Oracle® Database

Installation Guide

12*c* Release 1 (12.1) for Linux

E41491-20

July 2017

ORACLE®

Oracle Database Installation Guide, 12*c* Release 1 (12.1) for Linux

E41491-20

Contents

5   Configuring Users, Groups and Environments for Oracle Database

6   Oracle Grid Infrastructure for a Standalone Server

7 Installing Oracle Database

8 Oracle Database Postinstallation Tasks

10   Removing Oracle Database Software

A   Installing and Configuring Oracle Database Using Response Files

B   Cloning an Oracle Home

C   Using NAS Devices

D   How to Complete Preinstallation Tasks Manually

E   Configuring Networks for Oracle Database

F   Optimal Flexible Architecture

G   Managing Oracle Database Port Numbers

Preface

This guide explains how to install and configure Oracle Database for Linux. This guide also provides information about Optimal Flexible Architecture, cloning an Oracle home, troubleshooting the installation, and how to remove the database software.

The preface contains the following topics:

- Audience
- Documentation Accessibility
- Related Documentation
- Conventions

Audience

This guide is intended for anyone responsible for installing Oracle Database 12*c* Release 1 (12.1) on Linux systems. Additional installation guides for Oracle Database, Oracle Real Application Clusters, Oracle Clusterware, Oracle Database Examples, and Oracle Enterprise Manager Cloud Control are available on Oracle Technology Network at:

http://docs.oracle.com

> **See Also:** *Oracle Database Quick Installation Guide for Linux x86-64* to install Oracle Database using the default settings

> **See Also:**
>
> To install Oracle Database using the default settings refer to:
>
> - *Oracle Database Quick Installation Guide for Linux x86-64*
> - *Oracle Database Quick Installation Guide for IBM: Linux on System z*

Documentation Accessibility

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

Access to Oracle Support

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

http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documentation

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

- *Oracle Database Concepts*

- *Oracle Database New Features Guide*

- *Oracle Database Licensing Information*

- *Oracle Database Readme*

- *Oracle Universal Installer User's Guide*

- *Oracle Grid Infrastructure Installation Guide*

- *Oracle Database Release Notes for Linux*

- *Oracle Database Client Installation Guide for Linux*

- *Oracle Database Examples Installation Guide*

- *Oracle Real Application Clusters Installation Guide for Linux and UNIX*

- *Oracle Database Quick Installation Guide for Linux x86-64*

- *Oracle Database Client Quick Installation Guide for Linux x86-64*

- *Oracle Database Quick Installation Guide for IBM: Linux on System z*

- *Oracle Database Client Quick Installation Guide for IBM: Linux on System z*

- *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems*

- *Oracle Automatic Storage Management Administrator's Guide*

- *Oracle Database Upgrade Guide*

- *Oracle Database 2 Day DBA*

- *Oracle Application Express Installation Guide*

For information about Oracle error messages, see *Oracle Database Error Messages Reference*. Oracle error message documentation is available only in HTML. If you have access to the Oracle Database 12*c* Release 1 (12.1) Online Documentation Library, then you can browse the error messages by range. After you find the specific range, use your browser's "find in page" feature to locate the specific message. When connected to the Internet, you can search for a specific error message using the error message search feature of the Oracle online documentation.

Many books in the documentation set use the sample schemas of the seed database, which is installed by default when you install Oracle Database. See *Oracle Database Sample Schemas* for information about how these schemas were created and how you can use them yourself.

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

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

You can go directly to the documentation section at:

http://docs.oracle.com

See *Oracle Database Release Notes for Linux* for important information that was not available when this book was released. The release notes for Oracle Database is updated regularly. The most recent version is available at:

http://docs.oracle.com

Conventions

The following text conventions are used in this document:

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

Changes in This Release for Oracle Database Installation Guide

This preface contains:

- Changes in Oracle Database 12c Release 1 (12.1)

> **See Also:** *Oracle Database New Features Guide*

Changes in Oracle Database 12c Release 1 (12.1)

The following are changes in *Oracle Database Installation Guide* for Oracle Database 12*c* Release 1 (12.1):

- New Features
- Deprecated Features
- Desupported Features
- Other Changes

New Features

- Oracle ASM Filter Driver

  The Oracle ASM filter driver (Oracle ASMFD) feature is available starting with Oracle Database 12*c* Release 1 (12.1.0.2).

