator may be the same person. The instructions in this section are directed to the system administrator.

For setting or modifying `SYSGEN` parameters, OpenVMS provides the `AUTOGEN` utility. You can also use `SYSGEN`, but this is an older utility and its use is discouraged, except perhaps for checking current values. You must have the `SYSPRV` or `BYPASS` privilege to run these utilities.

`AUTOGEN` provides a permanent method for setting parameters, and it documents all changes. `AUTOGEN` also lets you recalculate any parameters that depend on other parameters that you may have changed. Remember to record parameter values before changing them, and determine in advance what results you expect from the changes you make. If the expected changes do not occur, then restore the old values before trying again.

The procedure is as follows:

1. Run the `AUTOGEN` utility with the `SAVPARAMS` parameter by entering the following command:

```
$ @SYS$UPDATE:AUTOGEN SAVPARAMS GETDATA
```

This step saves current parameters to a file named `AGEN$FEEDBACK.DAT`.

2. If you have not already done so, run `AUTOGEN GETDATA` to ensure that the file `PARAMS.DAT` has been generated.

```
$ @SYS$UPDATE:AUTOGEN GETDATA
```

3. Examine the parameter settings in the current `PARAMS.DAT`, and if they are correct, then change them in the `MODPARAMS.DAT` file by using the `MIN_parameter=value` format as follows:

```
MIN_GBLSECTIONS=3000
MIN_GBLPAGES=300000
```

The `MIN` prefix indicates that you are setting a lower limit for the specified parameter. To use the `MIN` prefix, identify the current value of the specified parameter (using `SYSGEN`) and the amount by which you want to increase the parameter. The sum of these two values is used for the `MIN_parameter` entry.

4. Run the `AUTOGEN` utility with the `GENPARAMS` and `REBOOT` parameters by entering the following command:

```
$ @SYS$UPDATE:AUTOGEN GENPARAMS REBOOT
```

Caution: The `REBOOT` parameter in this example causes the system to automatically restart when the `SYSGEN` is complete.

This step generates new `SYSGEN` parameters in a file named `SETPARAMS.DAT` and runs the `SYSGEN` utility to set these parameter values as specified in the file.

See Also: *OpenVMS System Management Utilities Reference* for more information about using the `AUTOGEN` utility

System Configuration

If the system fails to satisfy any listed requirement, then perform the tasks listed in the next section as necessary to configure the system to meet these requirements.

[Table 2–6](#) lists the requirements for installing Oracle Database.

Table 2–6 *OpenVMS System Configuration Summary*

System Factors	Requirements for Oracle Database
OpenVMS Parameters: Minimum Global Pages or Sections	Verify that the values for <code>MIN_GBLPAGES</code> and <code>MIN_GBLSECTIONS</code> are greater than, or equal to, the values that were calculated for these parameters, as documented in <code>READMEVMS.DOC</code> .
OpenVMS Accounts	You need an OpenVMS account that is dedicated solely to installing and upgrading Oracle products. The instructions in this book are provided with the assumption that the account is called <code>oracle10g</code> .
Permissions for File Creation	Default Protections

Performing Setup Tasks As the SYSTEM User

Log in as the `SYSTEM` user and perform the following tasks to set up the environment for Oracle Database:

- [Section , "Creating an Oracle Database Account"](#)
- [Section , "Creating an Apache Server Account"](#)

Creating an Oracle Database Account

The Oracle Database account is the OpenVMS user account that owns Oracle Database software after installation. Run Oracle Universal Installer with this user account.

Details of creating this account are covered in [Section , "Creating a User Account"](#)

Use the `AUTHORIZE` utility to create an `oracle` account with the properties listed in [Table 2-7](#).

Table 2-7 Properties of the Oracle Database Account

Account	Properties
Login Name	Any name, but this document refers to it as the Oracle Database account.
UIC	The group UIC number, which must be greater than <code>MAXSYSGROUP</code> .
Home Directory	Select a home directory consistent with other user home directories. The home directory of the <code>ORACLE10g</code> account does not need to be the same as the <code>ORA_ROOT</code> directory.

Caution: Use the Oracle Database account only for installing, maintaining and managing Oracle software. Do not use it for purposes unrelated to Oracle Database server. Do not use `SYSTEM` as the `ORACLE10g` account.

Sites with multiple `ORA_ROOT` directories on one system may install Oracle software with the same `ORACLE10g` account, or separate accounts. Each `ORACLE10g` account must have the same group UIC.

Creating an Apache Server Account

The Apache account is an OpenVMS user account that owns the Apache server after installation. If you use a default Apache configuration (one that listens to ports lower than 1024, which are reserved for the system), then Oracle recommends, for security reasons, that a separate account owner be set up for Apache. Configure the Apache server to assign the ownership of listener and module actions to this account. It is not necessary for the Apache user account to have privileges to run the Apache Server on port numbers that are lower than 1023. Images that are installed with privileges during installation in the `oracle` account ensure that this can be done.

For information about how to configure the server to run as a user other than the 'oracle' user, refer to [Section , "Customizing the Server Environment"](#).

The Apache account owner should have minimal user privileges, and should not be a member of any groups where files are not intended to be visible to the public.

Performing Setup Tasks As the oracle user

Log in to the oracle Database account and perform the following tasks as necessary:

- [Section , "Setting the Display"](#)
- [Section , "Setting Up a Batch Queue for the EMAgent and HAS"](#)

Setting the Display

Before starting Oracle Universal Installer, set the display by using the following command:

```
$ SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```

Verify that you have set the display correctly by using the following command:

```
$ RUN SYS$SYSTEM:DECW$CLOCK
```

Do not use the host name or IP address of the system where you install the software unless you perform the installation from the X Window console of that system. Use the computer name or IP address of the workstation if you are installing from a remote system. If you are not sure of the X server and screen settings, then use 0 (zero) for both.

Setting Up a Batch Queue for the EMAgent and HAS

In this release, the architecture and implementation of the Enterprise Manager Agent has undergone a major revision. The new agent is based on an HTTP listener that communicates with the Management Server using XML and that uses Perl to perform a number of tasks. This has been implemented to run in a batch queue instead of running as detached processes. The Enterprise Manager Agent runs a main process. To support monitoring of both internal (using the same Oracle home) and external databases and listeners, the Enterprise Manager Agent submits additional batch jobs.

From the operational and management perspective, this has been implemented to run under a batch queue to get a better collective view of all processes related to the Enterprise Manager Agent running at any given time. This makes it easy to monitor and control all processes spawned by the Enterprise Manager Agent. In addition, it may be possible to delete batch jobs or processes that can potentially fail when trying to connect to databases or listeners that are not running.

Note: Although the Enterprise Manager Agent is no longer installed as part of a database installation, it is available as part of the separate Enterprise Manager Grid Control release.

For High-Availability Services (HAS), the Oracle Clusterware daemons, OCSSD, EVMD, and CRSD, are collectively run as batch process. The EMAgent and HAS use the following method to determine the name of the batch queue on which to submit jobs:

- If the `ORA_BATCH_QUEUE_hostname` logical name is defined, then the value of this logical is used.
- If the `ORA_BATCH_QUEUE_hostname` logical is itself a valid queue name, then it is used.

The `ORA_BATCH_QUEUE_hostname` logical name need not always be defined. It is sufficient that it is defined at least once during installation. Alternatively, you can

define it before running `ORAUSER.COM` at any time after the installation. Note that the Enterprise Manager Agent and HAS services can only be started up after setting up this batch queue logical name and running `ORAUSER.COM`.

The two components record the batch queue name in the installation, and use that as a reference for subsequent calls to the services. The default batch queue `SYS$BATCH` generally points to a batch queue that runs under a lower priority (usually, 2) than the priority of a normal user (usually, 4). Therefore, it is inappropriate to use the default queue for the two services. This would result in poor and, in some cases, defective performance of the two services.

Note: Oracle recommends to use a dedicated batchqueue.

The number of jobs submitted for the EMAgent batch queue depends on the number of monitoring targets which include databases and listeners. An approximation of two to five monitoring targets would be a good number for the `job_limit`. In the case of HAS, a `job_limit` of about 10 would be sufficient. The batch queue priority should be the same as all other Oracle processes, which is typically 4 on a standard VMS system.

Setup Tasks for Oracle Products

The following products require preinstallation steps to be completed before you install Oracle Database software:

- [Section , "Precompilers and Tools"](#)
- [Section , "Network and System Management Products"](#)

Precompilers and Tools

The following sections list the preinstallation tasks for the precompilers and tools that are required for OpenVMS.

Network and System Management Products

This section describes tasks that need to be completed prior to installation if you have existing network and system management products.

Setup Tasks for Oracle Net Services

If you have an existing installation of Oracle Net Services running on the same TCP ports, then shut down all listeners before installing it. To determine if the TCP port is in use, enter the following command:

```
$ TCPIP SHOW DEVICE/PORT=port_number
```

If the port is in use, a device name will be displayed. Issue the following command:

```
$ SHOW DEVICE/FULL device_name
```

The output of this command will contain a process ID. Issue the following command:

```
$ SHOW PROCESS/CONTINUOUS/ID=pid
```

The full path to the program that is running and which is using the port will be indicated in this display. This path includes the Oracle home from which you should stop the Listener.

Identifying Required Software Directories

You must identify three directories for the Oracle software as described in the following sections:

- [Section , "Oracle Inventory Directory"](#)
- [Section , "Oracle Clusterware Home Directory"](#)
- [Section , "Oracle Home Directory"](#)

The following subsections describe the requirements for these directories.

Oracle Inventory Directory

The Oracle inventory directory (ORAINVENTORY) stores an inventory of all Oracle software installed on the system. It is required by, and shared by, all Oracle software installations on a single system.

The Installer creates this directory if it does not already exist and sets the correct owner, group, and permissions on it.

Note: Oracle recommends that you specify the inventory location by creating the `oraInst.loc` file in the `sys$login:[node_name]` directory. Create this directory if it does not exist. The contents of the `oraInst.loc` file is as follows:

```
inst_group=VMS1
inventory_loc=/VI04/INSTALL_TESTING/SRAVINDH/SAM_INVENTORY_CPQ064
```

VMS1 is the UIC group of the account used to install Oracle. All Oracle software installations depend on this directory. Ensure that you back it up regularly. Do not delete this directory unless you have completely removed all Oracle software from the system.

Oracle Clusterware Home Directory

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

```
device:[ORACLE.CRS_HOME102]
```

Oracle Home Directory

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

Oracle Universal Installer creates the directory path that you specify. It also sets the correct ownership and permissions. You do not need to create this directory.

Guidelines for Placing Oracle Recovery Files

Note: You must select a location for the recovery files only if you enable automated backups during the installation.

Use the following guidelines to place Oracle Database or recovery files:

- To prevent disk failure from making both the database files and the recovery files unavailable, place the recovery files on a different physical disk from the database files.

- The disk that you select must have at least 2 GB of free space.

The disk space requirement is the default disk quota configured for the flash recovery area. The default disk quota is specified by the `DB_RECOVERY_FILE_DEST_SIZE` initialization parameter. If you select the Custom installation type, then you can specify a different disk quota value. After you create the database, you can also use Oracle Enterprise Manager Grid Control to specify a different value.

See Also: *Oracle Database Backup and Recovery Basics (B10735-01)* for more information about sizing the flash recovery area

- The Oracle Database account must have write protection to create the files in the specified path.

Creating Directories for Oracle Database or Recovery Files

Use the following guidelines when deciding the location of Oracle Database or recovery files.

- You can choose either a single device or more than one device to store the database files depending on the following criteria.

- If you want to use a single device, then choose a device on a physical device that is dedicated to the database.

For performance and reliability, choose a redundant array of independent disks (RAID) device or a logical volume on more than one physical device and implement the stripe-and-mirror-everything (SAME) methodology.

- If you want to use more than one device, then choose devices that are dedicated to the database.

This method enables you to distribute physical I/O and create separate control files on different devices for increased reliability. You must choose either the Advanced database creation option or the Custom installation type during the installation to implement this method.

- If you intend to create a preconfigured database during the installation, the device (or devices) that you choose must have at least 1.2 GB of free disk space.

For production databases, you must estimate the disk space requirement depending on the use you want to make of the database.

- For optimum performance, the devices that you choose should be used only by the database.

- The Oracle Database account must have write permissions to create the files in the path that you specify.

Creating Required Directories

Note: You must complete this procedure only if you want to place Oracle Database or recovery files on a separate device instead of placing them in the Oracle base directory.

To create directories for Oracle Database or the recovery files on separate devices:

1. Use the `SHOW DEVICE device_name` command to determine the free disk space on each mounted device.
2. From the display, identify the devices that you want to use. [Table 2–8](#) lists the device requirements for recovery files.

Table 2–8 *Device Requirements for Recovery files*

File Type	Device Requirements
Data files	Choose either: <ul style="list-style-type: none"> ■ A single device with at least 1.2 GB of free disk space ■ Two or more devices with at least 1.2 GB of free disk space in total
Recovery files	Choose a device with at least 2 GB of free disk space.

If you are using the same device for more than one type of file, then add the disk space requirements for each type to determine the total disk space requirement.

3. Note the names of the root device directories for the devices that you identified.

Stopping Existing Oracle Processes

See Also: *Oracle Database Administrator's Reference* for information about stopping existing Oracle processes on Oracle RAC

Caution: If you are installing additional Oracle Database 10g products in an existing Oracle home, then stop *all* processes running in the Oracle home. You must complete this task to enable Oracle Universal Installer to relink certain executables and libraries.

If you choose to create a database during the installation, then most installation types configure and start a default Oracle Net Services listener using TCP/IP port 1521 and the IPC key value `EXTPROC`. However, if an existing Oracle Net Services listener process uses the same port or key value, Oracle Universal Installer can only configure the new listener, but it cannot start it. To ensure that the new listener process starts during the installation, you must shut down any existing listeners before starting Oracle Universal Installer.

To determine if an existing listener process is running, and to shut it down if necessary:

1. Log in as the Oracle Database account user.

-
2. Enter the following command:

```
$ PIPE SHOW SYSTEM/OWN=ORACLE | SEARCH SYS$INPUT <last 8 chars of listener name>
```

This command displays the Oracle processes that are running in the `oracle` account. Run the following command to identify the Oracle processes that are running as listener processes:

```
$ WRITE SYS$OUTPUT F$GETJPI(pid, "IMAGNAME")
```

For each process that is running as a listener process, run the following command to determine the Oracle home in which it is running:

```
$ SHOW PROCESS/CONTINUOUS process_name
```

In this example, *process_name* is the name of the listener process that is indicated by the command that you just entered.

Note: If no Oracle Net Services listeners are running, then refer to [Section , "Configuring the oracle User's Environment"](#) to continue.

3. Set up the Oracle environment for the Oracle home area where the listener is running as follows:

```
$ SET DEFAULT Oracle_home_dir
$ @ORAUSER
```

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

```
LSNRCTL STATUS listenername
```

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

5. If the TCP/IP port number determined in the preceding step is 1521, or the IPC key value is `EXTPROC`, then enter a command similar to the following:

```
LSNRCTL STOP listenername
```

6. Repeat this procedure to stop all listeners running on this system.

See Also: [Appendix I](#) for a list of commands that you can use when performing this procedure

Configuring the oracle User's Environment

Before you start Oracle Universal Installer, you must configure the environment of Oracle database account.

To achieve this, perform the following steps:

1. Start a new terminal session.
2. Set up display using the `SET DISPLAY` command.
3. Set up the inventory directory, and create the `ORAINST.LOC` file. This step is optional. However, Oracle recommends that you do it, if not an `ORAINST.LOC`

file is created under the `sys$login:node_name` directory. The default inventory location is the directory `INVENTORY_node_name` created as the top-level directory of `ORACLE_HOME`.

4. Define the logical name `ORA_BATCH_QUEUE_nodename` to point to an existing batch queue. Oracle recommends that you use a dedicated Batch Queue

See Also: [Appendix D, "Installing, Configuring, and Running EMAgent"](#) for more information about batch queues

5. Create a file, `CLUSTER_NODES.DAT` (if you are installing Oracle Clusterware), in the `SYS$LOGIN` directory with the list of nodes that are to comprise the Oracle RAC cluster. This file displays each node on a separate line with the primary node listed first.
6. Enter the `$ SET NOVERIFY` command.
7. Enter the `$ SET PROCESS/PRIV=ALL` command.
8. To start the `RUNINSTALLER` refer to [Section , "Installing Oracle Database Software"](#)

The ORATAB File

The `ORATAB` file contains information about all Oracle installations and databases installed on the node. This information is used to discover Oracle product installations. When an Oracle product is installed, this file is updated with information about the new Oracle home. If this file is not found, then a new `ORATAB` file is created.

To check if the `ORATAB` file exists:

1. Check for the `ORATAB` file in the `hostname` subdirectory of `SYS$LOGIN`. For example:

```
DISK:[mylogin.hostname]ORATAB.
```

2. If the `ORATAB` file is not found at the location described in Step 1, then use the file in the `SYS$LOGIN` directory. For example:

```
DISK:[mylogin]ORATAB.
```

3. If the `ORATAB` file is not found in any of the above locations, then it means there is no `ORATAB` file.

Installation Tasks

This chapter describes how to start Oracle Universal Installer and install Oracle products on the system. It includes information about the following topics:

- [Section , "Installation Overview"](#)
- [Section , "Downloading Oracle Software from the Oracle Technology Network Web Site"](#)
- [Section , "Installing Oracle Database Software"](#)
- [Section , "Installing Oracle Database Client"](#)
- [Section , "Reviewing the Log of an Installation Session"](#)
- [Section , "Cleaning Up After a Failed Installation"](#)

Note: Review the information in [Chapter 1, "Installation Overview"](#) and complete the tasks listed in [Chapter 2, "Preinstallation Tasks"](#) before beginning the installation.

Installation Overview

Oracle Database software is available on Installation media or you can download it from the Oracle Technology Network Web site. In most cases, you use the graphical user interface (GUI) provided by Oracle Universal Installer to install the software. However, you can also use Oracle Universal Installer to complete noninteractive installations without using the GUI. Refer to [Appendix B](#) for information about noninteractive installations.

This chapter describes how to install Oracle Database from the product media or from the hard disk. To install the software from the hard disk, you must either download it from Oracle Technology Network and unpack it, or copy it from the discs if you have them. Refer to one of the following sections, depending on the method that you want to use:

- To copy the software to a hard disk, refer to [Section , "Installing Oracle Database from the Hard Drive"](#).
- To download the software from Oracle Technology Network, refer to [Section , "Downloading Oracle Software from the Oracle Technology Network Web Site"](#).
- To install the software from disc or from an existing hard disk location, refer to [Section , "Installing Oracle Database Software"](#).

Reviewing Product-Specific Installation Guidelines

Review the following guidelines before starting Oracle Universal Installer:

- Oracle Universal Installer
Do not use Oracle Universal Installer from an earlier Oracle product release to install components from this release.
- Oracle Label Security
To install the Oracle Label Security option, you must select a Custom installation type.

Note: Before you perform a custom installation, ensure that all databases in the Oracle home where you want to install Oracle Label Security are shut down.

If you want to configure Oracle Label Security to use Oracle Internet Directory, then you must select the Oracle Internet Directory option when running Database Configuration Assistant.

- Installations on a Cluster
If Oracle Clusterware is already installed on the system, Oracle Universal Installer displays the Specific Hardware Cluster Installation Mode screen. You must select **Local Installation** on this screen.

If you want to install Oracle RAC, refer to [Chapter 5, "Installing Oracle RAC"](#) of this guide.

If Oracle Clusterware is already installed on the system and you want to install a non-RAC-enabled database, define the following logical name before starting RUNINSTALLER:

```
$ DEFINE ORA_DB_NONRAC_INSTALL TRUE
```

- Products not installed by default: choose **Custom** installation type. These products are:

Oracle Label Security

To configure Oracle Label Security to use Oracle Internet Directory, choose the Oracle Internet Directory option when running Database Configuration Assistant. If you are installing Oracle Label Security in an existing Oracle home, then shut down each database in the Oracle home.

Downloading Oracle Software from the Oracle Technology Network Web Site

This section describes how to download the installation archive files and extract them on your hard disk.

Downloading the Installation Archive Files

To download the installation archive files from Oracle Technology Network:

1. Use any browser to open the software download page on the Oracle Technology Network at

<http://www.oracle.com/technology/>

2. Navigate to the download screen for the product that you want to install.
3. On the download screen, identify the required disk space by adding the file sizes for each required file.
The file sizes are listed next to the file names.
4. Select a device with enough free space to store and expand the archive files.
In most cases, the available disk space must be at least twice the size of all of the archive files.
5. On the device that you select in Step 4, create a parent directory for each product, for example `ORADB10g`, to hold the installation directories.
6. Download all the installation archive files to the directory that you created in Step 5.
7. Verify that the files you downloaded are of the same size as the corresponding files on the Oracle Technology Network Web site.
8. The OTN download site specifies instructions for restoring the files downloaded into an OpenVMS directory tree. After following those instructions, you can proceed as if you had copied the media from the DVD.

Installing Oracle Database Software

You can install Oracle Database either from the hard disk or from installation media. The following sections describes the methods for installing Oracle Database from these media.

- [Section , "Installing Oracle Database from the Hard Drive"](#)
- [Section , "Installing Oracle Database from Install Media"](#)

Installing Oracle Database from the Hard Drive

You can avoid the need to mount and unmount installation media during installation by copying the contents of each installation medium an On-Disk Structure Level 5 (ODS-5) formatted disk. You must have enough disk space available, depending on the type of install.

1. Set up an X Window environment by using the following DCL command:

```
$ SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```
2. Copy the install media files onto a hard drive. Create a directory to hold the contents of the DVD on an ODS-5 disk. For example, you may perform the following steps, replacing `DISK$VMS` with an appropriate ODS-5 disk name, `DQA0` with the name of the DVD drive, and `ORACLE` with the name of the Oracle Database account:

```
$ CREATE/DIR
DISK$VMS:[10GKIT]/OWNER=ORACLE
$ SET DEF DISK$VMS:[10GKIT]
$ MOUNT/OVER=IDENT DQA0 ! Mount DVD
$ SET PROC/PARSE=EXTENDED
$ COPY/READ/WRITE DQA0:[000000...]*.*;*
[...]/EXCLUDE=*.SYS
$ DISMOUNT DQA0
```

The software is now staged on disk and ready for installation using Oracle Universal Installer.

3. Start Oracle Universal Installer from the [10GKIT.DATABASE] directory on the hard drive.
4. To continue with the installation process, complete Steps 3 through 12 in [Section , "Installing Oracle Database"](#).

Installing Oracle Database from Install Media

Set up an X Window environment by using the following DCL command:

```
$SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```

The following procedure describes the installation of products with the use of the Install Media files and Oracle Universal Installer.

- [Section , "Mount the DVD"](#)
- [Section , "Installing Oracle Database"](#)
- [Section , "Completing Oracle Database Server Enterprise Edition Installation"](#)

Mount the DVD

Mount the Oracle Installation DVD. For example:

```
$ MOUNT/OVERRIDE=IDENTIFICATION ddcn
```

where:

ddcn is a valid DVD drive.

Caution: Oracle *does not support* using Oracle Universal Installer that is shipped with any prior release to install Oracle Database 10g components.

Installing Oracle Database

To install Oracle Database from the DVD:

1. Log into the Oracle Database account
2. Start Oracle Universal Installer from the DATABASE directory on the DVD with the following command:

```
$ @ddcn: [DATABASE] RUNINSTALLER
```

If the default temp directory (SYS\$SCRATCH) is not ODS-5, then you must use an alternate temporary directory that is on an ODS -5 volume. For more information about ODS-5 format requirements, refer to [Section , "Changes and Enhancements"](#).

```
$ @ddcn: [DATABASE] RUNINSTALLER -SCRATCHPATH ODS5_temp_dir
```

After Oracle Universal Installer is started, the Welcome screen is displayed. Click **Next** to continue the installation.

3. The Inventory Location screen is displayed if this is the first time that any Oracle Database product is being installed on the system.
 - a. Enter a directory where the inventory of all installations should be kept.

Note: This directory must be on an ODS-5 disk.

- b. If this is not the first time any Oracle Database products are installed on the current system, then go to the next step.
 - c. Click **Next** to continue the installation.
4. The File Locations screen is displayed.
- a. **Source:** You need not change the default. This points to the installation files.
 - b. **Destination (Name):** Enter a name to refer to the Oracle home.
 - c. **Destination (Path):** Enter the path of the Oracle home.

Note: The Oracle home must be on an ODS-5 disk.

- d. Click **Next**
5. The Available Products screen is displayed.
- a. Select Oracle Database.
 - b. Click **Next** to continue.
6. The Installation Types screen is displayed.
- a. Select an installation type: Enterprise Edition, or Custom.
 - b. Click **Next**.
7. The Database Configuration screen is displayed.
- a. Select a database that is suited to your needs.
 - b. Click **Next**.
8. The Database Identification screen is displayed.
- a. The **Software Only** option will not install the starter database.
 - b. Enter a value for the **Global Database Name** for this database.
 - c. Enter a value for the system identifier.
-
-
- Note:** The System Identifier must contain no more than 5 characters on OpenVMS-based systems.
-
-

- d. Click **Next**.
9. The Datafiles Location screen is displayed. Enter a path for the location of the data files. Click **Next**.
10. The Database Character Set screen is displayed. Select the desired character set. Click **Next**.
11. The Install screen is displayed. This screen indicates installation progress. Any errors or warnings are displayed in a dialog box where you can select to fix and retry, to ignore and continue, or to quit.
- a. If you chose to install **Software Only**, then the End of Installation screen is displayed next.

-
- b. If you chose to install a starter database, then the **Configuration Tools** screen is displayed, followed by Database Configuration Assistant screen.
 - c. When configuration is complete, the End of Installation screen is displayed.
 12. From the End of Installation screen, you may exit or install additional products.
 - a. This screen is displayed when the installation is complete. You may choose to exit Oracle Universal Installer or to begin another installation:
 - b. Click **Exit** to end Oracle Universal Installer session.
 - c. Click **Next Install** to begin another installation without stopping and restarting Oracle Universal Installer.

Completing Oracle Database Server Enterprise Edition Installation

If no errors occur during installation, then the installation is complete. To create a database instance, use Database Configuration Assistant.

See Also: [Section , "Database Configuration Assistant"](#)

Installing Oracle Database Client

To install Oracle Database Client:

1. Log into the Oracle Database account
2. Start Oracle Universal Installer from the CLIENT directory on the DVD with the following command:

```
$ @ddcn: [CLIENT]RUNINSTALLER
```

If the default temp directory (SYS\$SCRATCH) is not ODS-5, then you must use an alternate temporary directory that is on an ODS -5 volume. For more information about the use of ODS-5 format, refer to [Section , "Changes and Enhancements"](#).

```
$ @ddcn: [CLIENT]RUNINSTALLER -SCRATCHPATH ODS5_temp_dir
```

3. After Oracle Universal Installer starts, the Welcome screen is displayed.
Click **Next** to continue the installation.
 4. The Inventory Location screen is displayed.
 - a. If this is the first time that any Oracle Database products are installed on the current system, then the Inventory Location screen is displayed.
Enter a directory where an inventory of all installations should be kept.

Note: This directory must be on an ODS-5 disk.

- b. If this is not the first time any Oracle Database products are installed on the current system, then refer to the next step.
 5. The File Locations screen is displayed.
 - a. **Source:** You should not need to change the default. This points to the installation files.
 - b. **Destination (Name):** Enter a name to refer to the Oracle home.
 - c. **Destination (Path):** Enter the path for the Oracle home.

Note: The Oracle home must be on an ODS-5 disk.

- d. Click **Next**.
6. The Available Products screen is displayed. Select **Oracle Database Client**. Click **Next** to continue.
7. The Installation Types screen is displayed. Select an installation type: Custom, Administrator, or Runtime, and then click **Next**.
8. The Summary screen is displayed. This screen displays all of the options that you select for the installation. Click **Install** to proceed with the installation, or click **Previous** to change your selections.
9. The Install screen is displayed. This screen indicates installation progress. Errors or warnings are displayed in a dialog box where you can choose to fix and retry, to ignore and continue, or to quit.
10. From the End of Installation screen, you can exit or install additional products. This screen is displayed when the installation is complete. You can exit Oracle Universal Installer or begin another installation:
 - a. Click **Exit** to end this session with Oracle Universal Installer.
 - b. Click **Next Install** to begin another installation without stopping and restarting the Installer.

Oracle Database Custom Installation

If you select the Custom Installation option, the Available Product Components screen is displayed. It lists all the products and components that are available for installation. A typical custom installation configuration is selected by default. Some actions that may differ from a standard installation are:

1. Select the products that you want to install, or deselect the products that you do not want to install, and click **Next**.
2. Provide responses to any screen prompts that are displayed.

With custom selection of products, you may want to change default values for items such as: port numbers or product-specific directory names.

In the end, the Summary screen is displayed.

Reviewing the Log of an Installation Session

Oracle Universal Installer creates the `ORAINVENTORY` directory the first time that it is run on a computer. It does this to keep an inventory of products that it installs on the system as well as an inventory of other installation information.

The location of the `ORAINVENTORY` directory is defined in the `ORAINST.LOC` file, which is located in the `nodename` sub directory which refers to the node where the install was done. An inventory is created the first time that an Oracle Universal Installer installation is performed on a particular system. The inventory is then detected and referenced during subsequent installation sessions.

Note: If the same node is used, the `ORAINST.LOG` is detected and referenced only during subsequent installation sessions. If on a cluster, this represents a problem.

Oracle suggests that you create additional nodename directories and copy the `ORAINST.LOG` file to them so that any node can subsequently be used for additional install actions.

The log file of the most recent installation is `INSTALLACTIONS.LOG` located in the `[.LOGS]` subdirectory of the `ORAINVENTORY` directory.

`INSTALLACTIONSdate_time.LOG`

For example:

`INSTALLACTIONS 2001-02-14_09-00-56-AM.LOG`

Caution: Do not delete or manually alter the `ORAINVENTORY` directory or its contents. Doing so can prevent Oracle Universal Installer from locating products that install on the system.

The `MAKE.LOG` file in the `ORA_ROOT:[INSTALL]` directory contains a log of every make file action that is run during the installation process. The `make.log` file also records any link errors during installation. Do not delete or alter the `make.log` file.

Cleaning Up After a Failed Installation

If an installation fails, then (before you attempt another installation) you may need to remove files that Oracle Universal Installer created during the last session.

To clean up after a failed installation:

1. Start Oracle Universal Installer.
2. Click **De-install Products** and select any products that were left after the failed installation.
3. Click **Remove**.

To complete the clean up, you may need to manually remove the `ORA_ROOT` directory.

Postinstallation Tasks

This chapter describes how to complete postinstallation tasks after you have installed the software. It includes information about the following topics:

- [Section , "Database Configuration Assistant"](#)
- [Section , "Required Postinstallation Tasks"](#)
- [Section , "Configuring New or Upgraded Databases"](#)
- [Section , "Changing User Passwords"](#)
- [Section , "Performing Configuration Tasks as the SYSTEM User"](#)
- [Section , "Performing Configuration Tasks as the oracle User"](#)
- [Section , "Required Product-Specific Postinstallation Tasks"](#)

If you install and intend to use any of the products listed in [Section , "Required Product-Specific Postinstallation Tasks"](#), you must perform the tasks listed in the product-specific subsections.

Note: This chapter describes basic configuration only. Refer to *Oracle Database Administrator's Reference* and the product-specific administration and tuning guides for more sophisticated configuration and tuning information.

Database Configuration Assistant

The following sections describe the procedure for starting Database Configuration Assistant:

- [Section , "Running Database Configuration Assistant in Interactive Mode"](#)
- [Section , "Using Database Configuration Assistant Response Files"](#)
- [Section , "Database Configuration Assistant Unsupported Options"](#)

Running Database Configuration Assistant in Interactive Mode

To start Database Configuration Assistant in interactive mode, perform the following:

1. Set up the X Window environment by using the following DCL command:

```
SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```

2. Set default to the [BIN] directory in the Oracle home directory, and then enter the following command to start DBCA:

```
$ @disk:[oracle_home_directory]ORAUSER
$ SET DEFAULT ORA_ROOT:[BIN]
$ DBCA
```

Note: Click **Finish** to go to the last screen. This accepts the default answers to the questions on the other screens. However, if DBCA determines that the default answers to one or more screens do not apply, the DBCA will not display the last screen.

Database Configuration Assistant graphical user interface (GUI) is displayed.

See Also: For information about running Database Configuration Assistant in noninteractive or silent mode, refer to [Appendix , "Running Database Configuration Assistant in Noninteractive or Silent Mode"](#)

Using Database Configuration Assistant Response Files

An alternative to creating an argument file to pass command-line options to Database Configuration Assistant is to use a response file. A sample response file (DBCA.RSP) is provided in the ORA_ROOT:[ASSISTANTS.DBCA.LOGS] directory. Make a copy of this file and customize it as necessary. Follow the instructions that are contained in comments. The value that is specified for the CREATE_TYPE field determines which of the other sections are applicable. Each field has a comment header block with the following entries: Name, Datatype, Description, Valid values, Default value, and Mandatory (Yes/No) descriptions.

For example:

```
#
# _____
# Name       : CREATE_TYPE
# Datatype   : String
# Description : Type of database creation
# Valid values : "createDatabase" \ "createTemplateFromDB" \
#               "createCloneTemplate"
# Default value : None
# Mandatory   : Yes
# _____
CREATE_TYPE = "createDatabase"
```

All mandatory values must be provided, but only for the applicable sections as indicated by the CREATE_TYPE value. After customizing the response file, start Database Configuration Assistant and pass the name of the response file at the command line as follows:

```
$ DBCA -SILENT -RESPONSEFILE full_response_file_name
```

Note: Either OpenVMS-based arguments or UNIX-based arguments may be used in fields that are contained within a response file, but the response file argument that is passed to the DBCA.COM script at the command line with the RESPONSEFILE keyword must be in OpenVMS-based syntax.

Database Configuration Assistant Unsupported Options

The following Database Configuration Assistant options are not supported with this release:

- OLAP
- UltraSearch
- Sales History Demonstration

Required Postinstallation Tasks

You must perform the tasks described in the following sections after completing an installation:

- [Section , "Downloading and Installing Patches"](#)
- [Section , "Configuring Oracle Products"](#)
- [Section , "Setting the NLS_LANG Environment Variable"](#)

Downloading and Installing Patches

Check the *OracleMetaLink* Web site for required patches for the installation. To download required patches:

1. Use a Web browser to view the *OracleMetaLink* Web site at <https://metalink.oracle.com>
2. Log on to *OracleMetalink*.

Note: If you are not an *OracleMetaLink* registered user, click **Register for MetaLink!** and follow the registration instructions.

3. On the main *OracleMetaLink* screen, click **Patches**.
4. Select **Simple Search**.
5. Specify the following information, then click **Go**:
 - In the **Search By** field, select **Product** or **Family**, then specify the RDBMS Server.
 - In the **Release** field, specify the current release number.
 - In the **Patch Type** field, specify **Patchset/Minipack**.
 - In the **Platform or Language** field, select the platform.

Configuring Oracle Products

Many Oracle products and options must be configured before you use them for the first time. Before using individual Oracle products or options, refer to the appropriate manual in the product documentation library.

Setting the NLS_LANG Environment Variable

NLS_LANG is a logical name that specifies the locale behavior for Oracle software. This variable sets the language and territory used by the client application and the database

server. It also sets the character set for the client, which is the character set for data entered or displayed by an Oracle client program, such as SQL*Plus.

See Also: [Appendix F, "Configuring Oracle Database Globalization Support"](#) for more information about the NLS_LANG logical name

Configuring New or Upgraded Databases

Oracle recommends that you run the UTLRP.SQL script after creating or upgrading a database. This script recompiles all PL/SQL modules that may be in an invalid state, including packages, procedures, and types. This is an optional step but Oracle recommends that you do it during installation and not at a later time.

To run the UTLRP.SQL script:

1. Log into the oracle Database account.
2. Set the default to the Oracle home directory and run the following command:

```
$ @ORAUSER sid
```

where *sid* is the database instance ID of the target database.

3. Start SQL*Plus as follows:

```
$ SQLPLUS "/ AS SYSDBA"
```

4. If necessary, start the database:

```
SQL> STARTUP
```

5. Run the UTLRP.SQL script as follows:

```
SQL> @ORA_ROOT:[RDBMS.ADMIN]UTLRP.SQL
```

Changing User Passwords

Oracle recommends that you change the password for system user such as SYS, SYSTEM and SYSMAN immediately after installation. To change a password:

1. Start SQL*Plus by using the following command:

```
$ SQLPLUS
```

2. Connect with the user name and password that you want to change by using the following command:

```
Enter user-name: username/password
```

3. Change the password by using the following command:

```
SQL> ALTER USER username IDENTIFIED BY password;
```

Performing Configuration Tasks as the SYSTEM User

You must log into a System Administrator account to perform the following configuration task

Automating Database Startup and Shutdown (Optional)

You can configure the system to automatically start Oracle Database when the system starts up (in your SYSTARTUP procedure) and to shut down Oracle Database when the system shuts down (in your SYSHUTDOWN procedure). Automating database startup is optional, but automatic shutdown is recommended because it guards against improper shutdown of the database.

Performing Configuration Tasks as the oracle User

You must log into the Oracle Database account to set initialization parameters.

Setting Initialization Parameters

When you create a database, the `INITSID.ORA` parameters are automatically set. You can manually modify the initialization parameters in the `initsid.ora` with an OpenVMS editor. Activate the modified `initsid.ora` file by shutting down and restarting the database.

Do not use logical names in parameter files unless they are defined at an appropriate level (in other words, at a group level or the system level).

See Also: *Oracle Database Administrator's Reference* for information about `INIT.ORA` parameters

Required Product-Specific Postinstallation Tasks

The following sections describe postinstallation tasks that you must perform if you install and intend to use the following products:

- [Section , "Configuring Oracle Net Services"](#)
- [Section , "Configuring Oracle Label Security"](#)
- [Section , "Installing Oracle Text Supplied Knowledge Bases"](#)
- [Section , "Configuring Oracle Messaging Gateway"](#)
- [Section , "Oracle Precompilers"](#)

Note: You must perform postinstallation tasks only for the products that you intend to use.

Configuring Oracle Net Services

If you have a previous release of Oracle software installed on this system, you may want to copy information from the Oracle Net Services `TNSNAMES.ORA` and `LISTENER.ORA` configuration files from the previous release to the corresponding files for the new release. These files should have a record format of `stream_lf`. If you copy a `TNSNAMES.ORA` or a `LISTENER.ORA` file from an earlier version, use the following command to convert the files:

```
$ CONVERT/FDL=SYS$INPUT filename
filename
RECORD
CARRIAGE_CONTROL carriage_return
FORMAT stream_LF ^Z
```

Note: The default location for the `TNSNAMES.ORA` and `LISTENER.ORA` files is the `ORA_ROOT : [NETWORK.ADMIN]` directory.

Modifying the listener.ora File

If you are upgrading from a previous release of Oracle Database, Oracle recommends that you use the current release of Oracle Net Services listener instead of the listener from the previous release.

To use the listener from the current release, you may need to copy static service information from the `LISTENER.ORA` file from the previous release to the version of that file used by the new release.

Modifying the tnsnames.ora File

Unless you use a central `TNSNAMES.ORA` file, copy Oracle Net Services service names and connect descriptors from the previous release `TNSNAMES.ORA` file to the version of that file used by the new release.

If necessary, you can also add connection information for additional database instances to the new file.

Configuring Oracle Label Security

If you install Oracle Label Security, you must configure it in a database before you use it. You can configure Oracle Label Security in two ways, with or without Oracle Internet Directory integration. If you configure Oracle Label Security without Oracle Internet Directory integration, you cannot configure it to use Oracle Internet Directory at a later stage.

Note: To configure Oracle Label Security with Oracle Internet Directory integration, Oracle Internet Directory must be installed in the environment and Oracle Database must be registered in the directory.

See Also: For more information about Oracle Label Security enabled with Oracle Internet Directory, refer to the *Oracle Label Security Administrator's Guide*

Installing Oracle Text Supplied Knowledge Bases

An Oracle Text knowledge base is a hierarchical tree of concepts used for theme indexing, `ABOUT` queries, and deriving themes for document services. If you plan to use any of these Oracle Text features, you can install two supplied knowledge bases (English and French) from the Oracle Database 10ginstallation media.

Note: You can extend the supplied knowledge bases depending on your requirements. Alternatively, you can create knowledge bases, possibly in languages other than English and French. For more information about creating and extending knowledge bases, refer to the *Oracle Text Reference*.

Configuring Oracle Messaging Gateway

To configure Oracle Messaging Gateway, refer to the section about Messaging Gateway in *Oracle Streams Advanced Queuing User's Guide and Reference*. When following the instructions listed in that manual, refer to this section for additional instructions about configuring the `LISTENER.ORA`, `TNSNAMES.ORA`, and `MGW.ORA` files.

Modifying the LISTENER.ORA File for External Procedures

To modify the `ORACLE_HOME:[NETWORK.ADMIN]LISTENER.ORA` file for external procedures:

1. Back up the `LISTENER.ORA` file.
2. Ensure that the default IPC protocol address for external procedures is set as follows:

```
(ADDRESS = (PROTOCOL=IPC) (KEY=EXTPROC))
```

3. Add static service information for a service called `mgwextproc` by adding the following lines to the `SID_LIST` parameter for the listener in the `LISTENER.ORA` file:

```
(SID_DESC =  
  (SID_NAME = mgwextproc)  
  (ORACLE_HOME = oracle_home)  
  (PROGRAM = extproc)  
)
```

For example, set the `SID_LIST` parameter as follows:

```
SID_LIST_LISTENER =  
(SID_LIST =  
  (SID_DESC =  
    (SID_NAME = PLSExtProc)  
    (ORACLE_HOME = somedisk:[oracle.product.10_1_0.db_1])  
    (PROGRAM = extproc)  
  )  
  (SID_DESC =  
    (SID_NAME = mgwextproc)  
    (ORACLE_HOME = somedisk:[oracle.product.10_1_0.db_1])  
    (PROGRAM = extproc)  
  )  
)
```

In this example:

- `ORACLE_HOME` is the path of the Oracle home directory.
- `extproc` is the external procedure agent executable file.

Modifying the TNSNAMES.ORA File for External Procedures

To modify the `ORACLE_HOME:[NETWORK.ADMIN]TNSNAMES.ORA` file for external procedures:

1. Back up the `TNSNAMES.ORA` file.
2. In the `TNSNAMES.ORA` file, add a connect descriptor with the net service name `MGW_AGENT` as follows:

```
MGW_AGENT =  
(DESCRIPTION=
```

```
(ADDRESS_LIST= (ADDRESS= (PROTOCOL=IPC) (KEY=EXTPROC)))
(CONNECT_DATA= (SID=mgwextproc) (PRESENTATION=RO))
```

In this example:

- The value specified for the `KEY` parameter must match the value specified for that parameter in the IPC protocol address in the `LISTENER.ORA` file.
- The value of the `SID` parameter must match the service name in the `LISTENER.ORA` file that you specified for the Oracle Messaging Gateway external procedure agent in the previous section (`mgwextproc`).

Setting Up the MGW.ORA Initialization File

To modify the `ORA_ROOT:[MGW.ADMIN]MGW.ORA` file for external procedures, define the `CLASSPATH` logical name. A sample `MGW.ORA` file can be found in the `ORA_ROOT:[MGW.ADMIN]SAMPLE_MGW.ORA` directory.

You must include the classes in [Table 4–1](#) and any additional classes required for Oracle Messaging Gateway to access non-Oracle messaging systems, such as WebSphere MQ or TIBCO Rendezvous classes.

Table 4–1 Classes for `CLASSPATH` Logical Name

Classes	Path
Oracle Messaging Gateway	<code>ORA_ROOT:[MGW.JLIB]MGW.JAR</code>
JRE run time	<code>ORA_ROOT:[JRE.LIB]RT.JAR</code>
Oracle JDBC	<code>ORA_ROOT:[JDBC.LIB]OJDBC14.JAR</code>
Oracle internationalization	<code>ORA_ROOT:[NLSRTL.JLIB]ORAI18N.JAR</code>
SQLJ	<code>ORA_ROOT:[SQLJ.LIB]TRANSLATOR.JAR</code> <code>ORA_ROOT:[SQLJ.LIB]RUNTIME12.JAR</code>
JMS Interface	<code>ORA_ROOT:[RDBMS.JLIB]JMSCOMMON.JAR</code>
Oracle JMS implementation	<code>ORA_ROOT:[RDBMS.JLIB]AQAPIL3.JAR</code>
Java Transaction API	<code>ORA_ROOT:[JLIB]JTA.JAR</code>

Oracle Precompilers

The following sections describe references and configuration files that are associated with Oracle precompilers.

Pro*C/C++

For the Pro*C/C++ precompiler, the configuration file is `ORA_ROOT:[PRECOMP.ADMIN]PCSCFG.CFGS`. This file installs without content. Use any text editor to customize it to the site requirements. For more information about configuring the `pcscfg.cfg` file, refer to *Pro*C/C++ Programmer's Guide*.

Pro*COBOL

For the Pro*COBOL precompiler, the configuration file is `PCBCFG.CFG`. This file installs without content. Use any text editor to customize it to the site requirements. For more information about configuring the `pcbcfg.cfg` file, refer to *Pro*COBOL Programmer's Guide*.

Pro*FORTRAN

For the Pro*FORTRAN precompiler, the configuration file is `PCCFOR.CFG`. This file installs without content. Use any text editor to customize it to the site requirements. For more information about configuring the `pccfor.cfg` file, refer to *Pro*FORTRAN Supplement to the Oracle Precompilers Guide*.



Installing Oracle RAC

This chapter provides an overview of the procedure for installing Oracle Database 10g RAC.

You may also need to refer to the following Oracle documents for detailed information about installing and administrating Oracle RAC on an OpenVMS system:

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

Oracle RAC configuration is described in [Chapter 6, "Configuring Oracle RAC"](#).

This chapter contains the following topics:

- [Section , "Oracle RAC Documentation Overview"](#)
- [Section , "Quota Considerations"](#)
- [Section , "Cluster Setup and Preinstallation Configuration Tasks"](#)
- [Section , "Preinstallation, Installation, and Postinstallation Overview"](#)
- [Section , "Oracle Universal Installer and Oracle Real Application Clusters"](#)
- [Section , "Additional Considerations"](#)
- [Section , "Oracle Database and Oracle Real Application Clusters Components"](#)
- [Section , "Oracle Real Application Clusters Version Compatibility"](#)
- [Section , "Installing Oracle Clusterware"](#)
- [Section , "Installation Tasks for Oracle Real Application Clusters"](#)
- [Section , "Selecting a Database Configuration Type"](#)
- [Section , "Procedure for Installation Setup"](#)
- [Section , "Installing Oracle Database with Oracle Real Application Clusters"](#)
- [Section , "Postinstallation Procedures for Oracle Real Application Clusters"](#)
- [Section , "Uninstalling Oracle Real Application Clusters Software"](#)

Oracle RAC Documentation Overview

This section describes the Oracle RAC documentation set. The platform-specific Oracle Database 10g installation media contains a copy of *Oracle Database Installation Guide* in both HTML and PDF format. This chapter contains the preinstallation, installation,

and postinstallation information for Oracle RAC. Additional information for this release may be available in the Oracle Database 10g README file or Release Notes.

The Server Documentation installation media contains the following additional documentation about Oracle Real Application Clusters administration and deployment:

- [Section , "Oracle Real Application Clusters Administrator's Guide"](#)
- [Section , "Oracle Real Application Clusters Deployment and Performance Guide"](#)
- [Section , "Documentation Sources"](#)

Oracle Real Application Clusters Administrator's Guide

Oracle Real Application Clusters Administrator's Guide provides Oracle Real Application Clusters-specific administration information. Some of the topics described in this book include the use of Oracle Enterprise Manager in Oracle Real Application Clusters environments. This book describes how to administer services and storage and how to use Oracle Real Application Clusters scalability features to add and delete instances and nodes in Oracle Real Application Clusters environments. This book also discusses how to use Recovery Manager (RMAN) and how to perform backup and recovery in Oracle Real Application Clusters.

In addition, *Oracle Real Application Clusters Administrator's Guide* describes how to use the Server Control (SRVCTL) utility to start and stop the database and instances, manage configuration information, and to delete or move instances and services. The appendix describes how you can resolve various Oracle Real Application Clusters tools messages. A troubleshooting section describes how to interpret the content of Oracle Real Application Clusters-specific log files.

Oracle Real Application Clusters Deployment and Performance Guide

The *Oracle Real Application Clusters Deployment and Performance Guide* highlights the main deployment topics for Oracle Real Application Clusters by briefly describing Oracle Clusterware, storage, database creation, and services deployment in Oracle Real Application Clusters. Design and deployment topics in this reference book describe service topologies and workload management in Oracle Real Application Clusters. Specifically, the book describes how the Automatic Workload Repository tracks and reports service levels and how you can use service level thresholds and alerts to ensure high-availability in a Oracle Real Application Clusters environment. A services deployment example is also available in the appendix of the book, and you can use it to learn more about how to deploy and to manage services in Oracle Real Application Clusters environments.

The *Oracle Real Application Clusters Deployment and Performance Guide* provides a high-level description of interconnect protocols, as well as information about how to monitor and tune performance in Oracle Real Application Clusters environments using both Oracle Enterprise Manager and using information in the Automated Workload Repository and Oracle performance views. The book highlights some application-specific deployment techniques for online transaction processing (OLTP) and data warehousing environments.

Documentation Sources

The information in this chapter is meant to supplement the detailed information you can find in *Oracle Real Application Clusters Administrator's Guide*. Refer to that document as necessary when reading through this chapter.

In addition, you will find important OpenVMS-specific installation and configuration information pertaining to Oracle Real Application Clusters in [Chapter 5, "Installing Oracle RAC"](#) and [Chapter 6, "Configuring Oracle RAC"](#).

For more general information about configuring and using Oracle Real Application Clusters, refer to the *Oracle Real Application Clusters Installation and Configuration Guide*.

Quota Considerations

Oracle Real Application Clusters is a completely redesigned feature from the Parallel Server Option in versions prior to Oracle Database 10g. Oracle Database includes a portable Distributed Lock Manager (DLM) internal to the database server.

Because of this, the usage of the ENQLM quota by the Oracle Real Application Clusters Option has been greatly reduced. Large ENQLM boosts for background processes are no longer necessary in the Oracle Database environment.

The Process Quota estimation functions of the Oracle Database attempt to make reasonable calculations for the various background processes started during server operations. You can change the quota that is allocated to a background process by defining certain system logical names before the instance is started.

To affect the process quota given to all background processes in an instance, define the following:

```
$ DEFINE/SYSTEM ORA_sid_PQL$_quota quota_value
```

If you have a SID named TOM1, and if you want to boost the BYTLM quota to 2 MB, then use the following:

```
$ DEFINE/SYSTEM ORA_TOM1_PQL$_BYTLM 2000000
```

Alternatively, you could effect the process quota of a particular background by defining the following:

```
$ DEFINE/SYSTEM ORA_sid_process_PQL$_quota quota_value
```

If you want the LMD0 process of the TOM1 SID to get 3 MB of BYTLM, then you would use the following:

```
$ DEFINE/SYSTEM ORA_TOM1_LMD0_PQL$_BYTLM 3000000
```

Cluster Setup and Preinstallation Configuration Tasks

Before installing Oracle Real Application Clusters, perform the following steps:

1. Ensure that you have a certified combination of operating system and Oracle software version by referring to the Oracle *MetaLink* certification information at <https://metalink.oracle.com>

On this Web site, click **Certify and Availability**, and then select **1. View Certifications by Product**.

2. Configure a high-speed interconnect that uses a private network. Configure a second interconnect for redundancy to avoid making the interconnect a potential single point of failure. Some platforms support automatic failover to the alternate interconnect. To enable this, you must configure the operating system-provided failover mechanism.

Preinstallation, Installation, and Postinstallation Overview

The following sections describes the installation procedures that are covered in detail in this chapter:

- [Section , "Preinstallation Overview"](#)
- [Section , "Installation Overview"](#)
- [Section , "Postinstallation Overview"](#)

Preinstallation Overview

The preinstallation procedures explain how to verify user equivalence, perform network connectivity tests, as well as how to set directory and file permissions. Complete all of the preinstallation procedures and verify that the system meets all of the preinstallation requirements before proceeding to the install phase.

Installation Overview

Oracle Real Application Clusters installation is a two-phase installation. In phase one, use Oracle Universal Installer to install Oracle Clusterware as described in [Section , "Oracle Clusterware"](#). Note that the Oracle home that you use in phase one is a home for the Oracle Clusterware software. This Oracle home *must* be different from the Oracle home that you use in phase two for the installation of Oracle Database software with Oracle Real Application Clusters components. The Oracle Clusterware preinstallation starts the Oracle Clusterware processes in preparation for installing Oracle Database with Oracle Real Application Clusters as described in [Section , "Installing Oracle Database with Oracle Real Application Clusters"](#).

After the installation is complete, Oracle Universal Installer starts the Oracle assistants, such as Database Configuration Assistant, to configure the environment and create the Oracle Real Application Clusters database. You can later use Database Configuration Assistant Instance Management feature to add or modify services and instances as described in [Section , "Creating the Oracle RAC Database"](#).

Postinstallation Overview

After you create the database, download and install the most recent patch sets for Oracle Database as described in [Section , "Downloading and Installing Patches"](#). If you are using other Oracle products with the Oracle Real Application Clusters database, then you must also configure them.

You must perform several postinstallation configuration tasks to use certain Oracle Database products such as the Sample Schema, Oracle Net Services, or Oracle Messaging Gateway. You must also configure Oracle precompilers, and if required, configure Oracle Advanced Security.

Use the installation media to install additional Oracle Database software that may improve performance or extend database capabilities, such as Oracle JVM, Oracle *interMedia* or Oracle Text.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about using Oracle Real Application Clusters scalability features of adding and deleting nodes and instances from Oracle Real Application Clusters databases

Oracle Universal Installer and Oracle Real Application Clusters

Oracle Universal Installer facilitates the installation of Oracle Clusterware and Oracle Database software. In most cases, you must use the graphical user interface (GUI) provided by Oracle Universal Installer to install the software. However, you can also use Oracle Universal Installer to complete scripted installations without using the GUI. Refer to [Appendix B](#) for information about scripted installations.

When Oracle Universal Installer installs the Oracle software, Oracle recommends that you select a preconfigured database or use Database Configuration Assistant interactively to create the cluster database. You can also manually create the database as described in procedures posted at

<http://www.oracle.com/technology>

You can configure Enterprise Manager Grid Control to manage multiple databases and application servers from a single console. To manage Oracle Real Application Clusters databases in Grid Control, you must install a Grid Control agent on each of the nodes of the cluster. The Agent installation is clusterized, which means you need to perform the install on only one of the cluster nodes.

See Also:

- *Oracle Universal Installer and OPatch User's Guide* for details about Oracle Universal Installer
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for information about using Enterprise Manager to administer Oracle Real Application Clusters environments

Additional Considerations

Following are some additional considerations for using Oracle Database 10g features in Oracle Real Application Clusters. Oracle recommends using the following Oracle Database features to simplify Oracle Real Application Clusters database management:

- Enterprise Manager
Use Enterprise Manager to administer the *entire* processing environment, not just the Oracle Real Application Clusters database. Enterprise Manager enables you to manage a Oracle Real Application Clusters database with its instance targets, listener targets, host targets, and a cluster target.
- Automatic undo management
This feature automatically manages undo processing.
- Automatic segment space management
This feature automatically manages segment freelists and freelist groups.
- Locally managed tablespaces
These enhance space management performance.

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

Oracle Database and Oracle Real Application Clusters Components

Oracle Database provides single-instance database software and additional components to operate Oracle Real Application Clusters databases. Some of the Oracle Real Application Clusters-specific components include:

- [Section , "Oracle Clusterware"](#)
- [Section , "Installed Oracle Real Application Clusters Components"](#)

Oracle Clusterware

Oracle Clusterware must be installed only on one node. However, you must configure and start the Oracle Clusterware on each node where you intend to install or run the Oracle Real Application Clusters-enabled database. The home that you select for Oracle Clusterware must be different from the Oracle Real Application Clusters-enabled Oracle home.

Installed Oracle Real Application Clusters Components

All instances in Oracle Real Application Clusters environments share the control file, server parameter file, redo log files, and all data files. These are accessed by all the cluster database instances. Each instance also has its own set of redo log files. During failures, shared access to redo log files enables surviving instances to perform recovery.

Oracle Real Application Clusters Version Compatibility

You can install and operate multiple Oracle homes and different versions of Oracle cluster database software on the same computer as described in the following points:

- You can install multiple Oracle Database 10g Oracle Real Application Clusters homes on the same node. The multiple homes feature enables you to install one or more releases on the same computer in multiple Oracle home directories. However, each node can have only one Oracle Clusterware home.
- If you run Oracle Universal Installer to install Oracle Real Application Clusters on a system that already has an Oracle Database 10g Oracle Real Application Clusters installation in the same home, then Oracle Universal Installer prompts you to install additional Oracle Database 10g products if you have not already installed all of them.
- You can use Oracle Universal Installer to complete some of the steps involved in removing and reinstalling Oracle Real Application Clusters if required.

Note: Do not move Oracle binaries from one Oracle home to another because this causes dynamic link failures.

Installing Oracle Clusterware

The following sections describe the procedure for installing Oracle Clusterware. This is the first phase of installing Oracle Database with Oracle Real Application Clusters. These sections also explain how to create Oracle Real Application Clusters databases, and describe the postinstallation tasks.

- [Section , "Installation Setup"](#)
- [Section , "Installing Oracle Clusterware"](#)

Installation Setup

In addition to the public IP address of the host computer, obtain two more IP addresses for each node that is going to be part of the installation. During the installation, enter the IP addresses into the DNS. One of the IP addresses must be a public IP address for the virtual IP address of the node. Virtual IP addresses are used for client-to-database connections. Therefore, the virtual IP address must be publicly accessible. The other address must be a *private* IP address for internode, or instance-to-instance Cache Fusion traffic. Using public interfaces for Cache Fusion can cause performance problems. Refer to [Section](#) for further Oracle Clusterware IP address requirements.

Installing Oracle Clusterware

This section describes the procedure for using Oracle Universal Installer to install Oracle Clusterware. Note that the Oracle Clusterware home that you identify in this phase of the installation is only for Oracle Clusterware software. This home cannot be the same home as the home that you use in phase two to install Oracle Database software with Oracle Real Application Clusters.

1. Start a new terminal session.
2. Set up display using the `$ SET DISPLAY` command.
3. Define logical `ORA_BATCH_QUEUE_nodename` to point to an existing batch queue (For more information about batch queue, refer to [Appendix D](#)).

Note: Oracle recommends that you use a dedicated batch queue.

4. Create a file, `CLUSTER_NODES.DAT`, in the `SYS$LOGIN` directory with the list of nodes that are to comprise the Oracle Real Application Clusters cluster. This file displays each node on a separate line with the primary node listed first.
5. Run the following command:

```
$ SET NOVERIFY
```
6. Run the `$ SET PROCESS/PRIV=ALL` command.
7. Start the `RUNINSTALLER` script located in the `[CLUSTERWARE]` directory on the installation medium.
8. If you are performing this installation in an environment where Oracle Universal Installer inventory is already set up, then Oracle Universal Installer displays the Specify File Locations screen. If the Specify File Locations screen is displayed, then proceed to Step 9.
9. The Specify File Locations screen contains predetermined information for the source of the installation files and the target destination information. Enter the Oracle Clusterware home name and its location in the target destination, click **Next**, and Oracle Universal Installer displays the Language Selection screen.

Note: The Oracle Clusterware home that you identify in this step *must be different* from the Oracle home that you will use in phase two of the installation.

-
10. In the Language Selection screen, select the languages that you want Oracle Clusterware to use, click **Next**, and Oracle Universal Installer displays the Cluster Configuration screen.
 11. The Cluster Configuration screen contains predefined node information if Oracle Universal Installer retrieved the nodes specified in `CLUSTER_NODES.DAT`. Otherwise, Oracle Universal Installer displays the Cluster Configuration screen without predefined node information.

Enter a public node name and a private node name for each node. When you enter the public node name, use the primary host name of each node. This node name can be either the permanent or the virtual host name.

In addition, the cluster name that you use must be globally unique throughout the enterprise and the allowable character set for the cluster name is the same as that for host names, that is, underscores (`_`), hyphens (`-`), and single-byte alphanumeric characters (a to z, A to Z, and 0 to 9). Ensure that you also enter a private node name or private IP address for each node. This is an address that is only accessible by the other nodes in this cluster. Oracle uses the private IP addresses for Cache Fusion processing. Click **Next** after you have entered the cluster configuration information, and Oracle Universal Installer performs validation checks such as node availability and remote Oracle home permissions verifications. These verifications may require some time to complete. When Oracle Universal Installer completes the verifications, it displays the Specify Network Interface Usage screen.

Note: The IP addresses that you use for all of the nodes in the current installation process must be from the same subnet.

12. On the Specify Network Interface Usage screen Oracle Universal Installer displays a list of clusterwide interfaces. Use the lists on this screen to classify each interface as `Public`, `Private`, or `Do Not Use`. The default setting for each interface is `Do Not Use`. You must classify at least one interconnect as `Public` and one as `Private`s.
13. When you click **Next** on the Specify Network Interface Usage screen, Oracle Universal Installer will look for the `OCR.LOC` file. Oracle Universal Installer will look for the file in the `logindisk:[logindir.nodename]` directory. If the `OCR.LOC` file exists, and if the `OCR.LOC` file has a valid entry for the Oracle Cluster Registry (OCR) location, then the Voting Disk Location screen is displayed and you should proceed to Step 14.

Otherwise, the Oracle Cluster Registry Location Information screen is displayed. Enter a complete path for the shared file system file for the Oracle Cluster Registry, click **Next**, and the Voting Disk Information screen is displayed.
14. On the Voting Disk Information screen, enter a complete path and file name for the file in which you want to store the voting disk, and then click **Next**.

Note: The path for OCR and the voting disk location must be in Unix style.

For example : `/disk/dir/ocr.dat`

Note: The storage size for the OCR should be at least 100 MB and the storage size for the voting disk should be at least 20 MB. In addition, Oracle recommends that you use a RAID array for storing the OCR and the voting disk to ensure the continuous availability of the partitions.

15. Oracle Universal Installer displays the list of components that will be installed on the Summary screen. Review the list of components and click **Install**.
16. You will be prompted to run `CRSSETUP.COM` on each node. After you have done so, click **Yes** in the dialog box.
17. Oracle Universal Installer runs the Oracle Private Interconnect Configuration Assistant. This assistant runs without your intervention.

Note: If one or more of the configuration assistants fails, then you can manually rerun the script created by Oracle Universal Installer to determine the cause of the failure. You can find this script in the `ORA_ROOT:[CFGTOOLLOGS]` directory. You may need to edit the script to replace placeholders for passwords, to remove steps that completed successfully, and so on.

18. When Oracle Universal Installer displays the End of Installation screen, click **Exit** to exit Oracle Universal Installer.
19. Run the `ORAUSER.COM` command procedure.
20. Verify the Oracle Clusterware installation by running the following `OLSNODES` command:

```
$ OLSNODES
```

The output from this command should be a listing of the nodes on which Oracle Clusterware was installed as shown in the following example:

```
$ OLSNODES -N
racserver1 1
racserver1 2
```

At this point, you have completed phase one, the installation of Oracle Clusterware, and are ready to install Oracle Database 10g with Oracle Real Application Clusters as described in [Section , "Installation Tasks for Oracle Real Application Clusters"](#).

Oracle Clusterware Background Processes

The following processes must be running in the environment after the Oracle Clusterware installation for Oracle Clusterware to function:

- `ORA_HAS_EVMD`
Event manager daemon that starts the `racgevt` process to manage callouts.
- `ORA_HAS_OCSSD`
Manages cluster node membership and runs in the Oracle Database account.
- `ORA_HAS_CRSDREB` (`ORA_HAS_CRSDRES` if the Clusterware has been restarted and the node has not)

Performs high-availability recovery and management operations such as maintaining the OCR.

- ORA_EVMLLOGabcd
Event Logger (abcd will be hex digits)
- ORA_HAS_OCLSVMO
Vendor Clusterware Monitor
- ORA_HAS_OCLSOMO
Clusterware Monitor

Installation Tasks for Oracle Real Application Clusters

The following sections describe phase two of the installation procedures for installing Oracle Database with Oracle Real Application Clusters. They also describe some of Oracle Universal Installer features.

Selecting a Database Configuration Type

This section describes Oracle Universal Installer features that you should understand before beginning phase two of the Oracle Real Application Clusters installation process. When you run Oracle Universal Installer and select Oracle Database 10g, you can select the General Purpose, Transaction Processing, Data Warehouse, or Advanced configuration type.

For the first three configuration types, you can complete additional procedures that are described later in this chapter. If you select the fourth type, or the Advanced configuration, then you can use Database Configuration Assistant to create the database as described in [Section , "Using Database Configuration Assistant for Oracle RAC"](#). Oracle recommends that you use Database Configuration Assistant to create the database.

You can also select the Advanced configuration, select a preconfigured template, customize the template, and use Database Configuration Assistant to create a database using the template. These templates correspond to the General Purpose, Transaction Processing, and Data Warehouse configuration types. You can also use Database Configuration Assistant with the Advanced template to create a database.

Oracle recommends that you use one of the preconfigured database options or use the Advanced option and Database Configuration Assistant to create the database. However, if you want to configure the environment and create the database manually, then select the **Do not create a starter database** configuration option and refer to the manual database creation procedures posted at

<http://www.oracle.com/technology>

This section discusses the following topics:

- [Section , "Configuration Type Descriptions"](#)
- [Section , "Action of Oracle Universal Installer, Database Configuration Assistant, and Other Assistants During Installation"](#)

Configuration Type Descriptions

The configuration type that you select, as described in [Table 5–1](#), determines how you proceed.

Table 5–1 Oracle Universal Installer Database Configuration Types

Configuration Type	Description	Advantages
General Purpose, Transaction Processing, and Data Warehouse	Installs a preconfigured starter database, licensable Oracle options (including Oracle Database with Oracle Real Application Clusters), networking services, Oracle Database utilities, and online documentation. At the end of the installation, Database Configuration Assistant creates and configures the Oracle Real Application Clusters database.	Minimal input required. You can create the database more quickly than with the Advanced type.
Advanced	Enables you to customize the database options and storage components.	Enables you to create arbitrary tablespaces and data files and customize all aspects of the database.
Do not create a starter database	Installs only the software. Does not configure the listeners, network infrastructure, and does not create a database.	

Using the Standard Configuration Types

If you select one of the first three configuration types on Select Database Configuration screen of Oracle Universal Installer, then complete the procedure described in ["Procedure for Installation Setup"](#). These three configuration types use preconfigured templates. After you complete these procedures, the Oracle Network Configuration Assistant and Database Configuration Assistant run without further input and Oracle Universal Installer displays a progress indicator during the installation. Database Configuration Assistant processing for these configuration types creates a starter database and configures the network services.

Using the Advanced Configuration Type

When you select the Advanced configuration type, Oracle Universal Installer runs Database Configuration Assistant which displays the following four preconfigured database template choices:

- General Purpose
- Transaction Processing
- Data Warehouse
- Advanced

The first three templates create a database that is optimized for that environment. You can also customize these templates. The Advanced type, however, creates a database without using preconfigured options.

The following section provides more detail about Oracle Universal Installer and Database Configuration Assistant processing when creating a Oracle Real Application Clusters database.

Action of Oracle Universal Installer, Database Configuration Assistant, and Other Assistants During Installation

After installation, Oracle Universal Installer starts the Oracle Network Configuration Assistant. After the Oracle Network Configuration Assistant completes its processing, Oracle Universal Installer runs Database Configuration Assistant to create the database. This means that Database Configuration Assistant creates the database files, including the default server parameter file (*SPFILE*), using standard file naming and file placement practices. The primary phases of Database Configuration Assistant processing are:

- Verifying that you correctly configure the shared disks for each tablespace if you use raw storage
- Creating the database
- Configuring Oracle network services
- Starting the listeners and database instances

You can also use Database Configuration Assistant in standalone mode to create a database.

See Also: The *Oracle Database Net Services Administrator's Guide* for more information about troubleshooting and LDAP support

The remainder of this chapter explains how to use Oracle Universal Installer to install Oracle Database with Oracle Real Application Clusters.

Procedure for Installation Setup

If the user who installed Oracle Clusterware is the same as the user who installs Oracle Real Application Clusters, then all of the user-level preinstallation steps must be completed.

1. Perform a network connectivity test to ensure that all nodes on which you want to perform this installation can communicate with each other. Ensure that the public and private network interfaces have the same interface names on each node of the cluster.
2. Set directory and file creation permissions so that you can write as `oracle` user to all the Oracle homes in the Real Application Clusters database environment in which you are performing this installation.
3. Create directories for the Oracle home and Oracle data files.

Note: The Oracle home that you create for installing Oracle Database 10g with Oracle Real Application Clusters software cannot be the same Oracle home that you used during the Oracle Clusterware installation. However, you must use the same inventory location as the one used for the Clusterware installation.

You can use the Oracle9i Database language and territory definition files with Oracle Database 10g that you are about to install. To enable this functionality, you must run Oracle Universal Installer from a command line, described in Step 1 in [Section , "Installing Oracle Database with Oracle Real Application Clusters"](#), and use the following statement to set the `b_cr9idata` variable to `true`:

```
$ @RUNINSTALLER ORACLE.RSF.NLSRTL_RSFB_CR9IDATA=TRUE
```

Installing Oracle Database with Oracle Real Application Clusters

Perform the following procedure to install Oracle Database software with Real Application Clusters:

1. Run the `RUNINSTALLER` command from the `[DATABASE]` directory of the Oracle Database 10g release 2 installation media.
2. When Oracle Universal Installer displays the Welcome screen, click **Next**, and Oracle Universal Installer displays the Specify File Locations screen.
3. The Source field on the Specify File Locations screen is pre-populated with the path to the file `PRODUCTS.XML`. Enter the Oracle home name and location in the target destination and click **Next**.

Note: The Oracle home name and path that you use in this step *must be different* from the home that you used during the Oracle Clusterware installation in phase one. In other words, you must not install Oracle Database 10g with Oracle Real Application Clusters software into the same home in which you installed the Oracle Clusterware software.

If you enter a preexisting Oracle home and the Oracle home is not registered with Oracle Universal Installer inventory, then Oracle Universal Installer displays a warning that the Oracle home that you selected is not empty. Click **OK** on this Oracle Universal Installer warning dialog box to continue installing in that directory. If the Oracle home exists and is registered with Oracle Universal Installer inventory, and if the Oracle home was created with a cluster install, then Oracle Universal Installer displays the Selected Nodes screen.

If you enter an Oracle home that does not exist, then Oracle Universal Installer displays the Specify Hardware Cluster Installation Mode screen.

Note: If Oracle Universal Installer does not detect an Oracle Clusterware installation, then the Installer assumes that you are performing a single-node, non-Oracle Real Application Clusters installation. If Oracle Clusterware has been installed, you can force the Installer to perform a single-node, non-Oracle Real Application Clusters installation by defining the logical name `ORA_DB_NONRAC_INSTALL` with the value `TRUE`.

4. The Selected Nodes screen is an informational screen that displays the selected nodes that are associated with a cluster home. Click **Select all** to select all nodes ,then click **Next** on this screen and Oracle Universal Installer displays the Select Installation Type screen.

When you click **Next** on the Selected Nodes screen, Oracle Universal Installer verifies that the Oracle home directory is writable on the remote nodes and that the remote nodes are operating. Oracle Universal Installer also revalidates user equivalence.

If Oracle Universal Installer detects a network or user equivalence problem on any node that you have included in this installation, then the Installer displays a

warning on the Selected Nodes screen. This warning is displayed next to the node and indicates that you should correct a problem on the affected node before proceeding. To resolve problems, examine Oracle Universal Installer actions recorded in the file `INSTALLACTIONSdate_time.LOG` in the `LOGS` subdirectory of the Inventory directory.

5. On the Specify Hardware Cluster Installation Mode screen, select an installation mode. The Cluster Installation mode is selected by default when Oracle Universal Installer detects that you are performing this installation on a cluster. In addition, the local node is always selected for the installation. Select additional nodes that are to be part of this installation session and click **Next**.

See Also: If you select Local Installation, then refer to *Oracle Database Installation Guide* to perform a single-node, non-Oracle Real Application Clusters installation

When you click **Next** on the Specify Hardware Cluster Installation screen, Oracle Universal Installer verifies that the Oracle home directory is writable on the remote nodes and that the remote nodes are operational. Oracle Universal Installer also revalidates user equivalence.

You can either make corrections and click **Next**, or you can deselect nodes that have errors and click **Next**. When you click **Next**, Oracle Universal Installer displays the Select Installation Type screen.

6. On the Select Installation Type screen, you can select Enterprise Edition, or Custom Install type. If you select Enterprise Edition, then the installation will include the components related to the chosen edition. If you select the Custom install, then select the individual components that you want to install from the list of available components.

After you make the selection and click **Next** on the Select Installation Type screen, Oracle Universal Installer displays the Database Configuration screen depending on the configuration and selections. This screen is displayed if no previous release of the database exists.

7. When you click **Next**, the screen that Oracle Universal Installer displays depends on the configuration and selection options as follows:
 - Available Products Components screen: This screen is displayed if you selected a Custom install.
 - Select Database Configuration screen: This screen is displayed if you did not select a Custom install and there is no database to upgrade.

Note: Database Upgrade Assistant is not supported on this release of Oracle on OpenVMS Alpha. To upgrade a database, refer to the steps for manually upgrading a database in *Oracle Database Upgrade Guide*.

8. On the Available Products Components screen, select the components you want to install and click **Next**. This opens the Component Locations screen in some cases, such as when insufficient disk space is available. Specify a location for the installation and click **Next**. Oracle Universal Installer will display the Privileged Operating System Groups screen described in Step 14.
9. The Select Database Configuration screen, provides options for you to create a preconfigured database as part of the installation process or to install the software without creating a database. If you want to create a database as part of the

installation, then select General Purpose, Transaction Processing, Data Warehouse, or Advanced. If you only want to install the software, then select **Do not create a starter database**. Click **Help** for more information about these choices.

If you choose to create a General Purpose, Transaction Processing, or a Data Warehouse database, then when you click **Next**, Oracle Universal Installer displays the Specify Database Configuration Options screen. If you select the Advanced option or select **Do not create a starter database**, then when you click **Next**, Oracle Universal Installer displays the Privileged Operating System Groups screen.

10. On the Specify Database Configuration Options screen, enter a global database name. A global database name is a name that includes the database name and database domain, such as `db.us.acme.com`. The name that you enter on this screen must be unique among all the global database names used in the environment. Accept or change the common prefix for the Oracle SID for each instance. Each instance has an SID that comprises the common prefix that you enter in this step and an instance ID that is automatically generated. Note that an SID prefix cannot exceed five characters. Also select a database character set, select any database examples to install from the Sample Schemas.

When you click **Next** on the Specify Database Configuration Options screen, Oracle Universal Installer displays the Select Database Management Option screen.

11. Select a data storage option on the Specify Database File Storage Option screen.

Enter the full path of the location for the data files destination on the file system and click **Next**. Oracle Universal Installer displays the Specify Backup and Recovery Options screen.

12. If you enable backup on the Specify Backup and Recovery Options screen, then you can select File System. You must also enter the user name and password.
13. On the Specify Database Schema Passwords screen, you can select different passwords for the `SYS`, `SYSTEM`, `DBSNMP`, and `SYSMAN` accounts, or you can select one password for all of the privileged accounts. When you click **Next** on the Specify Database Schema Passwords screen, the next screen that Oracle Universal Installer displays depends on the configuration and selection options as follows:
 - If the user does not have `ORA_DBA` right identifier, then the next screen is the Privileged Operating System Groups screen.
 - If the user has `ORA_DBA` right identifier, then the next screen is the Summary screen.

14. On the Privileged Operating System Groups screen, you enter the group name for the `SYSDBA` and `SYSOPER` users. When you click **Next** on the Privileged Operating System Groups screen, Oracle Universal Installer displays the Create Database screen. Finally, Oracle Universal Installer proceeds to the Summary screen.

15. The Summary screen displays the software components that Oracle Universal Installer will install and the space available in the Oracle home with a list of the nodes that are part of the installation session. Verify the installation details that are displayed on the Summary screen, and then click **Install** to continue with the installation or click **Back** to revise the installation.

16. At the end of the software install, the Oracle Universal Installer will automatically run the following two configuration assistants to complete the post-installation processing:

-
- Oracle Network Configuration Assistant
 - Database Configuration Assistant

With this you complete the second and final phase of the installation. Proceed to [Section , "Postinstallation Procedures for Oracle Real Application Clusters"](#) to perform the postinstallation tasks.

Note:

- If you need to change the virtual IP address on a Oracle Real Application Clusters node, then you should use the following command:

```
SRVCTL MODIFY NODEAPPS -A new_address
```

where *new_address* is defined in Appendix B of the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* .

- Use Database Configuration Assistant and Oracle Universal Installer to uninstall Oracle Real Application Clusters.
-

Postinstallation Procedures for Oracle Real Application Clusters

The following sections describe how to complete the postinstallation tasks after you have installed Oracle Database with Oracle Real Application Clusters software.

Note: This chapter only describes postinstallation tasks for basic configurations. Refer to the *Oracle Database Administrator's Reference* , and the product administration and tuning guides for more detailed configuration and tuning information.

You must perform the following tasks after completing the installation:

- [Section , "Backing Up the Voting Disk After Installation"](#)
- [Section , "Downloading and Installing Patches"](#)
- [Section , "Configuring Oracle Products"](#)
- [Section , "Verifying Enterprise Manager Operations"](#)
- [Section , "Setting Up Users Accounts After Installation"](#)

Backing Up the Voting Disk After Installation

After the Oracle Database with Oracle Real Application Clusters installation is complete and after you are sure that the system is functioning properly, make a backup of the contents of the voting disk. Also make a backup of the voting disk contents after you complete any node additions or node deletions and after running any deinstallation procedures.

Downloading and Installing Patches

Refer to the *OracleMetaLink* Web site for required patches for the installation. To download required patches:

1. Use a Web browser to view the *OracleMetaLink* Web site:

<https://metalink.oracle.com>

2. Log on to Oracle*MetaLink*.

Note: If you are not an Oracle*MetaLink* registered user, then click **Register for MetaLink** and register.

3. On the main Oracle*MetaLink* screen, click **Patches**.
4. On the Select a Patch Search Area screen, click **New MetaLink Patch Search**.
5. On the Simple Search screen, click **Advanced**.
6. On the Advanced Search screen, click the search icon next to the Product or Product Family field.
7. In the **Search and Select: Product Family** field, enter RDBMS Server in the **For** field and click **Go**.
8. Select **RDBMS Server** under the Results heading, and click **Select**.
RDBMS Server is displayed in the Product or Product Family field and the current release is displayed in the Release field.
9. Select the platform from the list in the **Platform** field and click **Go**.
10. All available patches are displayed under the Results heading. Click the number of the patch that you want to download.
11. On the Patch Set screen, click **View README** and read the contents of the screen that is displayed. The README screen contains information about the patch set and how to apply the patches to the installation.
12. Return to the Patch Set screen, click **Download**, and save the file on the system.
13. Use the unzip utility provided with Oracle Database 10g to uncompress the Oracle patches that you downloaded from Oracle*MetaLink*.

Note: If you have downloaded the patch to a PC, you must copy or ftp the file from the PC to OpenVMS in BINARY mode before unzipping.

Configuring Oracle Products

Many Oracle products and options must be configured before you use them for the first time. Before using individual Oracle Database 10g products or options, refer to the manual in the product documentation library which is available on the documentation installation media or on the Oracle Technology Network Web site at

<http://www.oracle.com/technology>

Verifying Enterprise Manager Operations

On all systems, you must run the following command to verify the Enterprise Manager configuration in the newly installed Oracle Real Application Clusters environment:

```
$SRVCTL CONFIG DATABASE -D db_name
```

This command (SRVCTL) displays the name of the node and the instance for the node. The following example shows a node named `db1-server` running an instance named `db1`. Run the following command:

```
$SRVCTL CONFIG DATABASE "-d" DB
```

The output should be similar to:

```
db1-server db1 /private/system/db
db2-server db2 /private/system/db
```

Setting Up Users Accounts After Installation

Oracle recommends that you complete this task after installing Oracle Real Application Clusters.

See Also: For information about setting up additional optional user accounts, refer to the *Oracle Database Administrator's Reference*

When you complete these procedures, proceed with the initial configuration tasks described in [Section , "Configuring Oracle RAC"](#).

Uninstalling Oracle Real Application Clusters Software

A successful uninstallation of Oracle software from a cluster requires you to run Oracle Universal Installer from the same node that was used for the initial installation. This is required for the Oracle Real Application Clusters and for the Oracle Clusterware software uninstallations. To determine the node used for the installation, perform the following steps:

1. Navigate to the Oracle home directory if you are uninstalling Oracle Real Application Clusters software or to the Oracle Clusterware home directory if you are uninstalling Oracle Clusterware software.
2. Start Oracle Universal Installer as follows:

```
$ @ORA_ROOT:[OUI.BIN]RUNINSTALLER.COM
```
3. Click **Installed Products...**
4. Expand the navigation tree for the Oracle home or the Oracle Clusterware home you wish to uninstall.
5. In the cluster node tree that lists the nodes on which this home is installed, the first node listed is the node on which the install was performed. This is the node where you should run Oracle Universal Installer in the following procedures.

Once you have identified the node where the software was installed, perform the following procedures to uninstall Oracle Real Application Clusters and Oracle Clusterware software. You must uninstall Oracle Database software first before uninstalling the Oracle Clusterware software. The steps to complete these uninstallation tasks are described in the following sections:

- [Section , "Procedure for Removing Oracle Real Application Clusters Software"](#)
- [Section , "Uninstalling Oracle Clusterware"](#)

Note: The following sections describe a complete uninstallation of the Oracle Real Application Clusters, and Oracle Clusterware software.

If you have multiple Oracle homes on the cluster, then check for any dependencies that may affect the other databases. Such dependencies can include listeners that run in the Oracle home to be deleted. To identify dependencies, review the ORATAB file to identify common Oracle homes.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about using Oracle Real Application Clusters scalability features, adding and deleting nodes and instances from Oracle Real Application Clusters databases, and viewing OCR content

Procedure for Removing Oracle Real Application Clusters Software

This section describes the procedure to uninstall Oracle Real Application Clusters. Before you perform these steps, consider making a backup of any databases that run from the Oracle home you are about to delete. You should then stop any instances and processes on all nodes, that depend on the software being uninstalled.

1. Delete all of the databases that are dependent on the Oracle home that you are deleting by using Database Configuration Assistant **Delete a database** option.
2. If the listener runs from this Oracle home, then use the Oracle Network Configuration Assistant to remove the listener and its configuration.
3. If the Oracle Clusterware node applications for virtual IP address, ONS, and GSD were created in this Oracle home, then you can remove them and re-create them in another Oracle Database 10g Oracle home, or you can modify the Oracle home for these applications so that they use an alternate Oracle Database 10g Oracle home. Both of these alternative procedures are described in the following two points.

Perform *one* of the following procedures:

- You can stop and remove the Oracle Clusterware node applications on each node that is associated with the Oracle home by running the following command:

```
$ srvctl stop nodeapps -n node_name
```

Do this, for all of the nodes that are affected by the deletion of the Oracle home. Then remove the Oracle Clusterware node applications by running the following command:

```
$ @ORA_ROOT:[INSTALL]ROOTDELETENODE.COM
```

Respond to any operating system prompts to confirm your operations for each node. If other Oracle Real Application Clusters Oracle homes exist, then re-create the node applications in that Oracle home by running the following command:

```
$srvctl create nodeapps
```

- Alternatively, you can modify the Oracle Clusterware node application Oracle home by running the following command:

```
$ SRVCTL MODIFY NODEAPPS "-o" oracle_home
```

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about using Oracle Real Application Clusters scalability features of adding and deleting nodes and instances from Oracle Real Application Clusters databases and for details of the SRVCTL command syntax

4. After you complete the previous steps in this procedure, run Oracle Universal Installer and on the Welcome screen, click **Deinstall Products** to display the list of installed products on which you can select the Oracle home to uninstall.

Note: You cannot perform a Oracle Real Application Clusters installation from the same Oracle Universal Installer session in which you perform a Oracle Real Application Clusters uninstallation. In other words, if you uninstall Oracle Real Application Clusters with Oracle Universal Installer and want to perform another Oracle Real Application Clusters installation, then you must start a new Oracle Universal Installer session.

Uninstalling Oracle Clusterware

Uninstall each Oracle Real Application Clusters home by running the procedure in [Section , "Procedure for Removing Oracle Real Application Clusters Software"](#). Then complete the procedure by removing the Oracle Clusterware software.

To uninstall the Oracle Clusterware software from an OpenVMS environment:

1. To disable the Oracle Clusterware applications that are running on the cluster node, run the following command from Oracle Clusterware home:

```
$ @ORA_ROOT:[INSTALL]ROOTDELETE
```

The ROOTDELETE.COM procedure requires three arguments. If you are running this command on a remote node of the cluster, then use REMOTE as the first argument. Otherwise, use LOCAL as the first argument. For the second argument, if the OCR.LOC file is on a shared file system, then use SHAREDVAR, otherwise use NOSHAREDVAR as the second argument. If the Oracle Clusterware home is on a shared file system, then use SHAREDHOME, otherwise use NOSHAREDHOME as the third argument. Repeat this step on each node of the cluster from which you want to uninstall Oracle Clusterware.

2. Run the following procedure on a local node from the Oracle Clusterware home to remove the OCR:

```
$ @ORA_ROOT:[INSTALL]ROOTDEINSTALL
```

3. Run Oracle Universal Installer and in the Welcome screen, click **Deinstall Products** to display the list of installed products on which you can select the Oracle Clusterware home to uninstall.

Note: A node is a local node if you plan to run Step 2 and Step 3 on that node.

Configuring Oracle RAC

This chapter provides an overview of the procedures for configuring Oracle RAC OpenVMS.

You may also need to refer to the following Oracle RAC documents for more complete information about installing and administering Oracle RAC on OpenVMS:

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

This chapter includes the following topics:

- [Section , "Creating Directories"](#)
- [Section , "Using Database Configuration Assistant for Oracle RAC"](#)
- [Section , "Benefits of Using Database Configuration Assistant"](#)
- [Section , "Oracle RAC High-Availability Services"](#)
- [Section , "Creating the Database After Installation"](#)
- [Section , "Creating the Oracle RAC Database"](#)
- [Section , "Deleting an Oracle RAC Database"](#)
- [Section , "Configuring Oracle RAC"](#)
- [Section , "Parameter Files and Oracle RAC"](#)
- [Section , "Using Server Parameter Files in Oracle RAC"](#)
- [Section , "Parameter File Search Order in Oracle RAC"](#)
- [Section , "Upgrading to the Server Parameter File in Oracle RAC Environments"](#)
- [Section , "Server Parameter File Errors in Oracle RAC"](#)
- [Section , "Understanding the Oracle RAC Installed Configuration"](#)
- [Section , "Database Components Created Using Database Configuration Assistant"](#)
- [Section , "Managing Undo Tablespaces in Oracle RAC"](#)
- [Section , "Configuring Service Registration-Related Parameters"](#)
- [Section , "Configuring the Listener File \(LISTENER.ORA\)"](#)
- [Section , "Directory Server Access \(LDAP.ORA File\)"](#)
- [Section , "Net Service Names \(TNSNAMES.ORA File\)"](#)

-
- [Section , "Profile \(SQLNET.ORA File\)"](#)

Creating Directories

If you decide to place the Oracle Clusterware, database, or recovery files on a file system, then use the following guidelines when deciding where to place them.

- [Section , "Guidelines for Placing Oracle Clusterware Files on a File System"](#)
- [Section , "Guidelines for Placing Oracle Database Files on a File System"](#)
- [Section , "Guidelines for Placing Oracle Database Recovery Files on a File System"](#)
- [Section , "Creating Required Directories"](#)

Guidelines for Placing Oracle Clusterware Files on a File System

Oracle Universal Installer does not suggest a default location for the Oracle Cluster Registry (OCR) or the Oracle Clusterware voting disk. If you choose to create these files on a file system, use the following guidelines when deciding where to place them:

- You must choose a disk mounted and available on all the nodes in the Oracle RAC.
- It must have at least 200 MB of free disk space for the OCR and 40 MB of free disk space for the Oracle Clusterware voting disk.
- For improved reliability, you should choose a file system on a highly available storage device, for example, a RAID device that implements mirroring.
- The Oracle Database account must have write permissions to create the files in the path that you specify.

Guidelines for Placing Oracle Database Files on a File System

If you choose to place the Oracle Database files on a file system, use the following guidelines when deciding where to place them:

- You must select a disk that is accessible across all the nodes in the Oracle RAC
- The default path suggested by Oracle Universal Installer for the database file directory is a subdirectory of the Oracle home directory. This default location is not recommended for production databases.
- Choose a set of disks dedicated to the database.

For best performance and reliability, choose a RAID device or a logical volume on more than one physical device and implement the stripe-and-mirror-everything (SAME) methodology.

This method enables you to distribute physical I/O and create separate control files on different devices for increased reliability. You must choose either the Advanced database creation option or the Custom installation type during the installation to implement this method.

- If you intend to create a preconfigured database during the installation, the file system (or file systems) that you choose must have at least 2.0 GB of free disk space.

For production databases, you must estimate the disk space requirement depending on the use that you want to make of the database.

- For optimum performance, the file systems that you choose should be on physical devices that are used only by the database.

-
- The Oracle Database account must have write permissions to create the files in the path that you specify.

Guidelines for Placing Oracle Database Recovery Files on a File System

Note: You must choose a location for recovery files only if you intend to enable automated backups during the installation.

If you choose to place the Oracle Database recovery files on a file system, use the following guidelines for placement of Oracle Database recovery files on the file system:

- To prevent disk failure from making both the database files and the recovery files unavailable, place the recovery files in a file system on a different physical disk from the database files.
- You should choose a device that is accessible on all the nodes of the Oracle RAC.
- The device that you choose should have at least 2 GB of free disk space.

The disk space requirement is the default disk quota configured for the flash recovery area (specified by the `DB_RECOVERY_FILE_DEST_SIZE` initialization parameter).

If you choose the Custom installation type, then you can specify a different disk quota value. After you create the database, you can also use Oracle Enterprise Manager Grid Control to specify a different value.

See Also: For more information about sizing the flash recovery area, refer to *Oracle Database Backup and Recovery Basics*

- The default path suggested by Oracle Universal Installer for the flash recovery area is a subdirectory of the Oracle home directory. You can choose this path only if you are using an Oracle Database base directory that is on a shared file system. This default location is not recommended for production databases.
- The `oracle` user must have write permissions to create the files in the path that you specify.

Creating Required Directories

Note: You must perform this procedure only if you want to place the Oracle Clusterware, database, or recovery files on a file system other than where the corresponding installations were created.

To create directories for the Oracle Clusterware, database, or recovery files on separate file systems from the installations:

1. If necessary, configure the disks that you want to use and mount them on each node.
2. Use the `$SHOW DEVICE` command to determine the free disk space on each mounted file system.
3. From the display, identify the file systems that you want to use. [Table 6–1](#) lists the disk space requirements for each file type.

Table 6–1 File Types and Their Disk Space Requirements

File Type	File System Requirements
Oracle Clusterware files	Choose a file system with at least 300 MB of free disk space
Database files	Choose either: <ul style="list-style-type: none">■ A single file system with at least 2.0 GB of free disk space■ Two or more file systems with at least 2.0 GB of free disk space in total
Recovery files	Choose a file system with at least 2.5 GB of free disk space.

If you are using the same disk for more than one type of file, then add the disk space requirements for each type to determine the total disk space requirement.

4. Note the names of the disks that you identified.
5. Enter commands similar to the following to create the recommended directories on each of these disks.
 - Oracle Clusterware file directory

```
$ CREATE/DIR/OWN=Oracle_Database_account device:[ORACRS]
```
 - Database file directory

```
$ CREATE/DIR/OWN=Oracle_Database_account device:[ORADATA]
```
 - Recovery file directory (flash recovery area)

```
$ CREATE/DIR/OWN=Oracle_Database_account device:[FLASH_RECOVERY_AREA]
```

Using Database Configuration Assistant for Oracle RAC

The primary functions of Database Configuration Assistant processing include:

- Creating the database and its instances
- Setting up network configuration for the database, its instances, and database services
- Starting the database, its instances, services, and any other node applications

See Also:

- [Section , "Creating the Database After Installation"](#) for more information about using Database Configuration Assistant in standalone mode
- *Oracle Database Net Services Administrator's Guide* if you experience problems (for example, with the listener configuration), and for more information about Lightweight Directory Access Protocol (LDAP)-compliant directory support

Benefits of Using Database Configuration Assistant

Oracle recommends that you use Database Configuration Assistant to create a Oracle RAC database, because Database Configuration Assistant preconfigured databases optimize the environment for features such as the server parameter file and automatic undo management.

By using Database Configuration Assistant, you can create site-specific tablespaces as a part of database creation. If you have data file requirements that differ from those offered by Database Configuration Assistant templates, then create the database with Database Configuration Assistant and modify the data files later. You can also run user-specified scripts as part of the database creation process.

Database Configuration Assistant also configures the Oracle RAC environment for various Oracle Database high-availability features such as services and cluster administration tools. It also starts any database instances required to support the defined configuration.

Oracle RAC High-Availability Services

When you configure high-availability services with Database Configuration Assistant's Database Services screen, you can also configure service instance preferences and transparent application failover (TAF) policies. These are described in the following subsections:

- [Section , "Service Configuration and Instance Preferences"](#)
- [Section , "Transparent Application Failover Policies"](#)

Service Configuration and Instance Preferences

Use the Database Services screen button in the column labeled Not Used, Preferred, or Available to configure service instance preferences as described in the following list:

- Preferred: The service runs primarily on the selected instance
- Available: The service may run on the instance if a preferred instance fails
- Not Used: The service never runs on the instance

Transparent Application Failover Policies

Use Database Configuration Assistant's Database Services screen to configure TAF failover policies. Database Configuration Assistant's Database Services screen also has a TAF policy selector row under the instance preference display. Make a selection in this row for your failover and reconnection policy preference as described in the following list:

- None: Do not use TAF
- Basic: Establish connections at failover time
- Preconnect: Establish one connection to a preferred instance and another connection to a backup instance that you have selected to be available

Creating the Database After Installation

To create a database by using Database Configuration Assistant in standalone mode, you must run the Oracle Net Configuration Assistant to configure the Oracle Net LISTENER.ORA file. To start Database Configuration Assistant, connect to one of the nodes with Oracle RAC installed and then enter the \$DBCA command from the command line.

Creating the Oracle RAC Database

The first screen that Database Configuration Assistant displays is the Welcome screen for Oracle RAC. Database Configuration Assistant displays this Oracle RAC-specific Welcome screen only if the Oracle Database home from which it is started is cluster installed.

If Database Configuration Assistant does not display the Welcome screen for Oracle RAC, then Database Configuration Assistant was not able to detect whether or not the Oracle home is cluster installed. In this case, verify that `oraInventory` is not damaged and that Oracle Universal Installer inventory, `ORAINST.LOC`, is present. By default, this file is placed in a directory that has the same name as the node from where the installation was performed, which, in most cases, is the primary node of the Oracle RAC cluster. This directory is in the login directory of the user who carried out the installation. Otherwise, perform the following steps to create a Oracle RAC database:

1. Select RAC database, click **Next**, and Database Configuration Assistant displays the Operations screen. Database Configuration Assistant enables the Configure Database Options, Delete a database, Instance Management, and Services Management options only if there is at least one Oracle RAC database configured on the cluster that runs from the Oracle Database home.
2. Select **Create a database** and click **Next**, and Database Configuration Assistant displays the Node Selection screen.
3. Database Configuration Assistant highlights the local node by default. Select the other nodes that you want to configure as members of the cluster database, click **Next**, and Database Configuration Assistant displays the Database Templates screen. If nodes that are part of the cluster installation are not displayed on the Node Selection screen, then perform inventory diagnostics and Oracle Clusterware diagnostics by running the `$OLSNODES` command.
4. The templates on the Database Templates screen are Custom Database, Transaction Processing, Data Warehouse, and General Purpose. The Custom Database template does not include data files or options specially configured for a particular type of application. Use one of the other templates, which include data files, if you want to create a database with specifically configured options. Select the template from which you wish to create the cluster database, click **Next**, and Database Configuration Assistant displays the Database Identification screen.
5. Enter the global database name and the Oracle Database system identifier (*SID*) prefix for the cluster database, click **Next**, and Database Configuration Assistant displays the Management Options screen.

Note: The global database name can be up to 30 characters in length and must begin with an alphabetic character. The *SID* prefix must begin with an alphabetic character and contain no more than 5 characters on OpenVMS-based systems. Database Configuration Assistant uses the *SID* prefix to generate a unique value for the `ORACLE_SID` for each instance.

6. On the Management Options screen, you can choose to manage the database with Enterprise Manager. Despite choosing the Grid Control option, this will only configure EM Agent on OpenVMS. The Console component of Grid Control is not supported on OpenVMS platform. Refer to [Appendix D, "Installing, Configuring, and Running EM Agent"](#) for more information about how to configure the

EMAgent for use with the Grid Control Enterprise Manager, and how to manage and maintain the EMAgent for the OpenVMS platform.

Click **Next**, and Database Configuration Assistant displays the Database Credentials screen.

7. Enter the passwords for the database on the Database Credentials screen. You can enter the same or different passwords for the users `SYS` and `SYSTEM`, plus `DBSNMP` and `SYSMAN` if you selected Enterprise Manager on the Management Options screen. Select the **Use the Same Password for All Accounts** option to assign the same password to the listed users. Alternatively, provide a different password for each of these users by selecting the **Use Different Passwords** option. Enter the password information, click **Next**, and Database Configuration Assistant displays the Storage Options screen.
8. Use the Storage Options screen to select a storage type for database creation. The Cluster File System option is the default. Select a storage option and click **Next** to proceed to the next screen. If you select Cluster File System, the Database File Locations screen is displayed.
9. The Database File Locations screen enables you to select the file storage for the database files: locations provided in a template, a common location for all database files (the files will not be Oracle Database-managed files), or Oracle Database-managed files in a common location. If you do not select the template option, then you can enter an existing directory path name in the space provided, or click **Browse** to open a selection list.

If you wish to multiplex the database redo log files and control files, click **Multiplex Redo Logs and Control Files** and provide the location for each copy you want. Click **Ok** when you have defined the multiplex locations to return to the Database File Locations screen.

You can also define variables for the file locations if you plan to use the Database Storage screen, explained in Step 14, to define individual file locations.

10. On the Recovery Configuration screen, you can select redo log archiving by selecting **Enable Archiving**. You can also select the flash recovery area and size on the Recovery Configuration screen. The flash recovery area defaults to `ORA_ROOT: [FLASH_RECOVERY_AREA]`.

You can also define variables for the file locations if you plan to use the Database Storage screen, explained in Step 14, to define individual file locations. When you have completed the entries, click **Next**, and the Database Content screen is displayed.

11. On the Database Content screen, if you chose the Custom Database option, you can select or deselect the database components and their corresponding tablespace assignment. For a seed database, you can select whether to include the sample schemas in the database and to run custom scripts as part of the database creation processing. After completing the selections, click **Next** to display the Databases Services screen.
12. To create services on the Database Services screen, expand the **Services** tree. Oracle Database displays the global database name in the top left corner of the screen. Select the global database name and click **Add** to add services to the database. Enter a service name in the **Add a Service** dialog box, and click **OK** to add the service and return to the Database Services screen.

The service name is displayed under the global database name. Select the service name and Database Configuration Assistant displays the service preferences for the service on the right side of Database Configuration Assistant Database Services

screen. Change the instance preference (Not Used, Preferred, or Available) and transparent application failover (TAF) policies for the service as needed.

Repeat this procedure for each service and when you are done configuring services for the database, click **Next**. Database Configuration Assistant displays the Initialization Parameters screen.

13. By default, the Initialization Parameters screen shows only the basic parameters. Each tab on the Initialization Parameters screen provides different sets of information that you can add or modify as follows:

- a. Memory Tab

Click **Typical** for default values based on the database type you selected or **Custom** to set values for the memory parameters. You can also see values for the advanced parameters by clicking **All Initialization Parameters**.

Carefully review the parameter settings displayed in this dialog box because Database Configuration Assistant configures these settings in the server parameter file. Instance-specific parameter settings for the Oracle RAC database are displayed at the bottom of this dialog box. The `sid` prefixes for these entries are displayed in the left column.

To review the instance-specific parameter settings, scroll downward using the scroll bar on the right side of the dialog box. Use the check box in the **Override Default** column to indicate that Database Configuration Assistant should place the parameter setting in the server parameter file. Database Configuration Assistant only places a parameter entry into the server parameter file if the entry displays a check mark in the **Override Default** column of the **All Initialization Parameters** dialog box.

Note:

- You cannot modify the value of `sid` in the Instance column
 - You can alter self-tuning parameters with this dialog box. However, setting these parameters to inappropriate values may disable Oracle Database self-tuning features.
 - You cannot specify instance-specific values for global parameters with Database Configuration Assistant.
 - You should set the value of the `CLUSTER_DB_INSTANCES` parameter to the number of instances you intend to use in the cluster if you are not including all the related nodes during the current Database Configuration Assistant session.
 - If the global database name is longer than eight characters, then the database name value (in the `DB_NAME` parameter) is truncated to the first eight characters and the `DB_UNIQUE_NAME` parameter value is set to the global name.
-
-

- b. Sizing Tab

Use this screen to select the database standard block size and process count.

- c. Character Sets Tab

Use this screen to set the database character set value.

- d. Connection Mode Tab

You can use this tab to select either dedicated or shared database connections to the database.

When you have specified all the information on the Initialization Parameters screen, click **Next**, and the Database Storage screen is displayed.

14. If you selected a preconfigured database template, such as the General Purpose template, then Database Configuration Assistant displays the control files, data files, and redo logs on the Database Storage screen. Select the folder and the file name underneath the folder to edit the file name. However, if you selected the Custom Database template, the template without data files, then Database Configuration Assistant displays the control files, tablespaces, data files, and redo logs. To change the tablespace properties, such as the data file or the tablespace size, click the tablespaces icon to expand the object tree on the left side of the screen and click the tablespace. The tablespace property dialog box is displayed on the right side. Make the changes, and click **Ok**.

After you specify all the information on the Database Storage screen, click **Next**, and Database Configuration Assistant displays the Creation Options screen.

15. On the Creation Options screen, select one of the following database options and click **Finish**.
 - **Create Database:** Creates the database
 - **Save as a Database Template:** Creates a template that records the database structure, including user-supplied inputs, initialization parameters, and so on. You can later use this template to create a database.
 - **Generate Database Creation Scripts:** Generates database creation scripts. Database Configuration Assistant only displays this option if you selected the Custom Database template.

After you click **Finish**, Database Configuration Assistant displays a Summary dialog box.

16. Review the Summary dialog box information and click **Ok** to create the database.

After you complete Step 16, Database Configuration Assistant performs the following steps:

- Creates an operative Oracle RAC database and its instances
- Creates the Oracle RAC data dictionary views
- Configures the network for the cluster database
- Starts the listeners and database instances and then starts the high-availability services

Deleting an Oracle RAC Database

This section explains how to delete a Oracle RAC database by using Database Configuration Assistant. This process deletes a database and removes the initialization parameter files, instances, and Oracle net configuration of the database.

To delete a database by using Database Configuration Assistant:

1. Start Database Configuration Assistant on one of the nodes by entering the DBCA command at the DCL command prompt.
2. Select Oracle RAC and click **Next**.

After you click **Next**, Database Configuration Assistant displays the Operations screen.

3. Select **Delete a database**, click **Next**, and Database Configuration Assistant displays the List of Cluster Databases screen.
4. If your user ID and password are not authenticated by the operating system, then the List of Cluster Databases screen displays the user name and password fields. If these fields are displayed, then enter a user ID and password that has SYSDBA privileges.
5. Select the database to delete and click **Finish**.

After you click **Finish**, Database Configuration Assistant displays a dialog box to confirm the database and instances that Database Configuration Assistant is going to delete.

6. Click **Ok** to begin the deletion of the database and its associated files, services, and environment settings, or click **Cancel** to stop the operation.

When you click **Ok**, Database Configuration Assistant continues the operation and deletes all of the associated instances for this database. Database Configuration Assistant also removes the parameter files, password files, and oratab entries.

At this point, you have accomplished the following:

- Deleted the selected database from the cluster
- Deleted high-availability services that were assigned to the database
- Deleted the Oracle net configuration for the database
- Deleted the data files

Configuring Oracle RAC

The following sections describe how to use the server parameter file (SPFILE) in Oracle RAC as well as how to configure Oracle RAC once it has been installed.

See Also: The *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about parameters and the *Oracle Real Application Clusters Deployment and Performance Guide* for a discussion of parallel execution-related parameters in Oracle RAC data warehouse environments

Parameter Files and Oracle RAC

Oracle Database uses parameter settings in parameter files to determine how to control various database resources. You can use two types of files for parameter administration: the server parameter file (SPFILE) or one or more traditional client-side parameter files.

Oracle recommends that you administer parameters using SPFILE. If you use client-side parameter files, then Oracle Database does not preserve parameter changes made for self-tuning after shutdown.

See Also: The *Oracle Database 10g Real Application Clusters* documentation for more information about using client-side parameter files

Using Server Parameter Files in Oracle RAC

By default, Oracle Database creates the server parameter file based on one SPFILE. You can change parameter settings in the server parameter file only by using Oracle Enterprise Manager or `ALTER SYSTEM SET SQL` statements; the server parameter file is a binary file that you should not edit.

Note: Oracle recommends that you avoid modifying the values for self-tuning parameters. Overriding these settings can adversely affect performance.

If you are upgrading from a previous Oracle Database release, then create and configure the server parameter file for Oracle RAC using the procedures described in the following section.

Location of the Server Parameter File

The default location of the server parameter file on a OpenVMS-based system is:

```
ORA_ROOT:[ORADATA.db_name]SPFILEsid.ORA
```

Oracle recommends that you use a PFILE in this directory:

```
ORA_ROOT:[DBS]INIToracle_sid.ora
```

To use Database Configuration Assistant to create the database and to use the server parameter file, on the Initialization Parameters screen select **Create server parameter file (spfile)** under the File Locations tab. Then enter a shared file system file name in the `Server Parameters Filename` field.

Note: When you use Database Configuration Assistant to create the server parameter file, the default PFILE file name is `ORA_ROOT:[DBS]INIToracle_sid.ora` on OpenVMS-based systems. This is the default PFILE name.

Parameter File Search Order in Oracle RAC

Oracle Database searches for the parameter file in the following order:

1. `ORA_ROOT:[ORADATA.db_name]SPFILEsid.ORA`
2. `ORA_ROOT:[ORADATA.db_name]SPFILE.ORA`
3. `ORA_ROOT:[DBS]INITsid.ora`

Upgrading to the Server Parameter File in Oracle RAC Environments

Upgrade to the server parameter file by creating and editing the server parameter file using the procedures described in this section.

- [Section , "Server Parameter File Placement in Oracle RAC"](#)
- [Section , "Procedures for Upgrading to the Server Parameter File"](#)

Server Parameter File Placement in Oracle RAC

For single-node cluster-enabled configurations, or if you are using a cluster file system, place the server parameter file on a file system.

Procedures for Upgrading to the Server Parameter File

Upgrade to the server parameter file by completing the following procedures:

1. Combine the initialization parameter files for all instances into one `INITdbname.ORA` file by copying all shared `IFILE` contents as is. All parameters defined in the `IFILE` parameter files are global. Therefore, create them as `parameter=value` without `sid` prefixes.

2. Copy all instance-specific parameter definitions from `INITsid.ORA` files using the following syntax where `sid` is the `sid` of the instance:

```
sid.parameter=value
```

3. Create the server parameter file using the `CREATE SPFILE` statement as in the following example:

```
CREATE SPFILE='ORA_DB:SPFILEsid.ORA'  
FROM PFILE='ORA_ROOT:[DBS]INITsid.ORA'
```

4. Oracle recommends that you use the server parameter file by running the `STARTUP` command as in this example:

```
STARTUP PFILE=ORA_ROOT:[DBS]INITsid.ORA
```

If you use this `STARTUP` command syntax, then Oracle Database uses the server parameter file entry specified in `INITsid.ORA`.

Server Parameter File Errors in Oracle RAC

Oracle Database reports errors that occur during server parameter file creation or while reading the file during startup. If an error occurs during a parameter update, then Oracle Database records the error in the `ALERT.LOG` file and ignores subsequent parameter updates to the file. If this happens, then do either of the following:

- Shut down the instance, recover the server parameter file, and restart the instance.
- Enable the instance to continue running without regard for subsequent parameter updates.

Oracle Database displays errors for parameter changes that you attempt when you incorrectly use the `ALTER SYSTEM SET` statement. Oracle Database does this when an error occurs while reading from or writing to the server parameter file.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about backing up the server parameter file

Understanding the Oracle RAC Installed Configuration

The following sections describe the Oracle RAC installed configuration:

- [Section , "Understanding the Configured Environment in Oracle RAC"](#)
- [Section , "Oracle Cluster Registry in Oracle RAC"](#)

-
- [Section , "OpenVMS oratab Configurations for Oracle RAC"](#)

Understanding the Configured Environment in Oracle RAC

The Oracle Net Configuration Assistant and Database Configuration Assistant configure the environment to meet the requirements for database creation and Enterprise Manager discovery of RAC databases.

Note: Configuration files are created on each node in the cluster database.

Oracle Cluster Registry in Oracle RAC

Database Configuration Assistant uses the Oracle Cluster Registry (OCR) for storing the configurations for the cluster databases that it creates. Oracle Universal Installer automatically initializes the OCR during the Oracle Clusterware installation.

OpenVMS oratab Configurations for Oracle RAC

Oracle Database creates an entry for each Oracle RAC database in the ORATAB configuration file. Oracle Enterprise Manager uses this file during service discovery to determine the name of the Oracle RAC database. The database entry has the following syntax:

```
db_unique_name:ORACLE_HOME:N
```

where *db_unique_name* is the database name for the Oracle RAC database, *ORACLE_HOME* is the UNIX-style directory path to the database, and *N* indicates that the database should not be started at startup time. A sample entry for a database named *db_name* is:

```
db_name:/DISK$PROD/ORACLE10G/10202:N
```

Note:

- The notation *db_name* displayed in the preceding example and throughout this chapter refers to the database name you enter when prompted by Database Configuration Assistant, or it refers to the entry you made for the *DATABASE* keyword of the *CREATE DATABASE* statement.
 - The N and Y terminators to a ORATAB entry are not used on OpenVMS. Automatic restart via this mechanism is not supported.
-
-

Database Components Created Using Database Configuration Assistant

This section describes the database components that Database Configuration Assistant creates, which include:

- [Section , "Tablespaces and Data Files"](#)
- [Section , "Control Files"](#)
- [Section , "Redo Log Files"](#)

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 stored on a disk. [Table 6–2](#) shows the tablespace names used by a Oracle RAC database and the types of data they contain:

Table 6–2 Tablespace Names Used by Oracle RAC Databases

Tablespace Name	Contents
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.
SYSAUX	An auxiliary system tablespace that contains the DRSYS (contains data for OracleText), ODM (for Oracle Data Mining), TOOLS (contains Enterprise Manager tables), INDEX, and EXAMPLE tablespaces.
USERS	Consists of application data. As you create and enter data into tables, Oracle Database fills this space with your data.
TEMP	Contains temporary tables and indexes created during SQL statement processing. You may need to expand this tablespace if you are running a SQL statement that involves significant sorting, such as <code>ANALYZE COMPUTE STATISTICS</code> on a very large table, or the constructs <code>GROUP BY</code> , <code>ORDER BY</code> , or <code>DISTINCT</code> .
UNDOTBS n	These are the undo tablespaces for each instance that Database Configuration Assistant creates for automatic undo management.
RBS	If you do not use automatic undo management, then Oracle Database uses the RBS tablespace for the rollback segments.

You cannot alter these tablespace names when using the preconfigured database configuration options from Oracle Universal Installer. However, you can change the names of the tablespaces if you use the advanced database creation method.

As mentioned, each tablespace has one or more data files. The data file names created by the preconfigured database configuration options vary by operating system.

Control Files

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

Redo Log Files

Each instance is configured with at least two redo log files that are stored in the shared storage. If you selected cluster file system, then these files are shared file system files. The file names of the redo log files that are created with the preconfigured database configuration options vary by storage type.

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 undo tablespace management mode that is easier to administer than manual undo management.

See Also: *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide* for more information about managing undo tablespaces

Configuring Service Registration-Related Parameters

Two key benefits of Oracle RAC are connection load balancing and failover. Oracle RAC extends the ability of single-instance Oracle Database load balancing, where connections are distributed among local dispatchers, to the balancing of connections among all instances in a cluster database. In addition, Oracle RAC provides failover by configuring multiple listeners on multiple nodes to manage client connection requests for the same database service. Connection load balancing and failover increase availability by taking advantage of the redundant resources within a cluster database. These features, however, require cross-instance registration.

Cross-instance registration in Oracle RAC occurs when the PMON process of an instance registers with the local listener and with all other listeners. Therefore, all instances in the cluster database register with all listeners running on nodes that run instances of the cluster database. This enables all listeners to manage connections across all instances for both load balancing and failover.

Cross-instance registration requires configuring the LOCAL_LISTENER and REMOTE_LISTENER initialization parameters. The LOCAL_LISTENER parameter identifies the local listener, and the REMOTE_LISTENER parameter identifies the global list of listeners. The REMOTE_LISTENER parameter is dynamic. Oracle Database changes the setting for REMOTE_LISTENER dynamically when you reconfigure the cluster database, for example, when you add or delete instances.

By default, Database Configuration Assistant configures the environment with dedicated servers. However, if you select the Shared server option on Database Configuration Assistant, then Oracle Database configures the shared server. In this case, Oracle Database uses both dedicated and shared server processing. When shared servers are configured, the DISPATCHERS parameter is specified as in the following example:

```
DISPATCHERS=" (protocol=tcp) "
```

If the DISPATCHERS initialization parameter does not specify the LISTENER attribute as in the previous example, then the PMON process registers information for all dispatchers with the listeners specified by the LOCAL_LISTENER and REMOTE_LISTENER parameters.

However, when the LISTENER attribute is specified, the PMON process registers dispatcher information with the listeners specified by the LISTENER attribute. In this case, setting the LISTENER attribute overrides REMOTE_LISTENER settings for the specified dispatchers as in the following example:

```
DISPATCHERS=" (protocol=tcp) (listener=listeners_db_name) "
```

See Also: *Oracle Database Net Services Administrator's Guide* for more information about cross-instance registration, shared and dedicated server configurations, and connection load balancing

Configuring the Listener File (LISTENER.ORA)

You can configure two types of listeners in the LISTENER.ORA file as described under the following headings:

-
- [Section , "Local Listeners"](#)
 - [Section , "Multiple Listeners"](#)
 - [Section , "How Oracle Database Uses the Listener \(LISTENER.ORA\)"](#)

Local Listeners

If you configured dedicated server mode by using Database Configuration Assistant Connection Mode tab on the Initialization Parameters screen, then Database Configuration Assistant automatically configures the `LOCAL_LISTENER` parameter when the listener uses a nondefault address port.

If you configured the dedicated server mode by setting the `REMOTE_LISTENER` initialization parameter, then you must also configure the instance-specific `LOCAL_LISTENER` initialization parameter.

For example, to configure the `LOCAL_LISTENER` parameter, add the following entry to the initialization parameter file, where `listener_sid` is resolved to a listener address through the `TNSNAMES.ORA` file located in the directory specified by the `TNS_ADMIN` logical name:

```
sid.local_listener=listener_sid
```

The following entry should be in the `TNSNAMES.ORA` file:

```
listener_sid=(address=(protocol=tcp)(host=node1-vip)(port=1522))
```

Multiple Listeners

If Database Configuration Assistant detects more than one listener on the node, it displays a list of the listeners. You can select one or all of these listeners with which to register the database.

How Oracle Database Uses the Listener (LISTENER.ORA)

Services coordinate their sessions using listener file entries by running a process on the server that receives connection requests on behalf of a client application. Listeners are configured to respond to connection requests sent to protocol addresses for a database service or nondatabase service.

Protocol addresses are configured in the listener configuration file, `LISTENER.ORA`, for a database service or a nondatabase service. Clients configured with the same addresses can connect to a service through the listener.

During a preconfigured database configuration installation, the Oracle Net Configuration Assistant creates and starts a default listener called `LISTENER_NODENAME`. The listener is configured with a default protocol listening addresses for the database and external procedures. The advanced installation process prompts you to create at least one listener with the Oracle Net Configuration Assistant. The listener is configured to respond to connection requests that are directed at one protocol address that you specify, as well as an address for external procedures.

Both installation modes configure service information about the Oracle RAC database and external procedures. A database service automatically registers its information with the listener, such as its service name, instance names, and load information. This feature, called service registration, does not require configuration in the `listener.ora` file. After listener creation, the Oracle Net Configuration Assistant starts the listener. A sample `LISTENER.ORA` file is:

```
listener_node1=
```

```
(description=
  (address=(protocol=ipc) (key=extproc))
  (address=(protocol=tcp) (host=node1-vip) (port=1521))
  (address=(protocol=tcp) (host=node1-ip) (port=1521)))
sid_list_listener_node1=
  (sid_list=
    (sid_desc=
      (sid_name=plsextproc)
      (oracle_home=disk:[oracle_home_path])
      (program=extproc))
```

Listener Registration and PMON Discovery

When a listener starts after the Oracle Database instance starts, and the listener is listed for service registration, registration does not occur until the next time the PMON discovery routine runs. By default, PMON discovery occurs every 60 seconds.

To override the 60-second delay, use the SQL `ALTER SYSTEM REGISTER` statement. This statement forces PMON to register the service immediately.

Oracle recommends that you create a script to process this statement immediately after starting the listener. If you run this statement while the listener is up and the instance is already registered, or while the listener is down, then the statement has no effect.

See Also: *Oracle Database Net Services Administrator's Guide* for more information about the listener and the `LISTENER.ORA` file

Directory Server Access (LDAP.ORA File)

If you configure access to a Lightweight Directory Access Protocol (LDAP)-compliant directory server with the Oracle Net Configuration Assistant during a Custom installation, then an `LDAP.ORA` file is created. The `LDAP.ORA` file contains the following types of information:

- Type of directory
- Location of the directory
- Administrative context from which this server can look up, create, and modify a net service name, and the database service entries

See Also: *Oracle Database Net Services Administrator's Guide* for more information about directory naming configuration and directory server access configuration

Net Service Names (TNSNAMES.ORA File)

A `TNSNAMES.ORA` file is created on each node with net service names. A connect identifier is an identifier that maps to a connect descriptor. A connect descriptor contains the following information:

- The network route to the service, including the location of the listener through a protocol address.
- The `SERVICE_NAME` for an Oracle Database release 8.1 or later.

Note: The `SERVICE_NAME` parameter you use in a connect identifier is unique because each connect identifier can map to only one `SERVICE_NAME`, even though it can map to multiple protocol addresses.

Database Configuration Assistant creates net service names for connections as follows:

Database Connections

Clients that connect to any instance of the database use the net service name entry for the database. This entry also enables Oracle Enterprise Manager to discover a Oracle RAC database.

A listener address is configured for each node that runs an instance of the database. The `LOAD_BALANCE` option causes Oracle Database to choose the address randomly. If the chosen address fails, then the `FAILOVER` option causes the connection request to fail over to the next address. Therefore, if an instance fails, then clients can still connect using another instance.

In the following example, `db.us.acme.com` is used by the client to connect to the target database, `db.us.acme.com`.

```
db.us.acme.com=
(description=
(load_balance=on)
(address=(protocol=tcp)(host=node1-vip)(port=1521)
(address=(protocol=tcp)(host=node2-vip)(port=1521)
(connect_data=
(service_name=db.us.acme.com)))
```

Note: `FAILOVER=ON` is set by default for a list of addresses. Therefore, you do not need to explicitly specify the `FAILOVER=ON` parameter.

When you set `DB_UNIQUE_NAME` by entering a global database name that is longer than eight characters, excluding `DB_DOMAIN`, then a net service entry similar to the following is created:

```
mydatabase.us.acme.com=
(description =
(address = (protocol = tcp)(host = node1-vip)(port = 1521))
(address = (protocol = tcp)(host = node2-vip)(port = 1521))
(load_balance = yes)
(connect_data =
(server = dedicated)
(service_name = mydatabase.us.acme.com)
)
)
```

Instance Connections

Clients that connect to a particular instance of the database use the net service name entry for the instance. This entry, for example, enables Oracle Enterprise Manager to discover the instances in the cluster. These entries are also used to start and stop instances.

In the following example, `db1.us.acme.com`, is used by Oracle Enterprise Manager to connect to an instance named `db1` on `db1-server`:

```
db1.us.acme.com=  
(description=  
  (address=(protocol=tcp)(host=node1-vip)(port=1521))  
  (connect_data=  
    (service_name=db.us.acme.com)  
    (instance_name=db1)))
```

Remote Listeners

As discussed in [Section , "Configuring Service Registration-Related Parameters"](#), the `REMOTE_LISTENER` parameter identifies the global list of listeners and it is dynamic. Oracle Database changes the setting for `REMOTE_LISTENER` when you reconfigure the cluster database.

Whether using shared servers or dedicated servers, the list of remote listeners is supplied using the `REMOTE_LISTENERS` parameter, for example:

```
REMOTE_LISTENERS=listeners_db_unique_name
```

This enables the instance to register with remote listeners on the other nodes; `listeners_db_unique_name` is resolved through a naming method such as a `TNSNAMES.ORA` file.

In the following example, `listeners_db.us.acme.com` is resolved to a list of listeners available on the nodes on which the cluster database has instances:

```
listeners_db.us.acme.com=  
(address_list=  
  (address=(protocol=tcp)(host=node1-vip)(port=1521))  
  (address=(protocol=tcp)(host=node2-vip)(port=1521)))
```

The instance uses this list to determine the addresses of the remote listeners with which to register its information.

Nondefault Listeners

As discussed in [Section , "Local Listeners"](#) and [Section , "Multiple Listeners"](#), the `LOCAL_LISTENER` parameter is set in the `INITsid.ORA` file if a nondefault listener is configured. For example:

```
sid.local_listener=listener_sid
```

Where `listener_sid` is resolved to a listener address through a naming method such as a `TNSNAMES.ORA` file.

In the following example, `listener_db1.us.acme.com` is resolved to the nondefault listener address:

```
listener_db1.us.acme.com=  
(address=(protocol=tcp)(host=node1-vip)(port=1522))
```

Services Entries

When you configure high-availability services using Database Configuration Assistant Services screen, then Database Configuration Assistant creates net service entries similar to the following. The three services in the following examples, `db_svc1`, `db_svc2`, and `db_svc3`, have TAF policies of `NONE`, `BASIC` and `PRECONNECT`, respectively.

```
db_svc1.us.acme.com=  
(description =
```

```

        (address=(protocol=tcp) (host=node1-vip) (port=1521))
        (address=(protocol=tcp) (host=node2-vip) (port=1521))
        (load_balance=yes)
        (connect_data=
            (server = dedicated)
            (service_name = db_svc1.us.acme.com)
        )
    )
)

db_svc2.us.acme.com=
(description=
    (address=(protocol=tcp) (host=node1-vip) (port=1521))
    (address=(protocol=tcp) (host=node2-vip) (port=1521))
    (load_balance=yes)
    (connect_data =
        (server = dedicated)
        (service_name=db_svc2.us.acme.com)
        (failover_mode =
            (type=select)
            (method=basic)
            (retries=180)
            (delay=5)
        )
    )
)
)

db_svc3.us.acme.com=
(description=
    (address=(protocol=tcp) (host=node1-vip) (port=1521))
    (address=(protocol=tcp) (host=node2-vip) (port=1521))
    (load_balance=yes)
    (connect_data=
        (server=dedicated)
        (service_name=db_svc3.us.acme.com)
        (failover_mode=
            (backup=db_svc3_preconnect.us.acme.com)
            (type=select)
            (method=preconnect)
            (retries=180)
            (delay=5)
        )
    )
)
)
)

```

When a service has a TAF policy of PRECONNECT, then a `service_name_preconnect` net service entry is also created as in the following example:

```

db_svc3_preconnect.us.acme.com =
(description =
    (address = (protocol = tcp) (host = node1-vip) (port = 1521))
    (address = (protocol = tcp) (host = node2-vip) (port = 1521))
    (load_balance = yes)
    (connect_data =
        (server = dedicated)
        (service_name = db_svc3_preconnect.us.amce.com)
        (failover_mode =
            (backup = db_svc3.us.acme.com)
            (type = select)
            (method = basic)
            (retries = 180)
            (delay = 5)
        )
    )
)
)
)

```

External Procedures

An entry for connections to external procedures. This enables an Oracle Database to connect to external procedures.

```
extproc_connection_data.us.acme.com=  
  (description=  
    (address_list=  
      (address=(protocol=ipc) (key=extproc0))  
    )  
    (connect_data=  
      (sid=plsextproc)))
```

Example 6-1 Example tnsnames.ora File

The following is a sample TNSNAMES.ORA file that is created during a preconfigured database configuration installation:

```
db.us.acme.com=  
  (description=  
    (load_balance=on)  
    (address=(protocol=tcp) (host=node1-vip) (port=1521))  
    (address=(protocol=tcp) (host=node2-vip) (port=1521))  
    (connect_data=  
      (service_name=db.us.acme.com)))  
  
db1.us.acme.com=  
  (description=  
    (address=(protocol=tcp) (host=node1-vip) (port=1521))  
    (connect_data=  
      (service_name=db.us.acme.com)  
      (instance_name=db1)))  
  
db2.us.acme.com=  
  (description=  
    (address=(protocol=tcp) (host=node2-vip) (port=1521))  
    (connect_data=  
      (service_name=db.us.acme.com)  
      (instance_name=db2)))  
  
listeners_db.us.acme.com=  
  (address_list=  
    (address=(protocol=tcp) (host=node1-vip) (port=1521))  
    (address=(protocol=tcp) (host=node2-vip) (port=1521)))  
  
extproc_connection_data.us.acme.com=  
  (description=  
    (address_list=  
      (address=(protocol=ipc) (key=extproc)))  
    )  
    (connect_data=  
      (sid=plsextproc)  
      (presentation=R0)))
```

See Also: *Oracle Database Net Services Administrator's Guide* for more information about the TNSNAMES.ORA file

Profile (SQLNET.ORA File)

The SQLNET.ORA file is automatically configured with:

- The computer domain

This domain is automatically appended to any unqualified net service name. For example, if the default domain is set to `us.acme.com`, then Oracle Database resolves `db` in the connect string `CONNECT scott/tiger@db` as: `db.us.acme.com`.

- A naming method the server uses to resolve a name to a connect descriptor

The following is a sample `SQLNET.ORA` file created during a preconfigured database configuration install:

```
names.default_domain=us.acme.com
names.directory_path=(tnsnames, ldap)
```

See Also: *Oracle Database Net Services Administrator's Guide* for more information about the `SQLNET.ORA` file

Getting Started with Oracle Database

This chapter provides information about the default preconfigured database, including information about Oracle database accounts, passwords, and file locations. It includes information about the following topics:

- [Section , "Reviewing Accounts and Passwords"](#)
- [Section , "Unlocking and Resetting User Passwords"](#)
- [Section , "Identifying Databases"](#)
- [Section , "Locating the Parameter Files"](#)
- [Section , "Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files"](#)
- [Section , "Adding or Upgrading Products"](#)
- [Section , "Relinking the Oracle Code"](#)

Reviewing Accounts and Passwords

All databases that are created by Database Configuration Assistant (DBCA) include the `SYS`, `SYSTEM`, `SYSMAN`, and `DBSNMP` database accounts. In addition, Oracle provides several other administrative accounts. Before using these accounts, you must unlock them and reset their passwords. [Table 7-1](#) describes these accounts and lists their user names and default passwords.

Note: You can use the Oracle Enterprise Manager Database Control to view the complete list of database accounts. For more information, refer to the ["Unlocking and Resetting User Passwords"](#) section on page 7-3.

Table 7-1 Database Accounts

User Name	Default Password	Description	See Also
ANONYMOUS	ANONYMOUS	Allows HTTP access to Oracle XML DB.	None
CTXSYS	CTXSYS	The Oracle Text account.	<i>Oracle Text Reference</i>
DBSNMP	Password set during installation or in the Database Configuration Assistant	The account used by the Management Agent component of Oracle Enterprise Manager to monitor and manage the database. It is created only if you configure the database to use the Database Control.	<i>Oracle Enterprise Manager Grid Control Installation and Basic Configuration</i>
DMSYS	DMSYS	The data mining account. DMSYS performs data mining operations.	<i>Oracle Data Mining Administrator's Guide</i>
HR	HR	The account that owns the Human Resources schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
LBACSYS	LBACSYS	The Oracle Label Security administrator account.	<i>Oracle Label Security Administrator's Guide</i>
MDDATA	MDDATA	The schema used by Oracle Spatial for storing Geocoder and router data.	<i>Oracle Spatial User's Guide and Reference</i>
MDSYS	MDSYS	The Oracle Spatial and Oracle <i>interMedia</i> Locator administrator account.	<i>Oracle Spatial User's Guide and Reference</i>
OE	OE	The account that owns the Order Entry schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
ORDPLUGINS	ORDPLUGINS	The Oracle <i>interMedia</i> user. Plugins supplied by Oracle and third-party plugins are installed in this schema.	<i>Oracle interMedia Reference</i>
ORDSYS	ORDSYS	The Oracle <i>interMedia</i> administrator account.	<i>Oracle interMedia Reference</i>
OUTLN	OUTLN	The account that supports plan stability. Plan stability enables you to maintain the same execution plans for the same SQL statements. OUTLN acts as a role to centrally manage metadata associated with stored outlines.	<i>Oracle Database Concepts</i>
PM	PM	The account that owns the Product Media schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
SCOTT	TIGER	An account used by Oracle sample programs and examples.	<i>Oracle Database Administrator's Guide</i>
SH	SH	The account that owns the Sales History schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas during an Enterprise Edition installation.	<i>Oracle Database Administrator's Guide</i>

Table 7–1 (Continued) Database Accounts

User Name	Default Password	Description	See Also
SYS	Password set during installation or in the Database Configuration Assistant	The account used to perform database administration tasks.	<i>Oracle Database Administrator's Guide</i>
SYSTEM	Password set during installation or in the Database Configuration Assistant	Another account used to perform database administration tasks.	<i>Oracle Database Administrator's Guide</i>
WKPROXY	WKSYS	The Ultra Search proxy user.	<i>Oracle Ultra Search Administrator's Guide</i>
XDB	XDB	The account used for storing Oracle XML DB data and metadata.	<i>Oracle XML DB Developer's Guide</i>

Unlocking and Resetting User Passwords

Passwords for all Oracle system administration accounts except SYS, SYSTEM, SYSMAN, and DBSMP are revoked after installation. Before you use a locked account, you must unlock it and reset its password. If you created a preconfigured database during the installation, but you did not unlock a required account, then you must unlock it, by using one of the following methods:

- [Section , "Using Grid Control to Unlock Accounts and Reset Passwords"](#)
- [Section , "Using SQL*Plus to Unlock Accounts and Reset Passwords"](#)

Note: If you use Database Configuration Assistant to create a new database, you can unlock accounts after the database is created by clicking **Password Management** before you exit from Database Configuration Assistant.

Using Grid Control to Unlock Accounts and Reset Passwords

To unlock and reset user account passwords using Oracle Enterprise Manager Grid Control, perform the following steps:

1. Log on to the Grid Control.
2. Click **Administration**.
3. In the Security section of the Administration screen, click **Users**.

Enterprise Manager displays a table listing all database accounts. The Account Status column indicates whether the account is locked and whether the password has expired.

4. Select the user account that you want to modify, then click **Edit**.
5. Use the General screen of the Users property sheet to unlock the account and (optionally) to change the password.

See Also: Click **Help** in the Grid Control screen for more information about using the Grid Control

Using SQL*Plus to Unlock Accounts and Reset Passwords

Perform the following steps to unlock and reset user account passwords manually:

1. Start SQL*Plus and log in as SYSDBA:

```
$ SQLPLUS/NOLOG
SQL> CONNECT / AS SYSDBA
```

2. Enter a command similar to the following, where *account* is the user account that you want to unlock and *password* is the new password:

```
SQL> ALTER USER account [ IDENTIFIED BY password ] ACCOUNT UNLOCK
```

In this example:

- The ACCOUNT UNLOCK clause unlocks the account.
- The IDENTIFIED BY *password* clause resets the password.

Note: If you unlock an account but do not reset the password, then the password remains expired. The first time you connect as that user, you must change the user account password.

To permit unauthenticated access to your data through HTTP, unlock the ANONYMOUS user account.

See Also: *Oracle Database Administrator's Guide* for more information about:

- Unlocking and changing passwords after installation
- Oracle security procedures
- Best security practices

Identifying Databases

The Oracle Database 10g software identifies a database by its global database name. A global database name consists of the database name and database domain. Usually, the database domain is the same as the network domain, but it need not be. The global database name uniquely distinguishes a database from any other database in the same network. You specify the global database name when you create a database during the installation, or using the Database Configuration Assistant. For example:

```
sales.us.oracle.com
```

In this example:

- *sales* is the name of the database. The database name is a string of maximum eight characters that can contain alphanumeric, underscore (_), and hash (#) characters. The DB_NAME initialization parameter specifies the database name.
- *us.oracle.com* is the network domain in which the database is located. The database name and the network domain together constitute the unique global database name. The domain is a string of maximum 128 characters that can contain alphanumeric, underscore (_), and hash (#) characters. The DB_DOMAIN initialization parameter specifies the domain name.

The `DB_NAME` and `DB_DOMAIN` parameters combine to create the global database name value. This value is assigned to the `SERVICE_NAMES` parameter in the initialization parameter file.

The system identifier (SID) identifies a specific database instance. The SID uniquely distinguishes the instance from any other instance on the same computer. Each database instance requires a unique SID and database name. In most cases, the SID is the same as the database name portion of the global database name.

Note: The SID must contain no more than 6 characters on OpenVMS-based systems.

Locating the Parameter Files

The database initialization parameter file is generated when a database is created. By default, the preconfigured database uses a server parameter file named `SPFILE sid .ORA`. The following subsections describe the process of locating the parameter files:

- [Section , "Locating the INIT.ORA File"](#)
- [Section , "Locating the Server Parameter File"](#)

Locating the INIT.ORA File

The database initialization parameter file (`INIT.ORA`) is generated when a database is created. The `INIT.ORA` file is used to start the database instance, and create the database and data dictionary. When the database creation is complete, the instance is shutdown, and the `INIT.ORA` file is saved to `ORA_ROOT:[DBS]INIT sid .ORA`. The logical name `ORA_PARAMS`, points to `ORA_ROOT:[DBS]INIT sid .ORA`.

The `INIT.ORA` file is a text file that contains a list of preconfigured instance configuration parameters that must exist for an instance to start. You do not need to edit this file to use the starter database.

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Reference* for more information about database-specific initialization parameters and their default values

Locating the Server Parameter File

By default, the preconfigured database uses a server parameter file named `SPFILE sid .ORA`, which is stored in the `ORA_ROOT:[DBS]` directory.

If the server parameter file is not located in this directory, then the database uses the `SPFILE` parameter in an initialization parameter file to locate it. The default file name and location of this initialization parameter file is `ORA_ROOT:[DBS]INIT sid .ORA`.

You can use the Oracle Enterprise Manager to view the location of the server parameter file and list all the initialization parameters as follows:

1. Log on to the Enterprise Manager Control.
2. Under Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.
4. In the Instance section of the Administration screen, click **All Initialization Parameters**.

Enterprise Manager displays a table listing the current value of each initialization parameter.

5. Click **SPFile**.

Enterprise Manager displays a table listing the value of each initialization parameter specified in the server parameter file. The location of the server parameter file is displayed above the table.

Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files

The following sections contain information about tablespaces and data files, redo log files, and control files:

- [Section , "Identifying Tablespaces and Data Files"](#)
- [Section , "Locating Redo Log Files"](#)
- [Section , "Locating Control Files"](#)

Identifying Tablespaces and Data Files

An Oracle Database is divided into smaller logical areas of space known as tablespaces. Each tablespace corresponds to one or more physical data files. Data files contain the contents of logical database structures such as tables and indexes. You can associate each data file with only one tablespace and database.

Note: The `SYSAUX` and `SYSTEM` tablespaces must be present in all Oracle Database 10g databases.

[Table 7–2](#) describes the tablespaces that are provided by the default preconfigured database and a description of what they store.

Table 7–2 *Tablespaces and Descriptions*

Tablespace	Description
EXAMPLE	Stores the sample schemas, if you included them.
SYSAUX	Serves as an auxiliary tablespace to the <code>SYSTEM</code> tablespace. Some products and options that previously used the <code>SYSTEM</code> tablespace now use the <code>SYSAUX</code> tablespace, which reduces the load on the <code>SYSTEM</code> tablespace.
SYSTEM	Stores the data dictionary, which includes definitions of tables, views, and stored procedures needed by Oracle Database.
TEMP	Stores temporary tables and indexes when processing SQL statements.
UNDOTBS1	Stores undo information.
USERS	Stores database objects created by database users.

See Also: *Oracle Database Concepts* and the *Oracle Database Administrator's Guide* for more information about tablespaces and data files

To use the Oracle Enterprise Manager Grid Control to view the list of data files that are used by the database and their associated tablespaces:

-
1. Log on to the Enterprise Manager Control.
 2. Under Targets tab, select the correct database target to display its home screen.
 3. Click **Administration**.
 4. In the Storage section of the Administration screen, click **Datafiles**.

Enterprise Manager displays a table listing each data file and the tablespace with which it is associated.

Note: For more information about using the Grid Control to view, modify, and create tablespaces, click **Help** in the Grid Control screen.

Locating Redo Log Files

The starter database and the custom database each contain three redo log files that are located in the `ORA_ROOT:[ORADATA.db_name]` directory. The preconfigured database uses three redo log files. These files record all the changes made to data in the database buffer cache. If an instance fails, then Oracle Database uses the redo log files to recover the modified data in memory.

Oracle Database uses redo log files in a cyclical fashion. For example, if three files constitute the online redo log, Oracle Database fills the first file, then the second file, and then the third file. In the next cycle, it reuses and fills the first file, the second file, and so on.

See Also: *Oracle Database Backup and Recovery Basics* for more information about redo log files

To use the Oracle Enterprise Manager Database Control to view or modify the redo log files for the preconfigured database, perform the following steps:

1. Log on to the Enterprise Manager Control.
2. Under the Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.
4. In the Storage section of the Administration screen, click **Redo Log Groups**.

Oracle Enterprise Manager displays a table listing the redo log groups that are used by the database.

5. To view the name and location of the redo log file associated with a particular group, select that group and then click **View**.

See Also: For more information about using the Database Control to view, modify, and create redo log files, click **Help** in the Database Control window

Locating Control Files

A control file is an administrative file. Oracle Database requires a control file to start and run the database. A control file defines the physical structure of the database. For example, it defines the database name and the names and locations of the data files and redo log files.

The starter database contains three control files. These files are located in the `ORA_ROOT:[ORADATA.db_name]` directory. The `CONTROL101.CTL`, `CONTROL102.CTL`, and `CONTROL103.CTL` files are automatically included in the custom database. A

preconfigured database uses three control files. Oracle recommends that you keep at least three control files (on separate physical drives) for each database and set the `CONTROL_FILES` initialization parameter to list each control file.

To use the Oracle Enterprise Manager Grid Control to view information about the control files for the preconfigured database, perform the following steps:

1. Log on to the Enterprise Manager Control.
2. Under Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.
4. In the Storage section of the Administration screen, click **Controlfiles**.

Enterprise Manager displays a table listing the control files used by the database.

See Also: For more information about using the Database Control to view information about control files and creating backups of these files to trace them, click **Help** in the Database Control window

For more information about setting the `CONTROL_FILES` initialization parameter value, refer to *Oracle Database Administrator's Guide*

Adding or Upgrading Products

The following section describe the steps to add or upgrade Oracle products:

- [Section , "Preparing to Add a Product"](#)
- [Section , "Adding a Product Summary"](#)

Preparing to Add a Product

Perform the following steps before you add or upgrade a product:

1. Verify that the system has Oracle Database installed.
2. Log on to the computer.
3. Run the `ORAUSER` command procedure for the database:

```
$ SET DEFAULT disk:[oracle_home_path]
$ @ORAUSER sid
```
4. If you plan to upgrade any Oracle product tables, then change the Oracle `SYSTEM` password to `MANAGER`.

Note: If you do not change the Oracle `SYSTEM` password to `MANAGER`, then the installation may fail. Oracle installation scripts often assume that the password is `MANAGER`. Alternatively, you can check for hard-coded user names and passwords in Oracle scripts and change them to the required user names and passwords.

5. Back up the old code and database files (data, redo log, and control).
6. Mount the distribution media.
7. Run Oracle Universal Installer, and select **Custom install**.

After completing these preliminary steps, you can load and configure Oracle products.

Adding a Product Summary

To add a product summary:

1. Start Oracle Universal Installer as follows:

```
$ SET DEFAULT dvd_device:[DATABASE]
$ @RUNINSTALLER.COM
```

2. Point to the existing Oracle home.
3. Select **Custom Install**.
4. Select **Products to Install**.
5. Configure the products if Oracle Universal Installer asks for any specifications.
6. Click **Install**.
7. Exit Oracle Universal Installer.
8. Start an instance.
9. Build or upgrade the database tables menu.

Relinking the Oracle Code

The Oracle code must be relinked under the following conditions:

- Change of operating system (including an upgrade)
- Changes made to Oracle Net Services configuration
- Changes made to RDBMS configuration

The following sections describe how you can relink Oracle code.

Step 1 Shutting Down the Given Database(s)

Verify that you point to the right database instance. To do this, enter the following at the DCL prompt:

```
$ SHOW LOGICAL ORA_SID
ORA_SID" = "V10G" (LNM$JOB_TABLE)
```

Start SQLPlus and connect to SYS. Check the V\$PROCESS view to see if other processes are active. If no other processes are active, and if you are pointing to the correct instance, then shut down the given database.

Oracle background processes stop after the database is shut down. To verify this, enter the SHOW SYSTEM command at the DCL prompt.

Step 2 Removing Images

To remove images, enter the following command at the system prompt:

```
$ REMORACLE
```

Note: Generally, no response is returned except for the return of the DCL prompt.

Step 3 Relinking the Oracle Code

To relink the Oracle code, enter the following command at the DCL prompt:

```
@ORA_ROOT:[BIN]RELINK product | ALL
```

Removing Oracle Software

This chapter describes how to completely remove all Oracle Databases, their instances, and related software from an Oracle home directory. It includes information about the following topics:

- [Section , "Identifying All Instances"](#)
- [Section , "Removing an Oracle Database"](#)
- [Section , "Reconfiguring Oracle Cluster Synchronization Services"](#)
- [Section , "Removing Oracle Software"](#)

Note: If you want to remove an Oracle Real Application Clusters installation, then refer to *Oracle Real Application Clusters Installation and Configuration Guide* for more information.

If you want to remove an individual product, then check the product-specific documentation for requirements and restrictions.

Identifying All Instances

To identify all instances associated with the Oracle home that you want to remove, log into the Oracle Database account and enter the following command:

```
$ SHOW LOGICAL SYS$LOGIN
```

Example output:

```
DISK$ORACLE1: [ORACLE10G]
```

The instance names are in the ORATAB file in the subdirectory for the current node:

```
$ TYPE DISK$ORACLE1: [ORACLE10G.nodename]ORATAB.
```

The output of this command contains entries similar to the following:

```
$ORACLE_SID:ORACLE_HOME:Y/N
```

Where:

SID is the system identifier of the Database.

ORACLE_HOME is the home directory of the Database.

For Example:

```
SID: /VMSDISK/ORACLE_HOME:N
```

Removing an Oracle Database

To completely remove Oracle Database software, you must remove any installed databases. To remove an Oracle Database:

Caution: Removing an Oracle Database deletes all of the data in the database. If you want to keep this data, then ensure that you back up the database before deleting it.

1. Log into the Oracle Database account.
2. Enter the following command to change the directory to the default database location:

```
$ set default oracle_home_directory
```
3. Run the ORAUSER command procedure, specifying the SID for the database that you want to remove.:

```
$ @ORAUSER sid
```
4. Use the `$ SET DISPLAY` command to direct X-Windows to your PC or workstation.
5. Start Database Configuration Assistant by using the following command:

```
$ DBCA
```

The Welcome screen is displayed.
6. Click **Next**.
The Operations screen is displayed.
7. Select **Delete a Database**, then click **Next**.
8. Select the database that you want to delete, and then click **Finish**.
9. In the screen that is displayed, confirm that you want to delete the database.
10. When Database Configuration Assistant removes the database, you are prompted to select whether you want to perform another operation. Click **Yes** to return to the Operations screen or click **No** to exit from Database Configuration Assistant. If you want to remove another database, click **Yes** and repeat Steps 6 through 8.

Reconfiguring Oracle Cluster Synchronization Services

On cluster systems with Oracle Real Application Clusters installations, the Cluster Synchronization Services daemon is configured during the Oracle Clusterware installation. If the system is running Oracle Clusterware, then refer to *Oracle Real Application Clusters Installation and Configuration Guide* for information about removing Oracle Real Application Clusters or Oracle Clusterware.

Before you remove an Oracle Database Oracle home, you must determine whether the Oracle Cluster Synchronization Services daemon is running from that Oracle home and whether any other Oracle Database Oracle homes exist on the system:

- If the Oracle Database Oracle home that you want to remove is the only Oracle Database installation on the system, then you can delete the Oracle Cluster Synchronization Services daemon configuration.

-
- If the Oracle Cluster Synchronization Services daemon is running from the Oracle Database Oracle home that you want to remove, and if other Oracle Database installations exist on the system, then you must reconfigure the Oracle Cluster Synchronization Services daemon to run from another Oracle Database Oracle home.

The following sections describe how to complete these tasks:

- [Section , "Identifying Oracle Database Oracle Homes"](#)
- [Section , "Reconfiguring the Oracle Cluster Synchronization Services Daemon"](#)
- [Section , "Deleting the Oracle Cluster Synchronization Services Daemon Configuration"](#)

Identifying Oracle Database Oracle Homes

Display the ORATAB file contents by using the following command:

```
$ TYPE DISK$ORACLE1:[ORACLE10G.nodename]ORATAB.
```

From the output, identify any Oracle home directories where Oracle Database is installed. Oracle homes that contain Oracle Database typically have paths similar to the following. However, they may use different paths.

```
/device/oracle/product/10_2_0/db_n
```

If there is only one Oracle home directory that contains an Oracle database, refer to [Section , "Deleting the Oracle Cluster Synchronization Services Daemon Configuration"](#) for information about deleting the Oracle Cluster Synchronization Services daemon configuration.

If you identify more than one Oracle Database Oracle home directory, refer to the following section for information about reconfiguring the Oracle Cluster Synchronization Services daemon.

Reconfiguring the Oracle Cluster Synchronization Services Daemon

To reconfigure the Oracle Cluster Synchronization Services daemon so that it runs from an Oracle home that you are not removing:

1. Enter the following command to identify the Oracle home directory being used to run the Oracle Cluster Synchronization Services daemon:

```
$ TYPE disk:[mylogin.nodename.ORACLE]OCR.LOC
```

The output from this command is similar to the following:

```
ocrconfig_loc= VMSDEVICE:[ANY_VMS_DIR]ocr.dat  
local_only=FALSE
```

The `ocrconfig_loc` parameter specifies the location of the Oracle Cluster Registry (OCR) used by the Oracle Cluster Synchronization Services daemon. The path is the Oracle home directory where the Oracle Cluster Synchronization Services daemon is running.

Note: If the value for the `local_only` parameter is `FALSE`, Oracle Clusterware is installed on this system. Refer to *Oracle Real Application Clusters Installation and Configuration Guide* for information about removing Oracle Real Application Clusters or Oracle Clusterware.

If this Oracle home directory is not the Oracle home that you want to remove, you can continue to [Section , "Removing Oracle Software"](#).

2. Navigate to the Oracle home directory for an Oracle Database installation that you are *not* removing.
3. Run the following command in this new Oracle home:

```
$ @ORAUSER
```

4. Enter the following command to reconfigure the Oracle Cluster Synchronization Services daemon to run from this Oracle home:

```
$ @ORA_ROOT:[BIN]LOCALCONFIG "reset" ORACLE_HOME
```

The script stops the Oracle Cluster Synchronization Services daemon, reconfigures it in the new Oracle home, and then restarts it. When the system starts, the Oracle Cluster Synchronization Services daemon starts automatically from the new Oracle home.

Deleting the Oracle Cluster Synchronization Services Daemon Configuration

To delete the Oracle Cluster Synchronization Services daemon configuration:

1. Navigate to the Oracle home directory that you are removing.
2. Run the following command in the Oracle home:

```
$ @ORAUSER
```

3. Enter the following command to delete the Oracle Cluster Synchronization Services daemon configuration from this Oracle home:

```
$ @ORA_ROOT:[BIN]LOCALCONFIG DELETE
```

The script stops the Oracle Cluster Synchronization Services daemon, then deletes its configuration. When the system restarts, the Oracle Cluster Synchronization Services daemon does not start.

Note: Delete the Oracle Cluster Synchronization Services daemon configuration only if you are certain that no other Oracle Database installation requires it.

Removing Oracle Software

The following steps describe how to use the Installer to remove Oracle software from an Oracle home:

Note: Always use Oracle Universal Installer to remove Oracle software. Do not delete any Oracle home directories without first using Oracle Universal Installer to remove the software.

1. Remove any database instances that are associated with this Oracle home, and delete or reconfigure the Oracle Cluster Synchronization Services daemon.
2. Stop any processes running in this Oracle home:

Table 8–1 Commands to Stop Processes Running in Oracle Home

Process Name	Command
Database Control	\$ EMCTL STOP AGENT
Oracle Net Services listener	\$ LSNRCTL STOP

3. If necessary, log into the Oracle Database account.
4. Run the ORAUZER.COM file to redefine the Oracle home.

```
$ @ORAUZER
```

5. Use the \$ SET DISPLAY command to direct X-Windows to your PC or workstation.
6. Start Oracle Universal Installer as follows:

```
$ @dvd_device:[DATABASE]RUNINSTALLER
```

If the product was installed from the [CLIENT] or [COMPANION] directory of the Installation Media, then use that directory instead of [DATABASE].

7. In the Welcome screen, click **Deinstall Products**.
The Inventory screen is displayed, listing all of the Oracle homes on the system.
8. In the Inventory screen, select the Oracle home and the products that you want to remove, then click **Remove**.

Caution: If you select to remove Oracle JVM, then Oracle Universal Installer removes all installed products that depend on Oracle JVM, including Oracle Database 10g.

9. When Oracle Universal Installer displays a confirmation screen asking you to confirm that you want to uninstall the products and their dependent components, click **Yes**.
The Installer displays a progress indicator as it removes the software.
10. When the products have been deleted, click **Cancel** to exit from Oracle Universal Installer, then click **Yes**.

Note: Always use Oracle Universal Installer to remove Oracle software. Do not delete any Oracle home directories without first using Oracle Universal Installer to remove the software.

Oracle Database 64-Bit Feature

This appendix introduces the 64-bit feature or Very Large Memory (VLM) and recommends the parameter settings for this feature.

The topics covered in this appendix are:

- [Section , "Introduction to Oracle Database 64-Bit Feature"](#)
- [Section , "Suggested Parameter Settings"](#)
- [Section , "Other Recommendations"](#)

Note: The 64-bit feature is a standard feature of Oracle Database for OpenVMS and cannot be removed. The 64-bit feature tuning is under constant revision. Therefore, this information will be updated as further tuning suggestions are realized.

Introduction to Oracle Database 64-Bit Feature

The Oracle 64-bit feature provides the ability to support Very Large Memory (VLM) system configurations with large amounts of RAM.

The following sections describe the benefits of the 64-bit feature and its implementation.

- [Section , "Benefits of Oracle Database 64-Bit Feature"](#)
- [Section , "Implementation of Oracle 64-Bit Feature"](#)

Benefits of Oracle Database 64-Bit Feature

The primary benefit of the 64-bit feature is performance, because many operations can now run at memory speed instead of disk speed. With larger amounts of data in

the memory, the database issues fewer calls to disk. This greatly reduces the delay of disk I/O.

Oracle Database 64-bit feature also provides the advantage of scalability to support more users and larger volumes of data. The system does not need to swap data in and out of memory to process all the transaction requests, and it can effectively accommodate requests for large amounts of data.

The 64-bit feature benefits both query-intensive decision support system (DSS) and read/write (OLTP) transactions. The 64-bit feature provides index builds, full table scans, ad hoc queries, and multiway joins for DSS. For OLTP, the feature provides the ability to support very large tables, large amounts of data, and large number of users.

Implementation of Oracle 64-Bit Feature

The OpenVMS operating system has native 64-bit memory addressing, which enables Oracle Database to implement the 64-bit feature. The need to estimate the maximum System Global Area (SGA) at installation time is now eliminated. This speeds up process startup. The server no longer includes an *SGAPAD*.

By default, SGA creation uses OpenVMS facilities to create global sections.

Suggested Parameter Settings

The recommended parameter settings for the Oracle 64-bit feature are:

1. Big Oracle Blocks (BOB) provide the ability to support larger I/O transfers between memory and disk. BOB complements large SGA configurations, because BOB enables the system to move data faster between memory and disk. With VLM configurations, system performance depends directly on the ability of the system to move database blocks into the SGA as efficiently as possible. Without the benefits of improved data transfer, performance can decline.

For a pure DSS application, select a large value (such as 32 KB) for `DB_BLOCK_SIZE`. For an OLTP type of application, choose a lower value (such as, 2 KB or 4 KB). The larger the `DB_BLOCK_SIZE`, the more serious the impact on single-row lookups.

2. Configure the size of the Oracle buffer cache such that it provides the best possible cache hit ratio without affecting memory requirements of other Oracle and system processes.

Note: You cannot combine the `DB_BLOCK_BUFFERS` with the dynamic `DB_CACHE_SIZE` parameter. If you combine these parameters in the same parameter file, then it will produce an error.

For example, for a 3 GB buffer cache with `DB_BLOCK_SIZE=8192`, set `DB_BLOCK_BUFFERS=400000`.

3. Set the `SORT_AREA_SIZE` parameter with care. `SORT_AREA_SIZE` is the space used in Program Global Area (PGA) for each sort run by each Oracle process. If the value is too high, then the PGA will use excessive memory when sorting. The default value of 512 KB is usually sufficient.

Check statistics, such as `V$SYSSTAT`, to see if the number of sorts to disk is high compared to in-memory sorts. If it is, then increase the value of `SORT_AREA_SIZE`.

Other Recommendations

Check the size of the number of rows of the tables involved in the query, and translate this size into the total number of blocks. Based on the query, try to fit as many of the frequently accessed table blocks in the buffer cache.

For example, if four tables are involved in a query, but columns from one table are used repeatedly in the `WHERE` clause in joins, `IN`, and so forth, then try to fit as many blocks as possible from this table into the cache to see if `DB_BLOCK_BUFFERS` can be increased. To ensure that the frequently accessed tables are cached and stay in the most recently used (MRU) end of the cache, perform either of the following steps:

- Enter the following command (using SQL*Plus):

```
alter table tablename cache
```

or

- At the time of table creation, enter the following:

```
create table tablename ... cache
```

If an adequate number of buffers are available to accommodate all blocks from all tables that are involved in the query, then use the `alter` command to cache all the blocks. The purpose is to cache most blocks into memory to ensure that I/O to disks is eliminated or remains low.

Noninteractive Installation and Configuration

This chapter describes how to install and configure Oracle products noninteractively using response files. It includes information about the following topics:

- [Section , "Silent Mode Installation"](#)
- [Section , "Performing Silent Installation"](#)
- [Section , "Running Oracle Universal Installer in Silent Mode or Suppressed Mode"](#)
- [Section , "Running Oracle Network Configuration Assistant in Silent Mode"](#)
- [Section , "Running Database Configuration Assistant in Noninteractive or Silent Mode"](#)

Silent Mode Installation

A silent installation with Oracle Universal Installer is noninteractive, meaning that it takes place without using the Java graphical user interface (GUI). Oracle Universal Installer gets its information from a text file that is specified at the command line by using the `-RESPONSEFILE` argument instead of interactive GUI input and output. If the `-RESPONSEFILE` flag is given, then the installation is assumed to be a silent mode type.

Note: Because the design of Oracle Universal Installer causes GUI classes to be loaded (but not displayed), a silent installation still requires an X-Windows connection.

A silent installation can also be performed in `BATCH` mode, as long as the process has established an X Windows connection through the `SET DISPLAY` command.

This section discusses the following topics:

- [Section , "What Is a Response File?"](#)
- [Section , "Why Perform a Silent Installation?"](#)
- [Section , "Modifying a Response File"](#)

What Is a Response File?

A response file contains answers to installation questions that otherwise would be provided by the user in an interactive installation session. Each answer is stored as a value for a variable that is identified in the response file. For example, values for `ORACLE_HOME` or `Install Type` can be set automatically within the response file. Response file templates are generated by the installation developer. The response file

template for the installation of the product can be found on the installation DVD under the [DATABASE.RESPONSE] directory. You should be able to find one for your choice of install type (for example, ENTERPRISE.RSP for an Enterprise Edition server installation, or CUSTOM.RSP for a custom installation).

Why Perform a Silent Installation?

Silent installations can be useful if you need to install an Oracle product multiple times on multiple computers, or if you prefer not to use the GUI because of performance reasons over a slow remote X Windows connection. If the options that you select while installing on each computer are always the same, then you save the time of reviewing each installation screen and selecting the various installation options.

Silent installations can also ensure that multiple users in your organization use the same installation options when they install Oracle products. You can therefore more easily support those users because you already know what components and options have been installed on each computer.

Before you perform a silent installation, you should review the settings in the response file template that is provided with the Oracle product.

Modifying a Response File

If the product installation includes a response file template, then you can find it on the shipped media in the *device*: [DISK1.RESPONSE] directory. For a list of response files, refer to [Section , "List of Response File Names"](#). Mandatory items have the following comment just before the definition:

```
# OPENVMS_VALUE_REQUIRED
```

Three values must be provided:

- FROM_LOCATION
- ORACLE_HOME
- ORACLE_HOME_NAME

FROM_LOCATION is the complete path of the source of the products to be installed (in other words, to the PRODUCTS.XML file in the STAGE subdirectory of the DATABASE, CLIENT, COMPANION, or CLUSTERWARE main directory of the DVD) from the shipped media. If contents of shipped media were copied to a local device, then specify that path. ORACLE_HOME is the directory into which the Oracle software will be installed. The account from which Oracle Universal Installer is run must have appropriate privileges to create a root-level directory if the directory that was specified for ORACLE_HOME does not already exist.

Note:

- FROM_LOCATION must be in UNIX-style syntax.
 - ORACLE_HOME must be on an ODS-5 disk
-
-

To modify the response file:

1. Make a copy of the response file for the product, and open it in a text editor.
2. Search for the three items that are listed in the preceding paragraph (or for the string OPENVMS_VALUE_REQUIRED), and provide values for those items.

-
3. Modify other fields as applicable (refer to comments within the response file that you are modifying).

List of Response File Names

The following table lists the response file names and their descriptions:

File Name	Description
ENTERPRISE.RSP:	Enterprise Edition Installation of Oracle Database Server
CUSTOM.RSP	Custom Edition Installation of Oracle Database Server
CLIENTADMIN.RSP	Administrator installation of Oracle Database Client
CLIENTRUNTIME.RSP	Run time installation of Oracle Database Client
CLIENTCUSTOM.RSP	Custom installation of Oracle Database Client
DBCA.RSP	Database Configuration Assistant

Performing Silent Installation

You can install Oracle software noninteractively by specifying a response file when you start Oracle Universal Installer. The Installer uses the values contained in the response file to provide answers to some or all of Oracle Universal Installer prompts. If you include responses for all of Oracle Universal Installer prompts in the response file, then you can run a completely noninteractive installation.

To install and configure Oracle products noninteractively, complete the following steps:

- [Section , "Creating the ORAINST.LOC File"](#)
- [Section , "Preparing Response Files"](#)
- [Section , "Running Oracle Universal Installer Noninteractively"](#)

These steps are described in the following sections.

Creating the ORAINST.LOC File

If you plan to install Oracle products noninteractively, then you must manually create the `ORAINST.LOC` file, if it does not already exist. This file specifies the location of the Oracle Inventory directory where Oracle Universal Installer creates the inventory of Oracle products that are installed on the system.

Note: If Oracle software has been installed previously on the system, then the `ORAINST.LOC` file may already exist. If the file does exist, then you do not need to create a new file.

To create the `ORAINST.LOC` file, perform the following steps:

1. Set default directory to the Oracle `oraInventory` directory. The `ORAINST.LOC` file resides in a subdirectory under the `SYS$LOGIN`

```
$ SET DEFAULT logindisk:[logindir.nodename]
```

2. Use a text editor to create the `ORAINST.LOC` file, containing the following lines:

```
inventory_loc=oraInventory_location  
inst_group=group_id
```

In this example, *oraInventory_location* is the location of the Oracle Inventory directory which is on an ODS-5 volume, and *group_id* is the group ID of the Oracle Database account.

3. Set the file protection on the ORAINST.LOC file as follows:

```
$ SET FILE/PROT=(O:RWED,S:RWED,:rwe,w:re):oraInst.loc
```

Preparing Response Files

Oracle provides response file templates for each product and installation type and for each configuration tool. These files are located in the [DATABASE.RESPONSE] directory on the installation media.

Table 8–2 lists the response files that are provided with Oracle Database 10g.

Table 8–2 Response Files

Response File	Description
ENTERPRISE.RSP	Enterprise Edition installation of Oracle Database
CUSTOM.RSP	Custom installation of Oracle Database
DECA.RSP	Database Configuration Assistant
NETCA.RSP	Oracle Net Services Configuration Assistant

To prepare a response file:

1. Copy the response file from the response file directory to a directory on the system:

```
$ COPY dvd_device:[DB.RESPONSE]response_file dest_path
```

In this example, *src_path* is the installation media mount point directory or the *db* directory on the DVD. If you have copied the software to a hard drive, then you can edit the file in the *RESPONSE* directory if you prefer.

2. Open the response file in a text editor:

```
$ EDIT [dest_path]response_file
```

3. Edit the file by following the directions in [Section , "Modifying a Response File"](#) on page B-2.

Note: Oracle Universal Installer or the configuration assistant fails if you do not correctly configure the response file. Refer to [Section , "Silent Response File Error Handling"](#) for more information about troubleshooting a failed noninteractive installation.

Running Oracle Universal Installer Noninteractively

To run Oracle Universal Installer noninteractively, enter a command similar to the following:

```
$ @[directory_path]RUNINSTALLER [-SILENT] -RESPONSEFILE filename
```

In this command:

- *directory_path* is the DATABASE, CLIENT, COMPANION, or CLUSTERWARE directory in the hard drive copy of the DVD

-
- `-SILENT` specifies a completely noninteractive installation
 - `filename` is the full path and file name of the installation response file that you configured

Note: Do not specify a relative path to the response file. If you specify a relative path, then Oracle Universal Installer stops.

Running Oracle Universal Installer in Silent Mode or Suppressed Mode

To run Oracle Universal Installer in silent mode or suppressed mode, use the following steps:

1. Complete the preinstallation tasks listed in [Chapter 2, "Preinstallation Tasks"](#).
2. Log into the Oracle Database account (typically, `oracle`).
3. If you are performing a suppressed-mode installation, set the `DISPLAY`.
4. To start Oracle Universal Installer in silent mode or suppressed mode, enter a command similar to the following command:

```
$ @[directory_path]RUNINSTALLER [-SILENT] -RESPONSEFILE filename
```

Note: Do not specify a relative path to the response file. If you specify a relative path, then Oracle Universal Installer fails.

In the example:

- `directory_path` is the `DATABASE`, `CLIENT`, `COMPANION`, or `CLUSTERWARE` directory in the hard drive copy of the DVD
- `-SILENT` indicates that you want to run Oracle Universal Installer in silent mode.
- `filename` is the full path and file name of the installation response file that you configured.

Note: For more information about other options for the `RUNINSTALLER` command, enter the following command:

```
$ @[directory_path]RUNINSTALLER -HELP
```

Running Oracle Network Configuration Assistant in Silent Mode

You can run Oracle Network Configuration Assistant in silent mode to configure and start an Oracle Net Services listener on the system, to configure naming methods, and to configure Oracle Net Services service names. To run the Oracle Network Configuration Assistant in silent mode, you must copy and edit a response file template. Oracle provides a response file template named `NETCA.RESP` in the `[DATABASE.RESPONSE]` directory on the DVD.

To run Oracle Network Configuration Assistant in silent mode, perform the following steps:

1. Copy the `NETCA.RSP` response file template from the response file directory to a directory on the system.

```
$ COPY [DATABASE.RESPONSE]NETCA.RSP local_directory
```

2. Open the response file in a text editor. For example:

```
$ $ EDIT [local_directory]NETCA.RSP
```

3. Edit the file by following the instructions in the file.

Note: Oracle Network Configuration Assistant stops if you do not correctly configure the response file.

4. Log into the Oracle Database account (typically `oracle`) and set your Oracle environment by running `ORAUSER.COM` in the top-level Oracle home directory.
5. Enter a command similar to the following to run the Oracle Network Configuration Assistant in silent mode:

```
$ @ORA_ROOT:[BIN]NETCA -SILENT -RESPONSEFILE=[local_dir]NETCA.RSP
```

In this example:

- `/silent` option indicates that you want to run the Oracle Network Configuration Assistant in silent mode.
- `local_dir` is the full path of the directory where you copied the `NETCA.RSP` response file template.

Running Database Configuration Assistant in Noninteractive or Silent Mode

You can run Database Configuration Assistant in noninteractive or silent mode to configure and start Oracle Database on the system. To run Database Configuration Assistant in noninteractive or silent mode, you must copy and edit a response file template. Oracle provides a response file template named `DECA.RESP` in the `[.db.response]` directory on the DVD.

Silent mode or noninteractive mode requires an appropriate response file. The following sections discuss how to use and run Database Configuration Assistant in silent mode or noninteractive mode:

- [Section , "Using Database Configuration Assistant in Noninteractive Mode"](#)
- [Section , "Using Database Configuration Assistant in Silent Mode"](#)
- [Section , "Running Database Configuration Assistant in Noninteractive or Silent Mode"](#)

Using Database Configuration Assistant in Noninteractive Mode

In noninteractive mode, Database Configuration Assistant uses values that you specify in the response file or as command line options to create a database. As Database Configuration Assistant configures and starts the database, it displays a screen that contains status messages and a progress bar. The screen that it displays is the same screen that is displayed when you create a preconfigured database during an Enterprise Edition installation.

To run Database Configuration Assistant in noninteractive mode, you must use a graphical display and set the `DISPLAY`.

Using Database Configuration Assistant in Silent Mode

In silent mode, Database Configuration Assistant uses values that you specify in the response file or as command-line options to create a database. Database Configuration Assistant does not display any screens or information while it creates the database. It logs all messages (including information, errors, and warnings) in a log file.

From the command line, enter the following command to see all of Database Configuration Assistant options that are available when you use the silent mode:

```
$ DBCA -help
```

Running Database Configuration Assistant in Noninteractive or Silent Mode

To run Database Configuration Assistant in noninteractive or silent mode, use the following steps:

Note: Because Database Configuration Assistant design causes GUI classes to be loaded (even in silent mode), a silent installation requires an X-Windows connection.

1. Copy the `DBCA.RSP` response file template from the response file directory to a directory on the system:

```
$ COPY device:[DB.RESPONSE]DBCA.RSP dest_path
```

In this example, `src_path` is the installation media directory or the `db` directory on the DVD. If you have copied the software to a hard drive, then you can edit the file in the `[.DB.RESPONSE]` directory if you prefer.

2. Open the response file in a text editor as follows:

```
$ EDIT NETCA.RSP
```

3. Edit the file by following the instructions in the file.

Note: Database Configuration Assistant stops if you do not correctly configure the response file.

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 Database Configuration Assistant. For information about the supported options, enter the following command:

```
$ DBCA -HELP
```

Because of the DCL limit of eight command-line parameters on OpenVMS, it is not possible to specify all the required parameters to create a database directly at the command line. Therefore, as a workaround, Database Configuration Assistant command-line arguments may be placed in a file called `SYS$SCRATCH:DBCA.ARGS`. The `dbca.com` script looks for this file, and if present, passes the specified arguments to Database Configuration Assistant Java application. The `DBCA.COM` command procedure ignores all arguments that are specified at the command line except: `-SILENT`, `-RESPONSEFILE response_file`, and `-HELP`. To avoid inadvertently applying the same arguments to the next invocation of Database Configuration Assistant when `DBCA.ARGS` exits, if an arguments file was used, then it is renamed to `SYS$SCRATCH:DBCA.ARGS_OLD`.

4. Log into the Oracle Database account (typically `oracle`) and set the `ORACLE_HOME` logical name to specify the correct Oracle home directory.
5. If you intend to run Database Configuration Assistant in noninteractive mode, then set the `DISPLAY`.
6. Enter a command similar to the following to run Database Configuration Assistant in noninteractive or silent mode with a response file:

```
$ DBCA -SILENT -RESPONSEFILE  
response_file
```

In the example:

- `-silent` indicates that you want to run Database Configuration Assistant in silent mode.
- `response_file` is the full path to the copy of `DBCA.RSP` response file template that will be used.

The following sections contain examples that illustrate the use of silent mode.

Database Configuration Assistant Silent Mode Example 1: Creating a Clone Database

To create a clone database, enter the following in `SYS$SCRATCH:DBCA.ARGS`:

```
-silent  
-createDatabase  
-templateName Transaction_Processing.dbc  
-gdbname ora10  
-sid ora10  
-datafileJarLocation /disk$disk1/oracle10g/ora10/assistants/dbca/templates  
-datafileDestination /disk$disk1/oracle10g/ora10/oradata
```

```
-responseFile NO_VALUE
-characterSet WE8ISO8859P1
```

Note: File specifications in an argument file may be in either OpenVMS syntax or in Unix syntax.

Then, ensure that the ORACLE_HOME environment is set up by running:

```
$ @disk:[oracle-home-path]ORAUSER
```

For example:

```
$ @DISK$DISK1:[ORACLE10g.ORA10]ORAUSER
```

Next, set the DISPLAY and start Database Configuration Assistant. Log files are present in the same location as with an interactive install. In this example:

```
DISK$DISK1:[ORACLE10g.ORA10.ADMIN.ORA10.CREATE]
```

In addition, check the ORA_ROOT: [ASSISTANTS.DBCA.LOGS] directory.

Database Configuration Assistant Silent Mode Example 2: Creating a Seed Template

To create a seed template, enter the following in SYS\$SCRATCH:DBCA.ARGS:

```
-silent
-createCloneTemplate
-sourceDB ora10
-sysDBAUserName sys
-sysDBAPassword change_on_install
-templateName copy_of_ora10
-datafileJarLocation /disk$disk1/oracle10g/ora10/assistants/dbca/templates
```

Then, ensure that the oracle environment is set up by running the ORAUSER script, set the DISPLAY and start Database Configuration Assistant. Log files for creating a seed template are located in the ORA_ROOT: [ASSISTANTS.DBCA.LOGS] directory.



Oracle RAC: Other Considerations

This appendix discusses troubleshooting and other considerations when using Oracle RAC.

Topics covered in this appendix are:

- [Section , "Documentation Sources"](#)
- [Section , "Troubleshooting the Oracle RAC Installation Process"](#)
- [Section , "Preconversion Considerations"](#)
- [Section , "Converting from Single-Instance to Oracle RAC"](#)
- [Section , "Postconversion Steps"](#)
- [Section , "OpenVMS Directory Structures for Oracle RAC"](#)

Documentation Sources

The information in this Appendix is meant to supplement the detailed information you will find in *Oracle Real Application Clusters Administrator's Guide* (B10765-02). Refer to this document as necessary when reading through this Appendix.

In addition, you will find important OpenVMS-specific installation and configuration information pertaining to Oracle RAC in [Chapter 5, "Installing Oracle RAC"](#) and [Chapter 6, "Configuring Oracle RAC"](#).

See Also: For more information about configuring and using Oracle RAC, refer to the *Oracle Real Application Clusters Installation and Configuration Guide*

Troubleshooting the Oracle RAC Installation Process

The following sections contain troubleshooting information to assist you during the installation of Oracle RAC.

See Also: The Oracle RAC documentation set that is included on the Oracle Database 10g Server Documentation DVD consists of the following documents:

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

Oracle RAC Management Tools Error Messages are in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*

Preconversion Considerations

The following sections describe the considerations for converting from Oracle Database 10g single-instance databases to Oracle RAC databases.

- [Section , "Prerequisites for Conversion"](#)
- [Section , "Administrative Considerations"](#)

If you are upgrading from Oracle Parallel Server to Oracle RAC or from an earlier version of Oracle RAC, then follow the steps described in [Section , "Upgrade Issues"](#). It is assumed that the original single-instance database and the target Oracle RAC database are of the same version of Oracle Database 10g and running on the same platform.

Prerequisites for Conversion

The system that you are converting to Oracle RAC must meet the following hardware and software requirements:

- A supported hardware and operating system software configuration
- An additional license for Oracle Database 10g Enterprise Edition with Oracle RAC

Administrative Considerations

Note the following administrative considerations before conversion:

- Backup procedures should be available before converting from a single-instance Oracle Database to Oracle RAC.
- The archive file format requires a thread number. In addition, the archived logs from all instances of a Oracle RAC database are required for media recovery. You must ensure that the disks on which the archive logs are placed are accessible from all nodes on which the cluster database has instances.

Converting from Single-Instance to Oracle RAC

Oracle recommends that you use Database Configuration Assistant to convert from single-instance Oracle Databases to Oracle RAC. Database Configuration Assistant automates the configuration of the control file attributes, creates the undo tablespaces and the redo logs, and makes the initialization parameter file entries for

cluster-enabled environments. It also configures the Oracle Net Services and Oracle Clusterware resources and performs the configuration for Oracle RAC database management for use by Oracle Enterprise Manager or by the `SRVCTL` utility. This section describes the following scenarios:

- [Section , "Single-Instance Database on a Noncluster Computer to Oracle RAC"](#)
- [Section , "Single-Instance Database on a Cluster to Oracle RAC"](#)

Single-Instance Database on a Noncluster Computer to Oracle RAC

To convert from a single-instance Oracle Database that is on a noncluster computer to Oracle RAC, perform the following procedures in the order shown:

1. [Section , "Backing Up the Original Single-Instance Database"](#)
2. [Section , "Performing the Preinstallation Steps"](#)
3. [Section , "Setting Up the Cluster"](#)
4. [Section , "Copying the Preconfigured Database Image"](#)
5. [Section , "Installing Oracle Database with Oracle RAC"](#)

Backing Up the Original Single-Instance Database

Use Database Configuration Assistant to create a preconfigured image of the single-instance database. To do this, perform the following:

1. Start Database Configuration Assistant with the `DBCA` command.
2. On the Welcome screen, select **Manage Templates**.
3. Select **Create a database template**.
4. Select the database name from **Database Name**.
5. Enter the template name and description and template data file location in **Template Name**. Use the database name as the default template name.
6. Click **Finish**.

Database Configuration Assistant generates two files, a database structure file (`template_name.DBC`) and a database preconfigured image file (`template_name.DFB`). These files are generated by default in the `ORA_ROOT:[ASSISTANTS.DBCA.TEMPLATES]` directory.

Performing the Preinstallation Steps

Perform the preinstallation steps as documented in this book. Then, set up shared storage by referring to [Section , "Creating Directories for Oracle Database or Recovery Files"](#).

See Also: Vendor-specific documentation about storage for information about how to mirror and stripe disks

Setting Up the Cluster

If have not already done so, create a cluster with the required number of nodes according to the vendor documentation. Once you have configured all the nodes in the cluster, install Oracle Clusterware by following the procedure in [Section , "Installing Oracle Clusterware"](#).

Copying the Preconfigured Database Image

This includes copying the database structure *.DBC file and the database preconfigured image *.DFB file to a temporary location on the node in the cluster from which you plan to run Database Configuration Assistant. These files are created by Database Configuration Assistant. For more information, refer to [Section , "Backing Up the Original Single-Instance Database"](#).

Installing Oracle Database with Oracle RAC

Perform the following tasks to install Oracle Database 10g with Oracle RAC:

1. Run Oracle Universal Installer.
2. Select **Cluster Installation Mode** on the Specify Hardware Cluster Installation screen, and select the nodes to include in the Oracle RAC database.
3. On the Database Configuration Types screen, select the **Advanced** install type.
After installing the Oracle software, Oracle Universal Installer runs the postinstallation configuration tools, such as the Oracle Network Configuration Assistant, Database Configuration Assistant, and so on.
4. On the Database Configuration Assistant Template Selection screen, use the template that you copied to a temporary location in the [Section , "Copying the Preconfigured Database Image"](#) procedure. Use the browse option to select the template location.
5. After creating the RAC database, Database Configuration Assistant displays the Password Management screen. On this screen, change the passwords for the privileged users with the SYSDBA and SYSOPER roles. Once Database Configuration Assistant exits, the conversion process is complete.

Single-Instance Database on a Cluster to Oracle RAC

There are three scenarios in which a single-instance database can exist on a cluster computer:

- The Oracle home from which the single-instance database is running is installed on a cluster (Oracle Database 10g with RAC).
- The Oracle home from which the single-instance database is running is installed on a cluster but the RAC feature is disabled.
- The Oracle home from which the single-instance database is running is not installed on a cluster.

Use the following procedures to convert the single-instance database on a cluster computer to Oracle RAC for all of these scenarios.

- [Section , "Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home"](#)
- [Section , "Automated Conversion"](#)
- [Section , "Manual Conversion"](#)
- [Section , "Single-Instance Database on a Cluster Running from a Oracle RAC-Disabled Oracle Home"](#)
- [Section , "Single-Instance Database on a Cluster Running from a Noncluster-Installed Oracle Home"](#)

Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home

Perform the following tasks to convert a single-instance database on a cluster running from a cluster installed Oracle home.

1. Use Database Configuration Assistant to create a preconfigured image of the single-instance database as described in [Section , "Backing Up the Original Single-Instance Database"](#). To perform the conversion manually, shut down the single-instance database.
2. To add nodes to the cluster, add and connect these nodes to the cluster as described in [Section , "Performing the Preinstallation Steps"](#). Ensure that all these nodes can access the shared storage.
3. From one of the newly added nodes, configure the listeners on the additional nodes using the Oracle Network Configuration Assistant. Select the same port number and protocol that you used on the existing node. If the Oracle Network Configuration Assistant displays the existing node on the Node List screen, then do not select this node because the listener is already configured on it.
4. Convert the database using one of the following procedures:
 - [Section , "Automated Conversion"](#)
 - [Section , "Manual Conversion"](#)

Automated Conversion

The following steps describe the procedure to automatically convert the database:

1. If you created the preconfigured image of the single instance database as described in [Section , "Backing Up the Original Single-Instance Database"](#), then use Database Configuration Assistant to complete the conversion to a Oracle RAC database.
2. Start Database Configuration Assistant from the initial node. Select the names of the nodes that you want to include as part of the cluster database. On the Template Selection screen, select the preconfigured template that you create in Step 1. Enter the database name and respond to the remaining Database Configuration Assistant prompts.

After creating the Oracle RAC database, Database Configuration Assistant displays the Password Management screen. Change the passwords for database privileged users who have `SYSDBA` and `SYSOPER` roles. Once Database Configuration Assistant exits, the conversion process is complete.

Manual Conversion

Perform the following steps to manually convert the database:

1. Re-create the control files by running the `CREATE CONTROLFILE` SQL statement with the `REUSE` keyword, and specify `MAXINSTANCES`, `MAXLOGFILES`, and so on, as needed for the Oracle RAC configuration. The default `MAXINSTANCES` value is 32 (recommended).
2. Shut down the database instance.
3. If the single-instance database was using an `SPFILE` parameter file, then create a temporary `PFILE` from the `spfile` using the following SQL statement:

```
CREATE PFILE='pfile_name' FROM SPFILE='spfile_name'
```

-
4. Set the `CLUSTER_DATABASE` parameter to `TRUE`. Set the `INSTANCE_NUMBER` parameter to a unique value for each instance, using the `sid.parameter=value` syntax.

If you optimized memory usage on the single-instance database, adjust the size of the System Global Area (SGA) to avoid swapping and paging when you convert to Oracle RAC. This is because Oracle RAC requires about 350 bytes for each buffer to accommodate the Global Cache Service (GCS). For example, if you have 10,000 buffers, Oracle RAC requires about $350 * 10,000$ bytes more memory. Therefore, adjust the size of the SGA by changing the `DB_CACHE_SIZE` and `DB_nK_CACHE_SIZE` parameters accordingly.

5. Start the database instance using the `PFILE` created in Step 3.
6. If the single-instance database was using automatic undo management, then create an undo tablespace for each additional instance using the `CREATE UNDO TABLESPACE SQL` statement.
7. Create redo threads that have at least two redo logs for each additional instance. Enable the new redo threads by using an `ALTER DATABASE SQL` statement. Then shut down the database instance.
8. Copy the Oracle password file from the initial node, or from the node on which you are working, to the corresponding location on the additional nodes on which the cluster database will have an instance. Ensure that you replace the `ORACLE_SID` name in each password file for each additional instance.
9. Add `REMOTE_LISTENER=LISTENERS_DB_NAME` and `sid.LOCAL_LISTENER=LISTENER_SID` parameters to the `PFILE`.
10. Configure the Oracle Net Services entries for the database and instances and address entries for the `LOCAL_LISTENER` for each instance and `REMOTE_LISTENER` in the `TNSNAMES.ORA` file, and copy it to all nodes.
11. Create the `SPFILE` from the `PFILE` as described in [Section , "Procedures for Upgrading to the Server Parameter File"](#).
12. Create the `ORA_ROOT:[DBS] INITsid.ora` file that contains the following entry:

```
spfile='spfile_path_name'
```

where `spfile_path_name` is the complete path name of the `SPFILE`.

13. Add the configuration for the Oracle RAC database and its instance-to-node mapping using the `SRVCTL` utility.
14. Start the Oracle RAC database using `SRVCTL`.

After starting the database with `SRVCTL`, the conversion process is complete. For example, you can run the following SQL statement to see the status of each instance in the Oracle RAC database:

```
SELECT * FROM v$active_instances
```

Single-Instance Database on a Cluster Running from a Oracle RAC-Disabled Oracle Home

This installation is possible if you performed a one-node cluster (with Oracle RAC) installation but later disabled the Oracle RAC feature by unlinking it from the `ORACLE` binary before creating the single-instance database. However, you can also select the local, noncluster selection on the Node Selection screen to create a single-instance home on a cluster that is not Oracle RAC-enabled.

Perform the following procedures to convert this type of single-instance database to a Oracle RAC database:

1. On the cluster node where the single-instance database is running, run Step 1 of [Section , "Single Instance on a Cluster Running from a Cluster-Enabled Oracle Home"](#).
2. Set default to `ORA_ROOT: [RDBMS.BIN]`.
3. Relink the ORACLE binary as follows:
 - Shut down the database if it is already running.
 - Remove the ORACLE binary by running the following command:

```
$ REMORACLE
```
 - Create the single-instance Oracle image by running the following command:

```
$ @ORA_ROOT:[BIN]RELINK ORACLE
```
 - Run the following command:

```
$ INSORACLE
```
4. Continue with Step 2 on page C-5.

Single-Instance Database on a Cluster Running from a Noncluster-Installed Oracle Home

This installation is only possible if you selected the local installation option on the Specify Hardware Cluster Installation screen during an Oracle Database 10g installation.

To convert this database to a RAC database, perform the procedures described under the following headings:

- [Section , "Backing Up the Original Single-Instance Database"](#)
- [Section , "Performing the Preinstallation Steps"](#)
- [Section , "Setting Up the Cluster"](#)
- [Section , "Installing Oracle Database with Oracle RAC"](#)

Ensure that you select a new Oracle home other than the one from which the single-instance database was running.

Postconversion Steps

After completing the conversion, note the following points as described in the Oracle RAC documentation:

- Follow the recommendations for using load balancing and transparent application failover (TAF) as described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.
- Use locally managed tablespaces instead of dictionary-managed tablespaces to reduce contention and manage sequences in Oracle RAC as described in *Oracle Real Application Clusters Deployment and Performance Guide*.
- Follow the guidelines for configuring an interconnect, using automatic segment space management, and using `SRVCTL` to administer multiple instances as

described in *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*.

The buffer cache and shared pool capacity requirements in Oracle RAC are slightly greater than those in single-instance Oracle Databases. Therefore, you may want to increase the size of the buffer cache by about 10% and the size of the shared pool by about 15%.

OpenVMS Directory Structures for Oracle RAC

Table 8–3 shows the hierarchical directory tree of a sample database for RAC on OpenVMS-based systems.

Table 8–3 *Directory Structure for a Sample OpenVMS Environment*

Root Level	Second Level	Third Level	Fourth Level
ORACLE_HOME			
[.ADMIN]			Administrative directory
	[.DB_UNIQUE_NAME]		Database unique name, the same as dbname when the database name is eight or fewer characters in length
		[.BDUMP] [.CDUMP] [.HDUMP] [.UDUMP]	Dump destinations for the database server
		[.PFILE]	Parameter file for the database server
ORA_CRS_HOME			
	[.BIN]		Subtree for Oracle binaries
	[.LOG]		Subtree for log files for CRS daemons/RACG commands
	[.NETWORK]		Subtree for Oracle Net

See Also: *Oracle Database Administrator's Reference* for more information about the ORACLE_HOME and admin directories

Installing, Configuring, and Running EMAgent

This appendix describes Enterprise Manager Agent (EMAgent) for Oracle Database 10g release 2 (10.2). It includes information about the following topics:

- [Section , "Introduction to EM Framework"](#)
- [Section , "Installation Requirements"](#)
- [Section , "Installation and Configuration"](#)
- [Section , "Management and Maintenance"](#)
- [Section , "Supported Targets and Jobs"](#)
- [Section , "Known Limitations"](#)

Introduction to EM Framework

This release of EMAgent is different from the previous releases of Oracle Intelligent Agent in terms of architecture and implementation. EMAgent is part of the Enterprise Manager Framework. The DBConsole and Application Control components are not supported on Oracle Database 10g for OpenVMS. To use EMAgent for OpenVMS, it is necessary to install the Oracle Management Server (OMS) component of Grid Control Enterprise Manager on a UNIX-based or Microsoft Windows-based computer. EMAgent running on OpenVMS will register with and function in coordination with the OMS.

Starting with Release 10gR2 (10.2.0.2), the EMAgent is not shipped with the Oracle Database 10g Kit. It is shipped separately as a standalone Grid Control Enterprise Manager kit. The EMAgent in 10gR2 will not work with the Enterprise Manager shipped with release 9i and earlier.

EMAgent is implemented in C, Java, and Perl. Unlike earlier versions, TCL scripts are no longer used. These are replaced by Perl scripts. Perl version 5.8 is included in the kit along with the EMAgent. Oracle Universal Installer automatically installs Perl. It is not required to install a separate Perl kit. Note that this kit includes Oracle-specific DBD or DBI, and is therefore, the only supported Perl kit for use with Oracle Database. The logical name `PERL_ROOT` will be defined to the physical path `ORA_ROOT : [PERL]` as a job level logical name. To avoid potential conflict with Perl from any other kit, including any future versions supplied by Oracle, it is recommended not to convert this logical name to a `GROUP` or `SYSTEM` logical name.

The Job model supported in previous releases is still available in this release in a similar, but not identical, fashion. Jobs can be submitted, their status can be monitored, and their output checked. The Event model, as it existed in earlier versions, is no

longer available. Instead, what is available is Targets. Refer to [Section , "Supported Targets and Jobs"](#) for more information.

See Also: For more details, refer to the `README_VMS_10GR2.TXT` file shipped with the Oracle Database Server Kit. The information in this file supersedes the information in this appendix

Installation Requirements

If you choose not to create a dedicated batch queue `ORA_BATCH_QUEUE_hostname` for EMAgent, then you must define a logical name `ORA_BATCH_QUEUE_hostname` (a Process or Job logical is sufficient) to the name of the batch queue to be used for running EMAgent. In addition to running in the batch queue, EMAgent also submits a number of other tasks, typically related to actions required on other Oracle installations on the same host.

The *hostname* part of the logical name should be the same as the value of the `TCPIP$INET_HOST` logical name. The batch queue specified should run at the same priority level as all other general purpose processes on the system, typically 4.

For example, the following qualifiers would be typical for creating the batch queue that is used for EMAgent.:

```
/BASE_PRIORITY=4 /CPUDEFAULT=INFINITE /CPUMAXIMUM=INFINITE /JOB_LIMIT=100  
/OWNER=[SYSTEM] /PROTECTION=(S:M,O:RSD,G:S,W:S)
```

EMAgent internally spawns a number of subprocesses and detached processes to run a majority of its tasks. The type of actions performed in these processes ranges widely from running simple DCL commands to running Java applications. It is imperative that the account used to install and run EMAgent not modify the typical OpenVMS session environment in any way that would alter the expected output of normal DCL commands. The `LOGIN.COM` file of the account used to start or stop the agent should not have any such redefinitions. If the redefinitions are required for any reason, then it is recommended that they be disabled in the batch mode, because EMAgent runs in the batch mode.

Installation and Configuration

When you initially run Oracle Universal Installer, it instantiates certain files under the `ORA_ROOT:[SYSMAN.EMD]` and `ORA_ROOT:[SYSMAN.CONFIG]` directories. These are the Targets files and the Properties files. However, these cannot be used directly. EMAgent is configured correctly when Database Configuration Assistant is run. The configuration files are created under a separate directory structure (`ORA_ROOT:[hostname_sid.SYSMAN. . .]`). Because the DB Console is not supported on OpenVMS, it is necessary to make certain manual changes to the configuration file or files before starting EMAgent.

Manual Changes Required Before Attempting to Start EMAgent

You must manually change the following properties in the `ORA_ROOT:[Hostname_SID.SYSMAN.CONFIG]EMD.PROPERTIES` file before starting EMAgent.

1. `REPOSITORY_URL`

This property is required to allow EMAgent to upload data to a central Grid Control EM Repository. When EMAgent starts, it registers its targets and other host configuration information with the Oracle Management Server (OMS) so that they are displayed on the Grid Control EM Console. It is then possible to manage

the targets from the EM Console. A sample `REPOSITORY_URL` property is as follows:

```
REPOSITORY_URL=http://orclhost01.us.oracle.com:4889/em/upload
```

2. `agentTZRegion`

This property indicates the time zone in which EMAgent is running. To update this property in the `EMD.PROPERTIES` file, run the following command:

```
$ EMCTL CONFIG AGENT UPDATETZ
```

A sample `agentTZRegion` property for Pacific Standard Time is as follows:

```
agentTZRegion=-08:00
```

Management and Maintenance

This section describes the procedures to manage and maintain EMAgent.

- [Section , "Startup, Shutdown, and Status"](#)
- [Section , "Troubleshooting and Maintenance"](#)

Startup, Shutdown, and Status

Perform the following steps to start, shut down, or view the status of EMAgent:

- After the manual configuration is complete, run the following command to start EMAgent:

```
$ EMCTL START AGENT
```

- To shut down EMAgent, run the following command:

```
$ EMCTL STOP AGENT
```

- To query the status of EMAgent at any time, run the following command:

```
$ EMCTL STATUS AGENT
```

Note: EMAgent is the only mechanism of communication between the Oracle Management Server and targets or applications running on the host. Therefore, it is not possible to start up or shut down EMAgent from the EM Console.

Troubleshooting and Maintenance

Perform the following tasks to manage or troubleshoot EMAgent:

- [Section , "EMAgent Fails to Start"](#)
- [Section , "Extended File Specification \(EFS\) Characteristics"](#)
- [Section , "TMP Directory"](#)
- [Section , "Monitoring the Batch Queue"](#)
- [Section , "Disk Space on EMAgent Install Area"](#)
- [Section , "Resetting EMAgent Environment"](#)

EMAgent Fails to Start

If EMAgent fails to start, check the following files for typical error messages:

Batch Job Log File

EMAgent is submitted as a batch job into the agent batch queue whenever it is started. Each time EMAgent starts, a fresh batch log file is created, which is available at `ORA_ROOT: [SYSMAN.LOG] START_AGENT_HOST.LOG`. Refer to the latest version of this log file. If it is readable (not locked by a running EMAgent process), look through the file to see if there are any failure messages. Typical errors would be `Failure to launch the EMAgent` because of some issues with related shared libraries.

EMAgent Log and Trace Files

If there are no errors in `ORA_ROOT: [SYSMAN.LOG] START_AGENT_HOST.LOG`, look at the `emagent.trc` and `emagent.log` files. These are located in the `ORA_ROOT: [host_sid.SYSMAN.LOG]` directory. A common error is `Address already in use`, when the port number on which EMAgent is listening is being used by some other application.

Note: If you are not able to restart the Agent, please make sure that there are no processes on the system that start with `ORA_EM*`.

You can stop all the process by entering the command `$ STOP/ID=<process-id>` on each one of them and then try to restart the agent.

Extended File Specification (EFS) Characteristics

EMAgent requires an extended file-specifications environment to handle files with multiple dots in their names, long file names, and so on. The default Oracle environment (after running `ORAUSER.COM`) does not provide this environment. Internally, EMAgent tools set up this extended environment when started, and reset the environment back to the original when completed. In a normal run, it is not necessary for an EMAgent administrator to require this environment for interactive use. However, in certain situations, when there are failures due to interruption in the host system, or due to lack of resources, it may be required to manually manipulate certain files, which in turn, would need the EFS environment.

There are two scripts included with EMAgent kit that provides the EFS environment:

- `ORA_ROOT: [SYSMAN.BIN] EFS.COM`
Enables the EFS environment..
- `ORA_ROOT: [SYSMAN.BIN] NOEFS.COM`
Resets the environment to non-EFS.

For convenience, two DCL symbols have been created to set and reset the EFS environment. Symbol `EMDEFS` sets the EFS environment, and symbol `NOEMDEFS` resets the EFS environment.

TMP Directory

EMAgent run time creates a number of temporary files during processing. All temporary files are created in a directory specified by the logical name `ORA_AGENT_TMP`. This logical name is automatically defined to the physical path, `ORA_ROOT: [SYSMAN.host.TMP]`, when setting up the Oracle environment.

It is possible that certain temporary files may not be deleted due to interruptions or failures. It is recommended that this directory content be monitored on a periodic basis and files that are more than three hours old be deleted.

Monitoring the Batch Queue

As mentioned in [Section , "Installation Requirements"](#), EMAgent runs as a batch job in a batch queue. EMAgent also submits a number of jobs to the batch queue for tasks such as monitoring, start up and shut down of databases and listeners running in other Oracle installations. There is a possibility that broken environments in these installations could cause these submitted jobs to stop responding. It is recommended that the batch queue be periodically monitored for any long-pending jobs (any job of more than three hours is long-pending) and such jobs be deleted.

Disk Space on EMAgent Install Area

There are two agent parameters listed in the `EMD.PROPERTIES` properties file related to disk space:

- `UploadMaxDiskUsedPct`
- `UploadMaxDiskUsedPctFloor`

Ensure that the agent parameters are set correctly at the required level. When the percentage of space used on the disk exceeds `UploadMaxDiskUsedPct`, uploads will stop, as well as updates from the EMAgent to the Oracle Management Server. When uploads are stopped, EMAgent status will appear as up and running, but the line reporting `Last Successful Upload` will show a time stamp that remains the same over periods exceeding the value of `UploadInterval` in the `EMD.PROPERTIES` file. Uploads will be enabled again when the percentage of space used on the disk falls below the value of `UploadMaxDiskUsedPctFloor`, which should be less than `UploadMaxDiskUsedPct`.

Resetting EMAgent Environment

You can shut down EMAgent that is already running, and reconfigure it to point to a different Oracle Management Server. To reset EMAgent environment, it is recommended to perform the following tasks:

1. Delete the `ORA_ROOT:[host_sid.SYSMAN.EMD]LASTUPLD.XML`; * file.
2. Delete all files in the `ORA_ROOT:[host_id.SYSMAN.EMD.UPLOAD]` directory.
3. Delete all files in the `ORA_ROOT:[host_id.SYSMAN.EMD.STATE]` directory.
4. Delete all files in the `ORA_ROOT:[host_id.SYSMAN.EMD.COLLECTION]` directory.
5. Delete all files in the `ORA_ROOT:[host_sid.SYSMAN.EMD.RECV]` directory.
6. Rename or delete EMAgent log and trace files:

```
ORA_ROOT:[host_sid.SYSMAN.LOG]EMAGENT.LOG  
ORA_ROOT:[host_sid.SYSMAN.LOG]EMAGENT.TRC
```

Supported Targets and Jobs

The following targets and jobs are supported by EMAgent.

Targets

EMAgent supports the following targets:

-
- Agent
 - Host
 - Database
 - Listener
 - Cluster Database (Oracle RAC)
 - Cluster

When EMAgent is started, EMAgent reads the TARGETS.XML file in the ORA_EMSTATE_EMD directory, and registers those targets with the Oracle Management Server. The status of each of the targets is reflected under the Targets tab of the EM Console. EMAgent monitors the registered targets on a periodic basis and uploads the status of the targets to the Oracle Management Server, which is reflected on the EM Console. In addition, a set of predefined metrics are also collected for each target and uploaded to the Oracle Management Server. Default thresholds are defined for each predefined metric on the Oracle Management Server. When a threshold is reached, an alert is generated and displayed on the console. This mechanism of automatic target monitoring replaces the Events model of earlier releases.

See Also: For more information about targets, refer to *Oracle Enterprise Manager Concepts 10g Release 2 (10.2)*

Jobs

The following jobs are supported by EMAgent:

- Operating System Command
- SQL Script

Known Limitations

EMAgent has the following known limitations:

1. Jobs will always indicate a successful completion status, as long as the agent has been able to create a detached process and run the command specified for the job. The status of job completion is not a reflection of the completion status of the job command. If the command fails for some reason, then the job itself will not be indicated as a failure, but the output of the job needs to be checked to verify if the command has succeeded or not.
2. The metric browser does not support all metrics. All metrics in the host metrics area are not applicable to Oracle Database. For more details, refer to the README_VMS_10GR2.TXT file shipped with the Oracle Grid Control EMAgent Kit.

Managing Oracle Database Port Numbers

During installation, OUI assigns port numbers to components from a set of default port numbers. This appendix lists the default port numbers and describes how to change the assigned port after installation. It includes information about the following topics:

- [About Managing Ports](#)
- [Viewing Port Numbers and Access URLs](#)
- [Port Numbers and Protocols of Oracle Components](#)
- [Changing the Oracle Enterprise Management Agent Port](#)

About Managing Ports

During installation, Oracle Universal Installer assigns port numbers to components from a set of default port numbers. Many Oracle Database components and services use ports. As an administrator, it is important to know the port numbers used by these services, and to make sure that the same port number is not used by two services on your host.

Most port numbers are assigned during installation. Every component and service has an allotted port range, which is the set of port numbers Oracle Database attempts to use when assigning a port. Oracle Database starts with the lowest number in the range and performs the following checks:

- Is the port used by another Oracle Database installation on the host?
The installation may be up or down 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, even a non-Oracle Database process.
- Is the port listed in the `TCPIP$ETC : SERVICES . DAT` ?

If the answer to any of the preceding questions is yes, Oracle Database moves to the next highest port in the allotted port range and continues checking until it finds a free port.

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. In addition, ports for some Oracle Database applications are

listed in the `PORTLIST.INI` file. This file is located in the `ORA_ROOT:[INSTALL]` directory.

If you change a port number, it is not updated in the `PORTLIST.INI` file, so you can only rely on this file immediately after installation. To find or change a port number, use the methods described in this appendix.

Port Numbers and Protocols of Oracle Components

The following table lists the 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 8–4 Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle SQL*Net Listener Allows Oracle client connections to the database over Oracle's SQL*Net protocol. You can configure it during installation. To reconfigure this port, use Net Configuration Assistant.	1521	1521	TCP
Data Guard Shares the SQL*Net port and is configured during installation. To reconfigure this port, use Net Configuration Assistant to reconfigure the Oracle SQL*Net listener.	1521 (same value as the listener)	1521	TCP
Oracle Management Agent HTTP port for Enterprise Management Agent. It is configured during installation. "Changing the Oracle Enterprise Management Agent Port" on page E-2 explains how to modify its port number	1832	1830–1849	HTTP
Oracle Clusterware CRS daemon (Oracle Clusterware Daemon) internode connection. The port number is assigned automatically during installation. You cannot view or modify it afterward.	49896	49896	TCP

Changing the Oracle Enterprise Management Agent Port

To find the current setting for the Oracle Management agent port, search for `EMD_URL` in the `ORA_ROOT:[host_sid].SYSMAN.CONFIG]EMD.PROPERTIES` file.

To change the Oracle Management Agent HTTP port, use the `emca -reconfig ports` command:

```
$EMCA -RECONFIG PORTS -AGENT_PORT 1831
```

Configuring Oracle Database Globalization Support

This appendix describes these Globalization Support topics:

About NLS_LANG Logical

Oracle provides Globalization Support that enables users to interact with a database in their preferred locale and character set settings. Setting the NLS_LANG logical name specifies locale behavior for Oracle software. It sets the language and territory used by the client application. It also sets the character set for entering and displaying data by a client program, such as SQL*Plus.

The NLS_LANG logical uses the following format:

```
$ DEFINE NLS_LANG=" language_territory.characterset"
```

where:

Parameter	Description
<i>language</i>	Specifies the language used for displaying Oracle messages, sorting, day names, and month names.
<i>territory</i>	Specifies the conventions for default date, monetary and numeric formats.
<i>characterset</i>	Specifies the encoding used by the client application (normally the Oracle character set that corresponds to the character set of the user terminal or the operating system)

The choices of locales available on your system depend on the locales installed. For more information on OpenVMS locale please refer to the Compaq C Run-Time Library Utilities Reference Manual.

The setting of NLS_LANG depends on the locale available in your session. Great care must be taken to set NLS_LANG correctly. An Incorrect NLS_LANG setting may lead to the input of incorrect data.

See Also: *Oracle Database Globalization Support Guide* for information about the NLS_LANG parameter and Globalization Support initialization parameters

The following table lists some of the valid values for the NLS_LANG environment variable.

Note: Refer to the operating system specific documentation on how to determine the operating system locale environment setting.

Operating system locale	NLS_LANG values
French (France)	FRENCH_FRANCE.WE8ISO8859P15, FRENCH_FRANCE.WE8ISO8859P1, FRENCH_FRANCE.WE8MSWIN1252, FRENCH_FRANCE.AL32UTF8
Japanese (Japan)	JAPANESE_JAPAN.JA16EUC, JAPANESE_JAPAN.JA16SJIS, JAPANESE_JAPAN.AL32UTF8

Troubleshooting

This appendix contains information about troubleshooting. It includes information about the following topics:

- [Section , "Verifying Requirements"](#)
- [Section , "X Window Display Errors"](#)
- [Section , "Errors During Installation"](#)
- [Section , "Reviewing the Log of an Installation Session"](#)
- [Section , "Troubleshooting Configuration Assistants"](#)
- [Section , "Silent Response File Error Handling"](#)
- [Section , "Cleaning Up After a Failed Installation"](#)

Verifying Requirements

Before performing any of the troubleshooting steps in this appendix, ensure that the system meets the requirements and that you have completed all of the preinstallation tasks specified in [Chapter 2, "Preinstallation Tasks"](#).

Read the Release Notes

Read the Release Notes for the product before installing it. The Release Notes are available on the Oracle Database DVD drive. The latest version of the Release Notes is also available on the Oracle Technology Network Web site at

<http://www.oracle.com/technology/documentation/index.html>

X Window Display Errors

If you are running Oracle Universal Installer on a remote system and you want to display the Oracle Universal Installer user interface on a local system, you may see error messages similar to the following:

```
%DECW-E-CANT_OPEN_DISPL, Can't open display
```

If you see the preceding error message, and the X server is running on a UNIX host, then:

1. In a local terminal window, log in as the user that started the X Window session.
2. Enter the following command:

```
$ xhost +
```

-
3. Enter the following commands, where *workstation_name* is the host name or IP address of your workstation:

- Bourne, Bash, or Korn shell:

```
$ DISPLAY=workstation_name:0.0
$ export DISPLAY
```

- C or tcsh shell:

```
% setenv DISPLAY workstation_name:0.0
```

4. To determine whether X Window applications display correctly on the local system, enter the following command:

```
$ xclock
```

The X clock should be displayed on the monitor.

5. If the X clock is displayed, then close X clock and start Oracle Universal Installer again.

Note: This procedure applies only to users of UNIX workstations. If you are using a PC or other system with X server software installed, refer to the X server documentation for information about how to permit remote systems to display X applications on the local system.

Errors During Installation

If you encounter an error during installation:

- Do not exit Oracle Universal Installer.
- If you clicked **Next** after you entered incorrect information about one of the installation screens, click **Back** to return to the screen and correct the information.
- If you encounter an error while Oracle Universal Installer is copying or linking files, refer to [Section , "Reviewing the Log of an Installation Session"](#).
- If you encounter an error while a configuration assistant is running, refer to [Section , "Troubleshooting Configuration Assistants"](#).
- If you cannot resolve the problem, remove the failed installation by following the steps listed in [Section , "Cleaning Up After a Failed Installation"](#).

Reviewing the Log of an Installation Session

During an installation, Oracle Universal Installer records all the actions that it performs in a log file. If you encounter problems during the installation, review the log file for information about possible causes of the problem.

To view the log file:

1. Determine the location of the `oraInventory` directory. This can be found as the value of the `inventory_loc` parameter in the `ORAINST.LOC` file, which is located in the `hostname` subdirectory of `SYS$LOGIN`.
2. Set default to the directory determined in step 1, and then to the `LOGS` subdirectory located there.

3. Obtain a directory listing to determine the file name of the log file. Installer log files have names similar to the following, where `date_time` indicates the date and time that the installation started:

```
INSTALLATIONSdate_time.LOG
```

This command lists the files in the order of creation, with the most recent file shown last. Installer log files have names similar to the following, where `date_time` indicates the date and time that the installation started:

```
installActionsdate_time.log
```

4. If an error displayed by Oracle Universal Installer or listed in the log file indicates a relinking problem, refer to the following file for more information:

```
ORA_ROOT:[INSTALL]MAKE.LOG
```

Troubleshooting Configuration Assistants

To troubleshoot an installation error that occurs when a configuration assistant is running:

- Review the installation log files listed in [Section , "Reviewing the Log of an Installation Session"](#).
- Review the specific configuration assistant log file located in the `ORA_ROOT:[CFGTOOLLOGS]` directory. Try to fix the issue that caused the error.
- If you see the `Fatal Error.Reinstall` message, look for the cause of the problem by reviewing the log files. Refer to [Section , "Fatal Errors"](#) for further instructions.

Configuration Assistant Failure

Oracle configuration assistant failures are noted at the bottom of the installation screen. The configuration assistant interface displays additional information if available. The configuration assistant execution status is stored in the following file:

```
device:[orainventory_location.LOGS]SILENTINSTALLdate_time.LOG
```

The execution status codes are listed in [Table 8-5](#):

Table 8-5 Execution Status Codes

Status	Result Code
Configuration assistant succeeded	0
Configuration assistant failed	1
Configuration assistant canceled	-1

Fatal Errors

If you receive a fatal error while a configuration assistant is running, you must remove the current installation and reinstall the Oracle software as follows:

1. Remove the failed installation as described in [Section , "Cleaning Up After a Failed Installation"](#).
2. Correct the cause of the fatal error.
3. Reinstall the Oracle software.

Silent Response File Error Handling

To determine whether a silent installation succeeds or fails, refer to the following log file:

```
device: [orainventory_location.LOGS]SILENTINSTALLdate_time.LOG
```

If necessary, refer to the [Section , "Reviewing the Log of an Installation Session"](#) for information about determining the location of the oraInventory directory.

A silent installation fails if:

- You do not specify a response file
- You specify an incorrect or incomplete response file
- Oracle Universal Installer encounters an error, such as insufficient disk space

Oracle Universal Installer or configuration assistant validates the response file at run time. If the validation fails, then the silent installation or configuration process ends. Oracle Universal Installer treats values for parameters that are of the wrong context, format, or type as if no value was specified in the file.

Cleaning Up After a Failed Installation

If an installation fails, you must remove files that Oracle Universal Installer created during the attempted installation and remove the Oracle home directory. Perform the following steps to remove the files:

1. Start Oracle Universal Installer as described in [Section , "Installing Oracle Database from the Hard Drive"](#).
2. Click **Deinstall Products** on the Welcome screen or click **Installed Products** on any Installer screen.
The Inventory screen is displayed, listing installed products.
3. Select the Oracle home that contains the products that you want to remove, then click **Remove**.
4. Manually remove the Oracle home directory created during the failed installation.
5. Reinstall the Oracle software.

Apache Server Installation and Configuration

This appendix lists the procedure and steps to install and configure Apache Server for OpenVMS. The following topics are included:

- [Section , "Postinstallation Checklist"](#)
- [Section , "Test the Installation"](#)
- [Section , "Running Oracle HTTP Server on OpenVMS"](#)
- [Section , "Security Information"](#)
- [Section , "Open Source Licenses"](#)

Postinstallation Checklist

After you configure Oracle HTTP Server for OpenVMS, perform the following tasks to ensure a successful startup:

- [Section , "Running AUTOGEN"](#)
- [Section , "Check Disk Quota"](#)
- [Section , "Checking for SET TERMINAL/INQUIRE"](#)

Each of these tasks is explained in the following sections. Once you have completed these, you can test the installation by starting Oracle HTTP Server.

Running AUTOGEN

After the installation, when you have a normal system workload running on your machine, run `SYS$UPDATE:AUTOGEN.COM (AUTOGEN)` to evaluate the system parameters and make adjustments based on the hardware configuration and system workload. On Oracle HTTP Server, AUTOGEN will probably increase the page file size and the number of swap file pages.

Check Disk Quota

If the disk quota is too low, then Oracle HTTP Server will not start. Either raise the disk quota for the Oracle Database account or grant the account the EXQUOTA privilege, which enables it to bypass disk quota restrictions. Use the following commands:

```
$ SHOW QUOTA/USER=[server-uic]/DISK=device_name
$ SET PROCESS/PRIVILEGES=EXQUOTA node-name::ORACLE
```

Checking for SET TERMINAL/INQUIRE

When Oracle HTTP Server for OpenVMS is started, the following login files are run:

- `SYS$MANAGER:SYLOGIN.COM`
- `LOGIN.COM` (login for the Oracle Database account)

Check these files to ensure that any `SET TERMINAL/INQUIRE` statements are processed only in the `INTERACTIVE` mode. For example:

```
$ IF F$MODE() .EQS "INTERACTIVE" THEN $ SET TERMINAL/INQUIRE
```

If you do not check for this, then the HTML that is sent to clients may not be well-formed and may be sent in an intermittent fashion. This problem may also appear when running CGI scripts.

Test the Installation

You must manually start Oracle HTTP Server to verify the installation and configuration of the server. Enter the following command:

```
$ APACHECTL START
```

Perform the following tasks to test the installation:

- [Section , "Browser Test"](#)
- [Section , "Troubleshooting"](#)

Browser Test

You can test the installation using a Web browser. Replace *host.domain* in the following URL with the corresponding information about Oracle HTTP Server that you installed:

```
HTTP://host.domain:port
```

If this is a new installation, then the browser should display the standard introductory screen with the following bold text at the top:

```
"Welcome to Oracle HTTP Server."
```

The Apache logo is displayed at the bottom.

Troubleshooting

If you do not receive a response from Oracle HTTP Server, check the following:

- In the `SYS$MANAGER:SYLOGIN.COM` file, ensure that there is no `SET TERMINAL/INQUIRE` statement for network processes.
- Look for the following files:

```
APACHE$ROOT: [000000]APACHE$SERVER.LOG  
APACHE$ROOT: [LOGS]ERROR_LOG
```

Running Oracle HTTP Server on OpenVMS

The default port for Oracle HTTP Server is port 7777. If this port is already in use by another application, or if you would like to use a different port, then modify the `HTTPPD.CONF` file, located in the `ORA_`

ROOT: [APACHE.APACHE.SPECIFIC.host.CONF] directory, to specify a different port number

The following subsections describes the process of running the Oracle HTTP server:

- [Section , "Starting and Stopping the Server"](#)
- [Section , "Server Log File"](#)
- [Section , "Customizing the Server Environment"](#)
- [Section , "How to Configure Apache"](#)
- [Section , "Modules and Directives"](#)
- [Section , "Supported and Unsupported Features"](#)
- [Section , "File Formats"](#)
- [Section , "File Naming Conventions"](#)
- [Section , "File Transfer Process and Access Control List"](#)
- [Section , "Logical Names"](#)
- [Section , "OpenVMS Cluster Considerations"](#)
- [Section , "CGI Programs"](#)

Starting and Stopping the Server

To start Oracle HTTP Server, enter the following command:

```
$ APACHECTL STARTUP
```

To stop Oracle HTTP Server, enter the following command:

```
$ APACHECTL STOP
```

Server Log File

The server log file for APACHE\$WWW is written to:

```
APACHE$SPECIFIC: [000000] APACHE$SERVER.LOG
```

Performance Considerations

You should have prior experience in tuning the performance of the OpenVMS operating system. For information about OpenVMS performance, refer to OpenVMS system documentation

Recommendations for improving performance on Oracle HTTP Server are provided in this appendix and the Release Notes.

Limits and Quotas for Light to Moderate Load

[Table 8–6](#) shows sample values for the Oracle Database account from a working and exercised Oracle HTTP Server with a light to moderate load. These values are presented as an example of a system performing well within its context.

If you should experience performance difficulties, refer to this table for guidelines about making adjustments. Remember that no one set of values will be appropriate for all situations.

Table 8–6 Sample Values for the oracle account

Parameter	Default	Sample value for Oracle HTTP Server
ASTLM (NonPooled) Total number of asynchronous system trap (AST) operations and scheduled wake-up requests the user can have queued at one time	250	610 Or BIOLM + DIOLM + 10
BIOLM (NonPooled) Number of outstanding buffered I/O operations permitted for a user process	150	300 You may also need to increase the SYSGEN parameter CHANNELCNT because it limits BIOLM,DIOLM, and FILLM.
BYTLM (Pooled) Amount of buffer space a user process can use	64000	200000 Increase this value for a heavy load.
CHANNELCNT	256	256 CHANNELCNT must be greater than or equal to FILLM
DIOLM (NonPooled) Number of outstanding direct I/O operations permitted for a user process	150	300 You may also need to increase the SYSGEN parameter CHANNELCNT because it limits BIOLM,DIOLM, and FILLM.
ENQLM (Pooled) Specifies the lock queue limit	2000	2000
FILLM (Pooled) Number of files a user process can have open at one time This includes the number of network logical links that can be active at the same time.	100	300 Increase this value for a heavy load. You may also need to increase the SYSGEN parameter CHANNELCNT because it limits BIOLM,DIOLM, and FILLM.
JTQUOTA (Pooled) Byte quota for the job-wide logical name table	4096	8192
PGFLQUO (Pooled) Number of pages the user process can use in the system page file	50000	250000 If you increase PGFLQUO, then you should monitor the free size of the system page and swap files, because these may need to be increased.
PRCLM (Pooled) Number of subprocesses a user process can create	8	20 You should increase this value for a heavy load.
TQELM (Pooled) Number of entries a user process can have in the timer queue or the number of temporary common event flag clusters a user process can have.	10	610 Or BIOLM + DIOLM + 10

Server with Medium to High Load

After you install the server and run it, look in the log file for errors of the "cannot open" type. Errors of this type often indicate that you need to modify system parameters.

Try the following:

- Set `FILLM` to limit the number of files that a user process can have open.
- Set the `SYSGEN` parameter, `CHANNELCNT`, to 1024 (unless it is already set to a higher value).

Note: Whenever you change system parameters, you must restart the system to enable the new settings.

Excessive File Build Up

A large number of `.LOG` and `.PID` files can amass over time in the `APACHE$ROOT:[0000000]` and `APACHE$ROOT:[LOGS]` directories. Purging these files can become a burden on application or system managers. System managers should manually use explicit `SET DIRECTORY/VERSION` commands on these two directories.

Customizing the Server Environment

The installation procedure creates a file named `HTTPD.CONF` and places it in `APACHE$ROOT:[CONF]`. The `HTTPD.CONF` file stores information that Oracle HTTP Server uses to set up the server environment. The `HTTPD.CONF` file has been tailored to use OpenVMS syntax, but its overall functionality is essentially identical to `httpd.conf` on the UNIX platform.

The `HTTPD.CONF` file contains an explanation for each line that it can process. You can refer to these explanations when customizing the file for your environment. You can also refer to any generally available Apache documentation on `HTTPD.CONF`.

Note the following about `HTTPD.CONF` on OpenVMS:

- `MOD_OSUSCRIPT` has been added to enable CGI scripts that were originally written for the OSU server.
- UNIX-style path names are recognized by OpenVMS. You can use either UNIX-style or OpenVMS-style path names in the configuration file. However, you cannot mix the two styles within a specification.
- In an OpenVMS Cluster, you can specify either clusterwide or system-specific files.

How to Configure Apache

Perform the following steps to configure Apache to run while using an account other than the Oracle Database account:

1. Modify the `HTTPD.CONF` file to include the following line:

```
User username
```

2. Modify `APACHEUSER.COM` to set the logical `APACHE_USERNAME` to the required user name. Ensure that the user name is exactly the same as the entry in the `HTTPD.CONF` file.

-
3. Restart the Apache Server if it is already running.

Modules and Directives

The Oracle HTTP Server provides the modules and directives that are provided by the HP Secure Web Server for OpenVMS (based on Apache), Version 1.3-1. Please refer to the HP documentation for that product for more information. All supported modules and directives function as documented by the Apache Software Foundation at:

<http://www.apache.org/docs>

Supported and Unsupported Features

Information about running Oracle HTTP Server that is specific to running the server on OpenVMS is provided in the following sections.

See Also: For more information about on the Apache server, refer to the Apache Software Foundation Web site at

<http://www.apache.org/docs/>

Modules Not Included

The following modules are not included in this version of Oracle HTTP Server:

- MOD_OSNINT
- MOD_OSSL1
- MOD_PERL
- MOD_PHP
- MOD_PROXY

Unsupported Directives

For information about directives that are not supported, refer to the HP documentation for the product HP Secure Web Server for OpenVMS (based on Apache), Version 1.3-1.

- AgentLog
- AllowCONNECT
- Anonymous
- Anonymous_Authoritative
- Anonymous_LogEmail
- Anonymous_MustGiveEmail
- Anonymous_NoUserID
- Anonymous_VerifyEmail
- AuthDBAuthoritative
- AuthDBGroupFile
- AuthDBMAuthoritative
- AuthDBMGroupFile
- AuthDBUserFile

-
- AuthDBMUserFile
 - AuthDigestFile
 - CacheDefaultExpire
 - CacheDirLength
 - CachedirLevels
 - CacheForceCompletion
 - CacheGcInterval
 - CacheLastModifiedFactor
 - CacheMaxExpire
 - CacheRoot
 - CacheSize
 - CheckSpelling
 - CookieExpires
 - CookieTracking
 - Example
 - ExpiresActive
 - ExpiresByType
 - ExpiresDefault
 - Header
 - Metadir
 - MetaFiles
 - MetaSuffix
 - MimeMagicFile
 - MMapFile
 - NoCache
 - ProxyBlock
 - ProxyDomain
 - ProxyPass
 - ProxyPassReverse
 - ProxyReceiveBufferSize
 - ProxyRemote
 - ProxyRequests
 - ProxyVia
 - RefererIgnore
 - RefererLog
 - RewriteBase
 - RewriteCond

-
- RewriteEngine
 - RewriteLock
 - RewriteLog
 - RewriteLogLevel
 - RewriteMap
 - RewriteOptions
 - RewriteRule
 - ScriptInterpreterSource
 - VirtualDocumentRoot
 - VirtualDocumentRootIP
 - VirtualScriptAlias
 - VirtualScriptAliasIP

Command-Line Options

This section describes the HTTPD command-line options supported on Oracle HTTP Server.

Then you can use the following format to enter a command-line option:

```
$ HTTPD -option
```

where `-option` is one of the following command line options:

- `-v`
Displays the HTTPD version and its build date.
- `-V`
Displays the HTTPD base version, its build date, and a list of compile settings that influence the performance of the server.
- `-h`:
Displays a list of the HTTPD options.
- `-l`:
Displays a list of all modules compiled into the server.
- `-L`:
Displays a list of directives with expected arguments and instances where the directive is valid.

The following example shows how to enter the `"L"` option to list the available configuration directives:

```
$ HTTPD "-L"
```

Virtual Host Support

The term, **virtual host**, refers to the practice of maintaining a single server to serve pages for multiple virtual hosts. Both IP-based and name-based virtual host support are available on Oracle HTTP Server for OpenVMS.

Note: The security profile of the running server is the same on all virtual hosts.

For more information about virtual hosts, refer to the Apache Software Foundation documentation at

<http://www.apache.org/docs/vhosts/index.html>

Dynamic Shared Object Support

Dynamic shared object support provides a method to format code so that it will load into the address space of an executable program at run time. For more information about dynamic shared object support, refer to the Apache Software Foundation documentation at

<http://www.apache.org/docs/dso.html>

File Handlers

Oracle HTTP Server for OpenVMS supports the ability to use file handlers explicitly. For more information about file handlers, refer to the Apache Software Foundation documentation at

<http://www.apache.org/docs/handler.html>

Content Negotiation

The MOD_NEGOTIATION module provides content negotiation. This module enables you to specify language variants of HTML files. To specify language variants, use an underscore instead of a period before the language extension.

For example:

- On UNIX, `filename.html.fr` is the French variant of `filename.html`.
- On OpenVMS, `FILENAME.HTML_FR` is the French variant of `FILENAME.HTML`.

For more information about content negotiation, refer to the Apache Software Foundation documentation at

<http://www.apache.org/docs/content-negotiation.html>

Apache API

You can use the standard Apache application programming interface (API) to write user-defined modules that run on Oracle HTTP Server. For more information about the Apache API, refer to the Apache Software Foundation documentation at

<http://www.apache.org/docs/misc/API.html>

suEXEC Support

The suEXEC feature provides the ability to run CGI programs under user IDs that are different from the user ID of the calling Web server. This is not supported by Oracle HTTP Server for OpenVMS.

File Formats

All file formats are supported. However, the Web browser status bar will not show page loading progress for Variable or VFC format files larger than 8 KB.

Page loading progress relies on an accurate byte count, which is not readily available for files in Variable or VFC format. For files in these formats, Oracle HTTP Server must count the bytes as the files load. The counting process can slow performance, so it has been turned off in this situation.

File Naming Conventions

In general, users running Oracle HTTP Server for OpenVMS can specify either UNIX-style file names or OpenVMS-style file names. Oracle HTTP Server usually displays UNIX-style file names.

The On-Disk Structure Level 5 (ODS-5) volume structure, shipped with OpenVMS version 8.2, supports long file names, enables the use of a wider range of characters within file names, and preserves case within file names. However, the DEC C RTL that is shipped with OpenVMS Alpha version 7.2-1 does not provide full support for extended file names on ODS-5 devices. This lack of full support imposes certain restrictions on users running Oracle HTTP Server for OpenVMS.

Because mixed UNIX-style and OpenVMS-style extended file names are not yet supported by the DEC C RTL, you may be required to use UNIX-style syntax when interacting with Oracle HTTP Server. An example would be appending additional directories or a file name to a root.

The following examples illustrate mixed UNIX-style and OpenVMS-style file names that are not supported in OpenVMS version 8.2:

```
doc/foo.bar.bar
./tmp/foo.bar.b^_ar
~foo^.bar
```

You can, however, modify the last example so that it works as an OpenVMS extended file name that has a tilde (~) as the first character. Precede the leading tilde (~) with the Extended File Specifications escape character (^) as shown in the following example:

```
^~foo^.bar
```

For more information about using the tilde (~) in OpenVMS extended file names, refer to the OpenVMS Guide to Extended File Specifications at the following Web site:

<http://h71000.www7.hp.com/doc/73final/6536/6536PRO.HTML>

File Transfer Process and Access Control List

When performing an FTP operation, ensure that the access control list (ACL) for the target directory on Oracle HTTP Server has FTP access enabled as follows:

When transferring new files:

```
$ SET SECURITY/ACL=(IDENTIFIER=yourFTPname,ACCESS=READ+WRITE) [directory]
```

When replacing existing files:

```
$ SET SECURITY/ACL=(IDENTIFIER=yourFTPname,ACCESS=READ+WRITE) [directory]*.*
```

Logical Names

Oracle HTTP Server for OpenVMS creates the following logical names, which are listed with their descriptions in table [Table 8-7](#).

Table 8–7 Oracle HTTP Server Logical Names and Their Descriptions

Logical Name	Description
APACHE\$COMMON	Concealed logical name that defines clusterwide files in APACHE\$ROOT (device: [APACHE])
APACHE\$FIXBG	System executive mode logical name pointing to installed, shareable images. Not intended to be modified by the user.
APACHE\$HTTPD_SHR	System executive mode logical name pointing to installed, shareable images. Not intended to be modified by the user.
APACHE\$INPUT	Used by CGI programs for PUT and POST methods of reading the input stream.
APACHE\$PLV_ENABLE_ username	System executive mode logical name defined during startup and used to control access to the services provided by the APACHE\$PRIVILEGED image. Not intended to be modified by the user.
APACHE\$PLV_LOGICAL	System executive mode logical name defined during startup and used to control access to the services provided by the image. Not intended to be modified by the user.
APACHE\$PRIVILEGED	System executive mode logical name pointing to installed, shareable images. Not intended to be modified by the user.
APACHE\$ROOT	System executive mode logical name defined during startup that points to the top-level directory. (device: [APACHE], device: [APACHE.SPECIFIC.node-name])
APACHE\$SPECIFIC	Concealed logical name that defines system-specific files in APACHE\$ROOT (device: [APACHE.SPECIFIC.node-name])
APACHE\$CGI_MODE	System logical name that controls how CGI environment logicals are defined in the running CGI process. There are three different options. Note that only one option is available at a time. 0: Default. Environment logicals are defined as local symbols and are truncated at 970 (limitable with DEC C). 1: Environment logicals are defined as local symbols unless they are greater than 970 characters. If the environment value is greater than 970 characters, then it is defined as a multi-item logical. 2: Environment logicals are defined as logicals. If the environment value is greater than 512 characters, then it is defined as a multi-item logical.
APACHE\$DEBUG_DCL_CGI	If defined, this system logical name enables APACHE\$VERIFY_DCL_CGI and APACHE\$SHOW_CGI_SYMBOL.
APACHE\$VERIFY_DCL_CGI	If defined, this system logical name provides information for troubleshooting DCL command procedure CGIs by forcing a SET VERIFY before running any DCL CGI. Use with APACHE\$DEBUG_DCL_CGI.
APACHE\$SHOW_CGI_SYMBOL	If defined, this system logical name provides information for troubleshooting the CGI environment by dumping all the symbols and logicals (job/process) for a given CGI. Use with APACHE\$DEBUG_DCL_CGI.

Table 8–7 (Continued) Oracle HTTP Server Logical Names and Their Descriptions

Logical Name	Description
APACHE\$PREFIX_DCL_ CGI_SYMBOLS_WWW	If defined, this system logical name prefixes all CGI environment logical symbols with WWW_. By default no prefix is used.
APACHE\$CREATE_ SYMBOLS_GLOBAL	If defined, this system logical name causes CGI environment symbols to be defined globally. They are defined locally by default.
APACHE\$CGI_USE_ DCLCOM_FOR_IMAGES	If defined, this system logical name forces CGI images to run within a DCL process. The default is to run CGI images directly. (Note: Direct running of CGI images is not currently supported.)
APACHE\$DL_NO_ UPPERCASE_FALLBACK	If defined to be true (1, T, or Y), this system logical name disables case-insensitive symbol name lookups whenever case-sensitive lookups fail. Refer to APACHE\$DL_FORCE_UPPERCASE.
APACHE\$DL_FORCE_ UPPERCASE	If defined to be true (1, T, or Y), this system logical name forces case-sensitive dynamic image activation symbol lookups. By default, symbol lookups are first done in a case-sensitive manner, and then if failed, a second attempt is made by using case-insensitive symbol lookups. This fallback action can be disabled with APACHE\$DL_NO_UPPERCASE_FALLBACK.

OpenVMS Cluster Considerations

An OpenVMS Cluster is a group of OpenVMS systems that work together as one virtual system. Oracle HTTP Server runs in an OpenVMS Cluster so that you can take advantage of the resource sharing that increases the availability of services and data.

Bear the following points in mind:

- Oracle HTTP Server is supported on OpenVMS Version 8.2-1 or later.
- Oracle HTTP Server runs in an Alpha Itanium, or in a mixed architecture cluster, separate Apache installations are required for Alpha and Itanium.

Individual System Versus Clusterwide Definition

To define clusterwide versus individual configuration files, APACHE\$ROOT uses the following concealed logical names:

- APACHE\$COMMON defines clusterwide files
- APACHE\$SPECIFIC defines system-specific files

When reading a file, the server first looks for a system-specific version of the file in APACHE\$SPECIFIC: [*directory*]. If it does not find one, then it looks for a clusterwide file in APACHE\$COMMON: [*directory*].

To avoid confusion, always use the appropriate concealed logical name to specify the file that you want to edit. For example, to edit a clusterwide version of HTTPD.CONF, refer to:

```
$ EDIT APACHE$COMMON: [CONF]HTTPD.CONF
```

If you refer to:

```
$ EDIT APACHE$ROOT: [CONF]HTTPD.CONF
```

then the server would open the clusterwide file but save it as a system-specific version. The latest version of `HTTPD.CONF` would then be visible only to the individual node on which it was saved.

Within `HTTPD.CONF` itself, you should make this distinction whenever you refer to a path or to a file location. This improves performance and ensures that the server will return a complete directory listing. For example, you should specify `APACHE$COMMON` or `APACHE$SPECIFIC` (instead of `APACHE$ROOT`) with `directory` directives.

The following extract, from the `HTTPD.CONF` file, refers to `APACHE$COMMON`, because the content for the default Web page is in the clusterwide directories.

```
DocumentRoot "/apache$common/htdocs"
...
<Directory "/apache$common/htdocs">
Options Indexes FollowSymLinks Multiviews
AllowOverride None
Order allow,deny
Allow from all
</Directory>
```

If there was content for one specific node in a cluster, then the `APACHE$SPECIFIC` logical name would be used.

Mixed-Architecture Cluster

In a mixed-architecture cluster containing VAX nodes, do not use a cluster alias IP address with Oracle HTTP Server. Because the VAX systems will not have Oracle HTTP Server running, they will not be able to service HTTP requests.

CGI Programs

Common gateway interface (CGI) programs run within the DCL shell on Oracle HTTP Server for OpenVMS. This section discusses the following CGI topics:

- [Section , "CGI Environment Logical"](#)
- [Section , "Referencing Input"](#)
- [Section , "Running CGI Images"](#)
- [Section , "Logical Names for Debugging CGI Scripts"](#)
- [Section , "Displaying Graphics with CGI Command Procedures"](#)

CGI Environment Logical

By default, an environment logical symbol takes the form that is designated by the name of the environment logical. You can determine how environment logicals are set when the server runs a CGI program. You can define the `APACHE$PREFIX_DCL_CGI_SYMOBLS_WWW` logical name to prefix all environment logical symbols with `WWW_`. By default, no prefix is used.

The `APACHE$CGI_MODE` logical name controls how CGI environment logicals are defined in the running CGI program as follows:

```
APACHE$CGI_MODE    option
```

where `option` can have one of the following values at a time:

- 0: Default. Environment logicals are defined as local symbols and are truncated at 970 (limitable with DEC C).

- 1: Environment logicals are defined as local symbols unless they are greater than 970 characters. If the environment value is greater than 970 characters, it is defined as a multi-item logical.
- 2: Environment logicals are defined as logicals. If the environment value is greater than 512 characters, it is defined as a multi-item logical.

APACHE\$DCL_ENV is a foreign symbol that lets you define CGI environment logical, as follows:

```
APACHE$DCL_ENV [-c] [-d] [-e env-file]
```

where:

- -c: Default. Indicates create environment logicals.
- -d: Indicates delete environment logicals.
- -e env-file: Specifies an alternate environment file.

The environment file does not need to be specified by the caller because the parent derives it (it can be easily determined by default).

The following example deletes the environment and then re-creates it:

```
Example: diff_mode.cgi.com
$ APACHE$DCL_ENV -d
$ Define APACHE$PREFIX_DCL_CGI_SYMBOLS_WWW 1
$ APACHE$DCL_ENV -c
```

Referencing Input

CGI scripts that reference input to Oracle HTTP Server must refer to APACHE\$INPUT.

Running CGI Images

On OpenVMS, CGI images run within a DCL process. You cannot run CGI images directly.

Logical Names for Debugging CGI Scripts

Use the following logical names to debug CGI scripts:

Logical Name	Description
APACHE\$DEBUG_DCL_CGI	If defined, this system logical name enables APACHE\$VERIFY_DCL_CGI and APACHE\$SHOW_CGI_SYMBOL.
APACHE\$VERIFY_DCL_CGI	If defined, this system logical name provides information for troubleshooting DCL command procedure CGIs by forcing a SET VERIFY before running any DCL CGI. Enabled by APACHE\$DEBUG_DCL_CGI.
APACHE\$SHOW_CGI_SYMBOL	If defined, this system logical name provides information for troubleshooting the CGI environment by dumping all of the symbols and logicals (job/process) for a given CGI. Enabled by APACHE\$DEBUG_DCL_CGI.

Displaying Graphics with CGI Command Procedures

To display a graphics file with a CGI command procedure, use the APACHE\$DCL_BIN foreign symbol in the following format:

```
APACHE$DCL_BIN [-s bin-size] bin-file
```

where:

- `-s bin-size`: Specifies the actual or approximate file size in bytes. The value of `bin-size` is automatically determined if the image file is larger than 32768 KB (default value). If the image file is smaller than 32768 KB, then you can provide an approximate (or actual) size. This boosts performance.
- `bin-file`: Specifies the file to be displayed.

For example:

```
$ SAY := WRITE SYS$OUTPUT
$ SAY "Content-type: image/gif"
$ SAY ""
$ APACHE$DCL_BIN APACHE$ROOT: [ICONS]APACHE_PB.GIF
$ EXIT
```

Security Information

Oracle HTTP Server for OpenVMS is a nonprivileged, user-mode, socket-based network application. `TMPMBX` and `NETMBX` are the only privilege requirements. The server runs under its own unique UIC and user account (`APACHE$WWW`).

Process Model

Oracle HTTP Server runs as a single job that consists of:

- One master process (`APACHE$WWW`)
- Several subprocesses

Subprocesses are created to service incoming HTTP requests and to run CGI scripts.

Because the server runs as a single job, the OpenVMS security profile for each process is identical and no enhanced mechanism is required for these processes to communicate with one another. Resource utilization is controlled by a single user account (`oracle`) where pooled quotas are defined.

Privileged Images

Oracle HTTP Server performs three operations that require additional privileges:

- Binding to a port below 1024 (privileged ports)
By default, the server binds to port 8080 (HTTP).
- Fetching path information for other users
The server provides a replacement for the `getpwnam` C RTL routine to enable the server to fetch default path information for other users (required by `MOD_UTIL` and `MOD_USERDIR`).
- Changing the carriage-control attribute on socket (BG) devices
The server also enables or disables (or both) the carriage-control attribute on BG (socket) devices for certain stream operations.

Two protected, shareable images are installed at startup to enable the server to perform the following functions:

- `APACHE$PRIVILEGED.EXE` (exec-mode services)

-
- `APACHE$FIXBG.EXE` (kernel-mode services)

The `APACHE$PRIVILEGED.EXE` image provides exec-mode services for binding to privileged sockets and fetching user default path information. Access to these services is limited to processes running under the `oracle` username and is controlled by the `APACHEPLV_ENABLE_APACHEWWW` logical name. This logical name is defined as:

```
"APACHE$PLV_ENABLE_APACHE$WWW" = "3,80,1023"
```

The "3,80,1023" string represents three parameters where:

- The first parameter (3) is a bit-mask that enables or disables the two services:
 - Binding to privileged ports
 - Fetching user default path information
- The second and third parameters indicate the minimum and maximum port that are allowed to be bound.

When a call to either service is made, the service code does the following:

1. Temporarily enables the `SYSPRV`, `OPER`, `SYSNAM`, and `NETMBX` privileges
2. Performs the function
3. Restores the process original privileges

The `APACHE$FIXBG.EXE_ALPHA` image provides a kernel-mode service for manipulating the carriage-control attribute for BG devices that are owned by the calling process. No special access control exists on this service. This function can also be performed using a `setsockopt` C RTL run-time call, but it is not supported by all TCP/IP stack vendors, which is the reason this service exists. This service does not enable privileges, but runs in kernel mode.

Privileges Required to Start and Stop the Server

Oracle HTTP Server runs under the `oracle` username and UIC and is started as a detached, network process. During startup, protected images are installed and logical names are placed in the system logical name table. Shutdown is accomplished by sending a `KILL` signal to the master process and its subprocess.

In order to startup the Oracle HTTP Server, the following privileges are required:

- `SYSPRV`
- `SYSNAM`
- `IMPERSONATE`
- `BYPASS`
- `CMKRNL`
- `ALTPRI`
- `WORLD`

The privileges `ALTPRI` and `BYPASS` are not essential privileges for installing and running an Oracle database. These privileges can either be added to the `oracle` account, or a separate account can be created to maintain and run the Oracle HTTP Server.

If you have already configured Oracle HTTP Server using the `oracle` account and want to run under a different account, then the file `ORA_`

ROOT: [APACHE.APACHE.SPECIFIC.node.CONF] HTTPD.CONF must be modified to change the USER parameter before attempting to start it up.

File Ownership and Protection

All the server files reside under the root directories that the APACHE\$ROOT logical name points to. During installation, file protection is set to (S:RWED, O:RWED, G,W). During configuration, all files are set to be owned by the oracle user.

Server Extensions (CGI Scripts)

Server extensions, such as CGI scripts, run within the context of Oracle HTTP Server process or its subprocesses. These extensions have complete control over the server environment. You can configure the server to enable processing of arbitrary user scripts, but standard practice is to limit such activity to scripts that are written by completely trusted users. Oracle HTTP Server includes directives that enable a Web administrator to control script execution and client access. The use of these directives is described in numerous books and is not duplicated here.

suEXEC Not Available for Protecting Script Execution

Oracle HTTP Server for OpenVMS does not currently support the suEXEC method of running scripts under the username that owns the script. Many sites use this feature to allow execution of arbitrary, user-written scripts without the fear of compromising the server environment.

Open Source Licenses

This section provides open source license acknowledgments and license references.

Apache

This product includes software developed by the Apache Software Foundation. You can visit the Web site of this organization at

<http://www.apache.org/>

You can view the license at the following Web site

<http://www.apache.org/licenses/LICENSE-2.0>

This product also includes software that is developed by Hewlett-Packard.



Identifying Listener Processes

The following set of commands can be used to display information about all listener processes:

```
$!  
$! To identify and report on any active SQL*Net listener processes  
$! on this OpenVMS Server (used as part of 10g pre-installation  
$! check activities)  
$!  
$ found_any = 0  
$ say = "write sys$output"  
$!  
$ say " "  
$ say " Collecting system information - please wait..."  
$!  
$! First, get a list of all disk on this system  
$!  
$ pipe show dev d/mount > sys$scratch:lsnrchk_devices.tmp  
$!  
$ close/nolog flsnr  
$ open/read flsnr sys$scratch:lsnrchk_devices.tmp  
$!  
$! Skip the records we know we don't need  
$!  
$ read flsnr record  
$ read flsnr record  
$ read flsnr record  
$!  
$! Loop on each disk we found  
$!  
$ loop:  
$ read flsnr/end=done_it record  
$ dev = f$element(0," ", record)  
$!  
$! Look for any processes accessing tnslnsr.exe  
$!  
$ pipe show device/files 'dev' | search/nowarn -  
  
$ sys$input tnslnsr > sys$scratch:lsnr_chk1.tmp  
$! If we found something, check the process(es) returned  
$!  
$ if $status .eqs. "%X1000001"  
$ then  
$   close/nolog ttmp  
$   open/read ttmp sys$scratch:lsnr_chk1.tmp  
$loop1:  
$   read/end=done_it_1 ttmp record
```

```

$   if record .nes. ""
$   then
$       record = f$edit(record,"COMPRESS,TRIM,UPCASE")
$! Skip tnslnsr.com records
$       if f$locate(".COM;",record) .ne. f$length(record) then goto loop1
$       proc = f$element(1," ",record)
$!
$! The following check ensures we skip records that we don't want
$! (for example, records that reflect that we have a installed image)
$!
$       if proc .nes. "" .and. f$locate(".EXE",proc) .eq. f$length(proc)
$       then
$           img = f$getjpi("'"proc',"IMAGENAME")
$           lsnr_loc = f$extract(0,f$locate(".]",img),img)
$           lsnr_loc = f$extract(0,f$locate(".NETCONFIG]",lsnr_loc),lsnr_loc)
$           lsnr_loc = lsnr_loc + ".NETWORK.ADMIN]"
$           prcnm = f$getjpi("'"proc',"PRCNAM")
$!
$! Display the result
$!
$       if found_any .eq. 0
$       then
$           say " "
$           say " The following listener(s) need to be checked : "
$           say " "
$           say " If they use the IPC Protocol with (key = EXTPROC) and/or "
$           say " the TCP Protocol with (Port = 1521) then they need to "
$           say " be shutdown when installing Oracle 10g"
$           say " "
$       endif
$       say "======"
$       say " Listener process name          : 'prcnm'"
$       say " VMS pid                          : 'proc'"
$       say " Probable LISTENER.ORA location : 'lsnr_loc'"
$       found_any = found_any + 1
$       endif
$       goto loop1
$   endif
$ done_it_1:
$   close/nolog sys$scratch:lsnr_chk1.tmp
$ endif
$ goto loop
$!
$ done_it:
$ close/nolog flsnr
$! Remove all work files
$!
$ delete/noconfirm sys$scratch:lsnrchk_devices.tmp;*
$ delete/noconfirm sys$scratch:lsnr_chk1.tmp;*
$!
$ if found_any .eq. 0
$ then
$   say " "
$   say "       No active SQL*Net Listeners detected on this node"
$   say " "
$ else
$   say "======"
$   say " "
$ endif
$!

```

```
$ say " End of processing"  
$ say " "  
$ exit
```


Numerics

64-bit database feature, Very Large Memory (VLM),
main topic, A-1

A

account quotas
setting, 2-14

accounts
reviewing accounts and passwords, 7-1
setting account privileges, 2-14
setting account quotas, using the AUTHORIZE
utility, 2-14
unauthenticated access through HTTP, 7-4

administrator
disk space requirements for Oracle Database
Client, 2-3

Advanced
configuration type, 5-11

Advanced configuration type, 5-10

advanced database type, 5-15

AL24UTFSS character sets, upgrading issues, 1-13

ANONYMOUS user
unauthenticated account access through
HTTP, 7-4

archive logs
destinations, conversion to multi-instance, C-2

authentication methods
Kerberos, 2-3
Secure Socket Layer, 2-3

AUTHORIZE utility, 2-12, 2-13

AUTOGEN utility
considers pre-allocated reserved memory when
tuning the system, 2-17
for setting or modifying SYSGEN
parameters, 2-18

automatic undo management, 6-14

automating startup and shutdown, 4-5

Available
service configuration policy, 6-5

Available Product Components screen
Oracle Universal Installer, 5-14

B

background process
crsd, 5-9
evmd, 5-9
ocssd, 5-9

backups
conversion to multi-instance, C-2

Basic
TAF failover policy, 6-5

Big Oracle Blocks (BOB), suggested parameter
settings for VLM or 64-bit feature, A-2

buffers, DB_BLOCK_BUFFERS, recommendations for
64-bit feature (VLM), A-3

C

cache hit ratio, Oracle buffer cache, suggested
parameter settings for 64-bit feature (VLM), A-2

cache, MRU (most recently used) end, tables in cache,
DB_BLOCK_BUFFERS, 64-bit feature
(VLM), A-3

caution
assignment of the rights identifier, 2-16
defining Oracle symbols and logicals prior to
database installation, 1-11
if installing additional Oracle Database 10g
products in an existing Oracle home, stop all
processes running in the Oracle home, 2-25
if you remove Oracle JVM, then Oracle Universal
Installer removes all installed products that
depend on Oracle JVM, including Oracle
Database, 8-5
Oracle does not support using Oracle Universal
Installer that is shipped with earlier
releases, 3-4
read file spec guide before changing disk structure
(ODS-2 to ODS-5), 1-3
REBOOT parameter, 2-19
regarding changing disk structure, ODS-2 or
ODS-5, 1-3
removing an Oracle Database deletes all the
data, 8-2
TCP/IP protocol stacks from other vendors may
work with Oracle, but customers use these
products at their own risk, 2-7

- use the oracle10g account only for installing and maintaining Oracle software, 2-20
- CD-ROM
 - copy to hard drive, for installation of Oracle Database, 3-3
 - installing Oracle Database from CD-ROM, 3-4
 - requirements, minimum hardware, 2-2
- certification, hardware and software, 1-11
- CGI programs, H-13
- changes and enhancements, 1-2
- Character Set Scanner, 1-13
- character sets
 - AL24UTF8SS, 1-13
 - UTF8, 1-13
- CLASSPATH logical, 4-8
- clientadmin.rsp file
 - modifying a response file, B-3
- clientcustom.rsp file
 - modifying a response file, B-3
- clientruntime.rsp file
 - modifying a response file, B-3
- Cluster Configuration Information screen
 - Oracle Universal Installer, 5-8
- cluster database
 - installed configuration, 6-14
- cluster file system
 - definition, 1-13
 - requirement for Oracle Real Application Clusters, 6-2
- cluster name, 5-8
- code example
 - caching tables for performance, 64-bit feature (VLM), A-3
 - command to check if requisite patches are installed, 2-10
 - command to determine version of OpenVMS, 2-7
 - command to run Oracle Universal Installer in silent or suppressed mode, B-5
 - command to run Oracle Universal Installer noninteractively, B-4
 - creating the oraInst.loc file for noninteractive installation, B-3
 - DCL command to check disk format, 1-4
 - DCL command to set up X Window environment, 3-3, 3-4
 - for more information about other options for the runInstaller command, B-5
 - format a new disk using initialize command, 1-4
 - getting a list of command line options when you run Database Configuration Assistant, B-8
 - image dump, command to turn off this feature, 1-3
 - preparing response files for noninteractive installation, B-4
 - reformat existing disk from ODS-2 to ODS-5, 1-4
 - running Database Configuration Assistant in silent mode, B-7
 - running Oracle Network Configuration Assistant in silent mode, B-5
 - verifying correct database instance before shutdown, 7-9
- commands
 - runInstaller
 - starting Client, 3-6
 - starting database, 3-4
 - SHOW SYSTEM
 - operating system information, 2-8
 - table of commands to stop processes running in Oracle home, 8-5
- compatibility issues
 - Oracle product dependencies, 1-11
- components
 - created when using Database Configuration Assistant, 6-13
- configuration
 - general purpose configuration with Database Configuration Assistant, 1-6
- Configuration Assistant
 - database creation methods, 1-6
 - troubleshooting, G-3
- configuration types
 - Advanced, 5-10, 5-11
 - Do not create a starter database, 5-10
 - General Purpose, 5-10
 - New Database, 5-10
 - Transaction Processing, 5-10
- configuring
 - global database name, 6-6
 - SID prefix, 5-15
- connection load balancing, 6-15
- control file
 - installed configuration, 6-14
- control files, 5-6
 - described, 6-14, 7-7
 - locating, 7-7
 - reviewing, 7-6
 - using Oracle Enterprise Manager Grid Control with, 7-8
- CONTROL_FILES initialization parameter, locating control files, 7-7
- convert to cluster database
 - administrative considerations, C-2
 - from non-cluster computer, C-3
 - from single-instance, C-4
 - post-conversion, C-7
- converting
 - to Oracle Real Application Clusters from single-instance Oracle Databases, C-2
- Create Database
 - option on Creation Options screen, 6-9
- creating
 - Oracle Real Application Clusters database by using Database Configuration Assistant, 5-12, 6-4, 6-5
- cross-node registration, 6-15
- crsd, 5-9
- CSD
 - requirements, 2-10
- Custom installation, 5-14
- customized database configuration, 1-8

- custom.rsp file
 - modifying a response file, B-3
 - preparing response files, B-4

D

- data files, 5-6
 - and Database Configuration Assistant, 6-14
 - creating separate directories for, 2-25, 6-3
 - defined, 7-6
 - described, 6-14
 - minimum disk space for, 2-24, 6-2
 - options for placing on file system, 2-24, 6-2
 - recommendations for file system, 2-24
 - reviewing, 7-6
 - setting up, 7-6
- Data Guard
 - ports, ranges and protocol, E-2
- Data Warehouse
 - configuration type, 5-10, 5-11
- data warehouse database type, 5-15
- database
 - accounts
 - in databases created by Database Configuration Assistant, 7-1
 - components, created when using Database Configuration Assistant, 6-13
 - configuration options, 1-7
 - configurations, types, 5-10
 - creation methods
 - selecting, 1-6
 - Database Configuration Assistant
 - database accounts and passwords, 7-1
 - troubleshooting, G-3
 - environments
 - general purpose, 1-8
 - transaction processing, 1-8
 - files, 7-6
 - initialization parameter file, 7-5
 - redo log files, in starter database, 7-7
 - removing, 8-2
 - tablespaces, 7-6
- Database Configuration Assistant and Oracle interMedia, 1-6
 - components created by, 6-13
 - control files, 6-14
 - creating a Oracle Real Application Clusters database
 - during installation, 5-12
 - creating Oracle Real Application Clusters database
 - after installation, 6-5
 - during installation, 6-4
 - Creation Options screen, 6-9
 - data files, 6-14
 - Database Content screen, 6-7
 - database creation methods, selecting, 1-6
 - Database Credentials screen, 6-7
 - database environments configured by, 1-6
 - Database File Locations screen, 6-7
 - Database Identification screen, 6-6
 - Database Services screen, 6-7
 - Database Storage screen, 6-9
 - Database Templates screen, 6-6
 - delete database, 6-9
 - deleting databases with, 6-9
 - Initialization Parameters screen, 6-8
 - List of Cluster Databases screen, 6-10
 - Management Options screen, 6-6
 - Node Selection screen, 6-6
 - Operations screen, 6-6, 6-10
 - Recovery Configuration screen, 6-7
 - redo log files, 6-14
 - response file, B-4
 - rollback segments, 6-14
 - Storage Options screen, 6-7
 - Summary dialog box, 6-9, 6-10
 - tablespaces, 6-14
 - Welcome screen, 6-6
- Database Configuration Assistant, running Database Configuration Assistant in noninteractive or silent mode, B-6
- database configuration type
 - selecting, 5-10
- database configuration types, 5-11
- database domain, 5-15
- database name, 5-15
- DB_BLOCK_BUFFERS
 - recommendations for 64-bit feature (VLM), A-3
 - suggested parameter settings for 64-bit feature (VLM), A-2
- DB_BLOCK_SIZE. suggested parameter settings for 64-bit feature (VLM), A-2
- DB_DOMAIN name parameter, global database name, identifying databases, 7-4
- DB_NAME parameter, global database name, identifying databases, 7-4
- dbca.rsp file
 - modifying a response file, B-3
 - preparing response files, B-4
- files
 - \$ORACLE_HOME/dbs/ORA_ROOT, 7-5
- DBSNMP
 - database account, in databases created by Database Configuration Assistant, 7-1
- DCL command, to check disk format, overview, 1-4
- decision support system (DSS)
 - parameter settings, VLM or 64-bit feature, A-2
 - query-intensive transactions, 64-bit database feature, or VLM, A-2
- dedicated servers, 6-15
- default initialization parameter file, init.ora, 7-5
- deinstalling
 - Oracle Database, 5-19
- deinstalling Oracle Database, 5-19
- delete database with Database Configuration Assistant, 6-9
- devices
 - requirements
 - issues and restrictions, 2-11

- directory
 - creating separate data file directories, 2-25, 6-3
 - database file directory, 2-24
 - Oracle home directory, 2-23
 - Oracle Inventory directory, 2-23
 - oraInventory, 2-23
- directory structure
 - OpenVMS, C-8
- disk space
 - requirements, 2-2
- DISPLAY
 - running Database Configuration Assistant in noninteractive mode, B-6
 - setting
 - for running Database Configuration Assistant in noninteractive mode, B-8
- Do not create a starter database
 - configuration type, 5-10
- documentation
 - Oracle Real Application Clusters, 5-1
 - Oracle Real Application Clusters Administrator's Guide, 5-2
 - Oracle Real Application Clusters Deployment and Performance Guide, 5-2
- DSS (also see decision support system),
 - query-intensive transactions, 64-bit database feature, or VLM, A-2

E

- End of Installation screen
 - Oracle Universal Installer, 5-9
- Enterprise Edition installation, 5-14
- Enterprise Manager Agent (EMAgent), D-1
- enterprise.rsp file
 - modifying a response file, B-3
 - preparing response files, B-4
- environment
 - descriptions, database types, 1-7
 - requirements, 2-21
- environment variables
 - NLS_LANG, F-1
- error messages
 - Oracle Real Application Clusters management tools, C-2
- errors
 - configuration assistants, G-3
 - installation, G-2, G-3
 - noninteractive installation, G-4
 - silent mode, G-4
 - X Window, G-1
- EV56, minimum hp OpenVMS ALphaServer chip set required for Oracle Database 10g, 2-2
- evmd, 5-9
- examples
 - see "code example" for sample codes, 1-3

F

- failover

- and service registration, 6-15
- fatal errors, G-3
- file system
 - data file and recovery file placement options, 6-2
 - requirements for Oracle Real Application Clusters, 6-2
 - using for data files, 2-24
- files
 - Files-11 On-Disk Structure Level 5 (ODS-5)
 - format, 1-3
 - listener.ora, 4-6
 - mgw.ora, 4-8
 - Oracle Universal Installer log files, 3-7
 - oraInst.loc, B-3
 - permissions for creation, 2-19
 - setting permissions for creation, 2-16
 - supported devices, 2-11
 - tnsnames.ora, 4-6

G

- General Purpose
 - configuration type, 5-10
 - description of configuration type, 5-11
- general purpose configuration by using Database Configuration Assistant, 1-6
- general purpose database environments, 1-8
- general purpose database type, 5-15
- Generate Database Creation Scripts
 - on Creation Options screen, 6-9
- GENPARAMS parameter, 2-19
- global database name, 5-15, 6-6
- global database name, identifying databases, 7-4
- globalization support, F-1

H

- hard drive installation of Oracle Database, 3-3
- hardware
 - certification, 1-11
 - requirements, 2-1
- home directory
 - description, also see Oracle home directory, 2-23
 - properties of Oracle Database accounts, choosing a home directory, 2-20

I

- identifiers, process rights, 2-15
- image dump, recommendation - do not define dump
 - logical, 1-3
- images
 - described, 2-2
- initialization parameter file
 - description, 7-5
 - in database, 7-5
 - in databases, 7-5
 - init.ora, description, 7-5
 - initsid.ora, 7-5
- initialization parameter files
 - for instances, 6-10

- listener parameters, 6-16
- initialization parameters
 - deprecated, 1-5
 - DISPATCHERS, 6-15
 - LOCAL_LISTENER, 6-15
 - REMOTE_LISTENER, 6-15, 6-19
 - unsupported, 1-4
- Initialize command, formatting a new disk to
 - ODS-5, 1-4
- init.ora file, initialization parameter file,
 - description, 7-5
- initsid.ora, 7-5
- initsid.ora initialization parameter file, 7-5
- installation
 - overview, 3-1
 - cleaning up, 3-8
 - considerations, 1-11
 - errors, G-2, G-3
 - silent mode, G-4
 - guidelines
 - noninteractive installation and configuration, 3-2
 - Oracle Label Security, 3-2
 - Oracle Universal Installer, 3-2
 - reviewing product-specific installation guidelines, 3-2
 - ldap.ora file, 6-17
 - listener.ora file, 6-16
 - log files, G-2
 - noninteractive
 - error handling, G-4
 - oraInst.loc file, B-3
 - Oracle software owner, 2-19
 - overview, 5-5
 - postinstallation tasks, 4-1, 7-1, 8-1
 - procedure requirements, 2-12
 - requirements
 - environment, 2-21
 - response files
 - preparing, B-4
 - silent mode, G-4
 - silent mode, B-5
 - terminated, 3-8
 - tnsnames.ora file, 6-17
- instance
 - preferences, 6-5
- instance ID, 5-15
- instance, account required to start, 2-16
- instances
 - initialization parameter files, 6-10
 - SID prefix, 5-15
- issues and restrictions, port-specific, 2-11

J

- JDK
 - installation issues, 2-11
 - internationalization class, 4-8
 - run time class, 4-8

K

- Kerberos authentication, requirements, 2-3

L

- Language Selection screen
 - Oracle Universal Installer, 5-8
- ldap.ora file, 6-17
 - creating, 6-17
 - default configuration, 6-17
- licensing information, 1-6
- Lightweight Directory Access Protocol (LDAP), 6-17
- List of Cluster Databases screen, 6-10
- listener
 - identifying Oracle home for, 2-26
 - local, 6-15
 - lsnrctl command, 2-26
 - registration, 6-17
 - stop command example, 2-26
 - stopping existing listener process before new db installation, 2-25
- listener.ora file, 6-16
 - configuring, 6-15
 - default configuration, 6-16
 - modifying for external procedures, 4-7
 - static service information, 4-6
- listeners
 - listener.ora file, 6-16
- load balancing
 - and service registration, 6-15
- local device
 - using for data files, 6-2
- local device, using for data files, 2-24
- local listener, 6-15
- log files, troubleshooting
 - installation error while a configuration assistant is running, G-2
 - installation error while the Installer is copying or linking files, G-2
 - reviewing log file of an installation session, G-2
- logical volume manager (LVM), database storage
 - options, 1-9
- LRS, also see Log roll-forward server (LRS), A-1, C-1
- lsnrctl command
 - stopping the listener process, 2-26

M

- Mailbox driver, OpenVMS
 - preinstallation, 2-8
- MAXSYSGROUP, 2-13
- memory
 - advantages to reserving memory for an SGA, 2-17
 - memory-resident global sections, using the Reserved Memory Registry, 2-17
- MetaLink
 - Oracle patches, 5-16
- MGW_AGENT service name, 4-7
- mgwextproc service, adding static service

- information, 4-7
- mgw.ora file, modifying, 4-8
- MIN_GBLPAGES, calculating, 2-17
- MIN_GBLSECTIONS, calculating, 2-17
- modifying system parameters, 2-17
- most recently used MRU, tables in cache, DB_
 - BLOCK_BUFFERS, 64-bit feature (VLM), A-3
- MQSeries
 - class, 4-8
- MRU (most recently used), tables in cache, DB_
 - BLOCK_BUFFERS, 64-bit feature (VLM), A-3
- multiple databases, 2-16
- multiple Oracle homes, supported by Oracle Database
 - 10g, 1-11

N

- NAS
 - requirement for Oracle Real Application
 - Clusters, 6-2
- Net Configuration Assistant, troubleshooting, G-3
- net service names, 6-18
- network attached storage
 - See NAS
- network configuration files
 - ldap.ora.ora, 6-17
 - listener.ora, 6-16
 - sqlnet.ora, 6-21
 - tnsnames.ora, 6-17
- network directory, C-8
- New Database
 - configuration type, 5-10
- NLS_LANG environment variable, F-1
- NLS_LANG parameter
 - about, F-1
- node applications, 6-4
- None
 - TAF failover policy, 6-5
- noninteractive installation
 - oraInst.loc file, B-3
 - preparing response files, templates, B-4
 - silent mode
 - errors, G-4
 - running Oracle Universal Installer, B-5
- noninteractive installations
 - running Oracle Universal Installer, B-4
- noninteractive mode, running Database
 - Configuration Assistant in noninteractive or
 - silent mode, B-6
- Not used
 - service configuration policy, 6-5

O

- OCR
 - installed configuration, 6-13
 - specify location during install
- OCR contents, 6-13
- ocr.loc file, 5-8, 5-20
- ocssd, 5-9

- ODS-5
 - disk structure, Files-11 On-Disk Structure Level 5
 - format, 1-3
 - formatted CD-ROM disks provided by
 - Oracle, 2-2
- OLS
 - also see Oracle Label Security, 3-2
- olsnodes command, 5-9
- OLTP (Online Transaction Processing)
 - database environments, 1-8
- OLTP, Online Transaction Processing
 - transactions, read/write transactions and 64-bit
 - database feature, or VLM, A-2
- Online Transaction Processing (OLTP) environment
 - descriptions, 1-8
- OpenVMS
 - system parameters, 2-17
- OpenVMS Mailbox driver
 - preinstallation, 2-8
- operating system
 - configuration, 2-19
- options
 - database configuration options, 1-7
- Oracle
 - account
 - home directory, 2-20
 - postinstallation tasks, 4-5
 - preinstallation tasks, 2-21
 - buffer cache, cache hit ratio, suggested parameter
 - settings for 64-bit feature (VLM), A-2
 - home directory
 - description, 2-23
 - recommended path, 2-23
 - requirement for, 2-23
 - requirements, 2-23
 - home name, 2-23
 - internationalization class, 4-8
 - JDBC class, 4-8
 - Label Security
 - installation guidelines, 3-2
 - postinstallation tasks, 4-6
 - Oracle Client is installed separately from the
 - database, 1-5
 - Oracle Cluster Ready Services daemon, 1-12
 - Oracle Spatial
 - Motif requirement, 2-6
 - requirement for example programs, 2-6
 - oracle user
 - configuring environment for, 2-26
 - Oracle Wallet Manager and Secure Socket
 - Layer, 2-3
 - parallel server, re-installing, 3-8
 - precompilers, postinstallation tasks, 4-8
 - Server DBA account, 2-12
 - Server, automating startup and shutdown, 4-5
 - software
 - owner user, configuring user's
 - environment, 2-26
 - owner user, preinstallation setup tasks to
 - perform, 2-21

- removing, 8-4
- tools, adding or upgrading, 7-8
- Oracle Advanced Security
 - add-on product for Oracle Net Server or Oracle Net Client, 2-3
 - Oracle Wallet Manager, 2-3
- Oracle Cluster Ready Services
 - install with Oracle Universal Installer, 5-7
 - installation setup procedures, 5-7
- Oracle Cluster Ready Services home
 - cannot be same as home for database software with Oracle Real Application Clusters, 5-7
- Oracle Cluster Ready Services software
 - removing, 5-20
- Oracle Cluster Registry
 - See OCR
- Oracle Clusterware
 - ports, E-2
 - ports, ranges and protocol, E-2
- Oracle Database
 - 64-bit feature cannot be removed, A-1
 - account quotas required, 2-14
 - Client
 - chip set requirements, 2-2
 - disk space requirements, 2-3
 - creating account, 2-13
 - creating data file directories, 2-25, 6-3
 - custom installation, 1-5
 - Enterprise Edition installation, 1-5
 - minimum disk space requirements, 2-24, 6-2
 - relinking Oracle code, 7-9
 - removing, 5-19
 - Server
 - disk space requirements, 2-3
 - setting system parameters, 2-18
- Oracle Enterprise Management Agent
 - HTTP port, changing, E-2
 - ports
 - ranges and protocol, E-2
- Oracle Enterprise Manager
 - postinstallation configuration, 5-17
- Oracle Enterprise Manager, Grid Control
 - using to modify control files, 7-8
 - using to modify redo log files, 7-7
 - using to view control files, 7-8
 - using to view redo log files, 7-7
- Oracle Inventory
 - description, 2-23
 - directory description, 2-23
- Oracle inventory
 - log files, 3-7
- Oracle Label Security
 - installation guidelines, 3-2
- Oracle Messaging Gateway
 - class, 4-8
- Oracle Network Configuration Assistant, 5-16
- Oracle Network Configuration Assistant, running
 - Oracle Network Configuration Assistant in silent mode, B-5
- Oracle Network Manager
 - and Oracle Advanced Security, 2-3
 - Configuration Assistant
 - response file, B-4
 - configuration file directory, 4-6
 - listener, running Oracle Network Configuration Assistant in silent mode, B-5
 - lsnrctl command, 2-26
 - Services, postinstallation tasks, 4-5
 - stopping a listener process, 2-26
 - stopping existing listener, 2-25
 - stopping the listener, 2-26
- Oracle Real Application Clusters
 - components, 5-6
 - documentation, 5-1
 - file system requirements, 6-2
 - installed components of, 5-6
 - management tools error messages, C-2
 - other considerations, C-1
- Oracle Real Application Clusters high availability extensions
 - configuring services for, 6-5
 - TAF policies for, 6-5
- Oracle recovery files
 - guidelines, 2-24
- Oracle SQL*Net Listener
 - ports, ranges and protocol, E-2
- Oracle Technology Network Web site
 - downloading installation software from, 3-2
- Oracle Universal Installer, 3-2
 - Available Product Components screen, 5-14
 - Cluster Configuration Information screen, 5-8
 - creating an Oracle Database during the installation, 1-7
 - database creation methods, 1-6
 - End of Installation screen, 5-9
 - installation from CD-ROM, 3-4
 - installation from hard disk, 3-4
 - installation guidelines, 3-2
 - inventory
 - oraInventory directory, 3-7
 - Language Selection screen, 5-8
 - list of response file names, B-3
 - log files, 3-7
 - overview of processing, 5-5
 - Privileged Operating System Groups screen, 5-15
 - Select Database Configuration screen, 5-14
 - Select Installation Type screen, 5-14
 - Selected Nodes screen, 5-13
 - Specify Backup and Recovery Options screen, 5-15
 - Specify Database Configuration Options screen, 5-15
 - Specify Database File Storage Option screen, 5-15
 - Specify Database Schema Passwords screen, 5-15
 - Specify File Locations screen, 5-7, 5-13
 - Specify Hardware Cluster Installation Mode screen, 5-14
 - Specify Network Interface Usage screen, 5-8
 - Summary screen, 5-9, 5-15
 - Voting Disk Information screen, 5-8

- Welcome screen, 5-13, 5-20
- ORACLE_HOME logical
 - oracle account home directory, 2-20
 - requirements, 2-11
- ORACLE_HOME logical, setting for running
 - Database Configuration Assistant in noninteractive mode, B-8
- OracleMetaLink, 5-16
- oraInventory directory, 3-7
 - See Oracle Inventory directory
- oratab file, 6-13

P

- paging
 - determine amount of paging configured on the system, 2-2
- parameter
 - DB_DOMAIN name, global database name, identifying databases, 7-4
 - DB_NAME, global database name, identifying databases, 7-4
 - GENPARAMS, 2-19
 - SAVPARAMS, 2-18
 - SERVICE_NAMES, global database name, identifying databases, 7-5
 - SYSGEN, 2-17
- parameter file search order, 6-11
- passwords
 - changing, 4-4
 - management, unlocking database accounts, 7-3
 - resetting, 7-3
 - reviewing, 7-1
 - revoked after installation, for all Oracle system administration accounts, 7-3
 - unlocking, 7-3
- patches
 - download, 5-16
 - install, 5-16
 - OracleMetaLink, 5-16
 - requirements, 2-8
- performance, benefit of 64-bit database feature, or VLM, A-1
- permissions
 - file creation, 2-19
- port numbers
 - managing, E-1
- portlist.ini file, E-1
- ports
 - access URLs, E-1
 - configured for applications, E-1
 - Data Guard, ranges and protocol, E-2
 - default ranges, E-1
 - Oracle Clusterware, E-2
 - Oracle Clusterware, ranges and protocol, E-2
 - Oracle Enterprise Management Agent HTTP, changing, E-2
 - Oracle Enterprise Management Agent, ranges and protocol, E-2
 - Oracle SQL*Net Listener, ranges and

- protocol, E-2
- post-conversion recommendations, C-7
- postinstallation
 - Oracle Enterprise Manager configuration, 5-17
 - patch download and install, 5-16
 - product configuration, 5-17
 - required tasks
 - main topic, 4-3
 - Oracle Label Security, configuring, 4-6
 - Oracle Net Services, configuring, 4-5
 - Oracle Precompilers, 4-8
 - patches, installing and downloading, 4-3
 - user accounts setup, 5-18
- precompilers
 - Pro*C/C++, 4-8
 - Pro*COBOL, 4-8
 - Pro*FORTRAN, 4-9
- preconfigured database installation types, 5-11
- Pre-connect
 - TAF failover policy, 6-5
- Preferred
 - service configuration policy, 6-5
- preinstallation tasks
 - as the System user, 2-20
 - Oracle Network Manager setup tasks, 2-22
- Privileged Operating System Groups screen
 - Oracle Universal Installer, 5-15
- Pro*C/C++ precompiler, 4-8
- Pro*COBOL precompiler, 4-8
- Pro*FORTRAN
 - requirements, 2-7
- Pro*FORTRAN precompiler, 4-9
- process
 - stopping existing, 2-25
 - stopping existing listener process, 2-25
- process rights identifier, 2-15
- Program Global Area (PGA)
 - sorts and SORT_AREA_SIZE parameter, A-2
 - suggested parameter settings for 64-bit feature (VLM), A-2
- ps command, 2-26

Q

- query-intensive transactions (DSS), 64-bit database feature, or VLM, A-2
- quotas, account, 2-14

R

- RAID, using for Oracle data files, 2-24
- RAM
 - determining amount of RAM installed on the system, 2-2
- RBS tablespace
 - description, 6-14
- readme.txt file, E-1
- read/write transactions (OLTP), 64-bit database feature, or VLM, A-2

- recovery files
 - options for placing on file system, 6-2
- recovery files, options for placing on file system, 2-24
- redo log file
 - installed configuration, 6-14
- redo log files, 5-6
 - described, 6-14
 - in starter database, 7-7
 - locating, in starter database, 7-7
 - reviewing, 7-6
 - using Oracle Enterprise Manager Grid Control with, 7-7
- redundant array of independent disks
 - also see RAID, 2-24
- registration
 - cross-node, 6-15
- relative path to response file causes installation failure, B-5
- relinking
 - Oracle code, 7-9
 - Oracle products, 7-9
- removing Oracle Cluster Ready Services software, 5-20
- removing Oracle Database, 5-19
- requirements
 - CD-ROM, 2-2
 - devices, 2-11
 - disk space, 2-2
 - file permissions, 2-19
 - hardware, 2-1
 - JDK
 - used by Oracle Java applications, 2-11
 - Kerberos authentication, 2-3
 - Oracle Advanced Security
 - an add-on product, 2-3
 - ORACLE_HOME, 2-11
 - Secure Socket Layer
 - requires a compatible wallet, 2-3
 - system configuration, 2-19
- Reserved Memory Registry, 2-17
- response file
 - how to install and configure Oracle products noninteractively, main topic, B-1
 - names, B-3
 - relative path to response file causes installation failure, B-5
 - table of files provided with Oracle Database for noninteractive installation, B-4
 - template, running Database Configuration Assistant in noninteractive or silent mode, B-6
- RMAN
 - enabling automated backups with Oracle Recovery Manager, 1-10
- rollback segments
 - described, 6-14
- run time, disk space requirements, 2-3
- runInstaller command
 - options, B-5

S

- SAME (stripe and mirror everything) methodology, file system recommendations, 1-9
- Save as a Database Template
 - option on Creation options screen, 6-9
- SAVPARAMS parameter, 2-18
- scalability, advantage of 64-bit database feature, or VLM, A-2
- security issues, 2-16
- seed database, Database Configuration Assistant, 1-6
- Select Database Configuration screen
 - Oracle Universal Installer, 5-14
- Select Installation Type screen
 - Oracle Universal Installer, 5-14
- Selected Nodes screen
 - Oracle Universal Installer, 5-13
- server parameter files, 5-6, 6-11
 - errors, 6-12
- Service Management
 - using, 6-5
- service registration
 - configuring, 6-15
- SERVICE_NAMES parameter, global database name, identifying databases, 7-5
- services, 6-5
- Set Volume command, changing a disk to ODS-5, 1-4
- setting account privileges, 2-14
- setting account quotas, 2-14
- SGA protection, 2-16
- shared configuration file, 6-13
- shared images
 - overview, 2-2
- shared server, 6-15
- SHOW SYSTEM command
 - operating system information, 2-8
- shutdown and startup, 4-5
- SID, 5-15
- sid Prefix, 5-15
- silent mode
 - installation, B-5
 - running Database Configuration Assistant in silent mode or noninteractive mode, B-6
 - running Oracle Network Configuration Assistant in silent mode, B-5
- sizer command, 2-7
- software certification, 1-11
- Software Only
 - configuration type, 5-11
 - software only, installation choice, 1-8
- SORT_AREA_SIZE parameter. suggested parameter settings for 64-bit feature (VLM), A-2
- Specif File Locations screen
 - Oracle Universal Installer, 5-7
- Specify Backup and Recovery Options screen
 - Oracle Universal Installer, 5-15
- Specify Database Configuration Options screen
 - Oracle Universal Installer, 5-15
- Specify Database File Storage Option screen

- Oracle Universal Installer, 5-15
- Specify Database Schema Passwords screen
 - Oracle Universal Installer, 5-15
- Specify File Locations screen
 - Specify File Locations screen, 5-13
- Specify Hardware Cluster Installation Mode screen
 - Oracle Universal Installer, 5-14
- Specify Network Interface Usage screen
 - Oracle Universal Installer, 5-8
- spfile
 - default creation, 6-11
 - default location, 6-11
 - upgrade to, 6-11
- SQLJ class, 4-8
- sqlnet.ora file, 6-21
 - default configuration, 6-22
- starting instances, 2-16
- startup and shutdown, 4-5
- static service information, adding for mgwextproc service, 4-7
- stripe and mirror everything (SAME) methodology
 - file system recommendations, 1-9
- Summary dialog box, 6-9, 6-10
- Summary screen
 - Oracle Universal Installer, 5-9, 5-15
- suppressed mode, running Oracle Universal Installer
 - in silent or suppressed mode, B-5
- SYS account, 2-13
- SYS database account, in databases created by Database Configuration Assistant, 7-1
- SYSGEN parameters, 2-17
- SYSMAN
 - database account, in databases created by Database Configuration Assistant, 7-1
- utility
 - reserving memory for an SGA, 2-17
 - using the Reserved Memory Registry, 2-17
- system
 - parameters, 2-17
 - summary of requirements, 2-19
- SYSTEM account, 2-13
- SYSTEM database account, in databases created by Database Configuration Assistant, 7-1
- System Global Area (SGA)
 - no need to estimate maximum SGA, a 64-bit database feature, or VLM, A-2
- System Identifier (SID) identifies a specific database instance, 7-5
- SYSTEM tablespace
 - description, 6-14
- System user, preinstallation tasks, 2-20

T

- tablespaces
 - defined, 7-6
 - in databases, 7-6
 - reviewing, 7-6
 - setting up, 7-6
 - and Database Configuration Assistant, 6-14

- expanding for large sorts, 6-14
- RBS, 6-14
- SYSTEM, 6-14
- TEMP, 6-14
- undo tablespaces for automatic undo management, 6-14
- USERS, 6-14
- TEMP tablespace
 - described, 6-14
- template, response file template, running Database Configuration Assistant in noninteractive or silent mode, B-6
- tnsnames.ora file, 6-17
 - adding a connect descriptor, 4-7
 - configuring Oracle Network Manager services, 4-6
 - default configuration, 6-17
 - MGW_AGENT service name, 4-7
 - modifying for external procedures, 4-7
- tools
 - adding or upgrading, 7-8
- Transaction Processing
 - configuration type, 5-10
 - description of configuration type, 5-11
- transaction processing database type, 5-15
- transaction processing, database environments, 1-8
- Transparent Application Failover (TAF) policies, 6-5
- troubleshooting
 - main topic, G-1
 - fatal errors, G-3
 - oraInventory log files, 3-7

U

- UAF, 2-12
- UIC GROUP number, 2-13
- undo management, 6-14
- uninstalling
 - Oracle Database, 5-19
- uninstalling Oracle Database, 5-19
- upgraded databases, configuring, 4-4
- upgrading
 - Oracle products, 7-8
 - to Oracle Database 10g from previous Oracle version, 1-11
 - upgrade issues, 1-12
- user accounts
 - postinstallation setup, 5-18
- user and group numbers, 2-12
- user equivalence
 - testing, 5-13
- USERS tablespace
 - described, 6-14
- UTF8 character set, 1-13
- UTLRP.SQL
 - recompiling invalid SQL modules, 4-4

V

- Very Large Memory (VLM)

- introduction, A-1
- reserving memory, 2-17
- voting disk
 - specify location during install, 5-8
- Voting Disk Information screen
 - Oracle Universal Installer, 5-8

W

- WebSphere MQ
 - checking requirements, 2-8
 - class, 4-8
 - CSDs required, 2-10
- Welcome screen
 - Oracle Universal Installer, 5-13, 5-20

X

- X Window
 - and X/Motif Software development, Motif requirement, 2-6
 - display errors, G-1