  Oracle ASMFD is a kernel module that resides in the I/O path of the Oracle ASM disks.

  Oracle ASMFD rejects any write requests that are not issues by Oracle software. This action eliminates accidental overwrites of Oracle ASM disks that would cause corruption in an Oracle ASM disk and files within the disk group.

  The filter has additional functionality to fence I/Os from entities that are no longer communicating with Oracle ASM.

  See "About Oracle ASM with Oracle ASM Filter Driver" on page 6-11.

  See "Using Oracle ASM Filter Driver" in *Oracle Automatic Storage Management Administrator's Guide*

  > **Note:** Oracle ASM Filter Driver is not supported on IBM: Linux on System z.

- Root Scripts Automation

  Starting with Oracle Database 12c Release 1 (12.1), Oracle Universal Installer provides options to automatically run root configuration scripts required during a grid infrastructure installation. You also have the option to manually run the root configuration scripts.

  See "Determining Root Script Execution Plan" on page 5-16

- Oracle Flex ASM

  Oracle Flex ASM enables an Oracle ASM instance to run on a separate physical server from the database servers. Many Oracle ASM instances can be clustered to support a large number of database clients.

  Oracle Database instances can be set up as clients to Oracle Flex ASM where metadata is provided to the database instance by an Oracle Flex ASM instance that may be on a different node than the database instance.

  Note that Oracle Flex ASM can apply to a collection of databases, each one a single instance but running in a Flex ASM Cluster.

  See *Oracle Automatic Storage Management Administrator's Guide* for more information about Oracle Flex ASM

- Deinstallation Tool Integrated with Installation Media

  Starting with Oracle Database 12*c*, the deinstallation tool is integrated with the database installation media and is no longer provided on a separate installation media.

  See "About the Deinstallation Tool" on page 10-1 and "Change for Standalone Deinstallation Tool" in *Oracle Database Upgrade Guide.*

- Simplified Oracle Label Security Installation

  Starting with Oracle Database 12*c*, Oracle Label Security is installed by default as part of the Oracle Database installation. You can no longer select Oracle Label Security as a custom component during an Enterprise Edition database installation.

  See "Configuring Oracle Label Security" on page 8-7 and *Oracle Label Security Administrator's Guide*.

- Simplified Oracle Database Vault Installation

  Starting with Oracle Database 12*c*, Oracle Database Vault is installed by default as part of the Oracle Database installation. However, you can configure, enable, or disable Oracle Database Vault after the Oracle Database installation, either using Oracle DBCA, or by running SQL statements as described in the "Disabling and Enabling Oracle Database Vault" section in *Oracle Database Vault Administrator's Guide*.

  See "Registering Oracle Database Vault with an Oracle Database" in *Oracle Database Vault Administrator's Guide*

- Unified Database Audit Configuration

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

This new audit policy framework is included with the database installation.

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

- Oracle Enterprise Manager Database Express 12c

Oracle Database 12*c* introduces Oracle Enterprise Manager Database Express 12*c*, a web-based management tool built into Oracle Database without any need for special installation or management. Using Oracle Enterprise Manager Database Express, you can perform basic administrative tasks such as user, performance, memory, and space management. You can also view performance and status information about your database.

See "Introduction to Oracle Enterprise Manager Database Express" in *Oracle Database 2 Day DBA* and "Getting Started with Oracle Database" on page 9-1

Note that starting with Oracle Database 12*c*, Oracle Enterprise Manager Database Control is no longer available. See *Oracle Database Upgrade Guide* for more information.

- Multiprocess and Multithreaded Oracle Database

Starting with Oracle Database 12*c*, Oracle Database may use operating system threads to allow resource sharing and reduce resource consumption.

See "Introduction to Processes" in *Oracle Database Concepts*.

- Support for Separation of Database Administration Duties

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

See "Creating Required Operating System Groups and Users" on page 5-1 and "Managing Administrative Privileges" in *Oracle Database Security Guide*.

- Oracle DBCA Support for CDBs and PDBs

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

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

See the "Specify Database Identifiers" screen in "Running Oracle Universal Installer" on page 7-9, and *Oracle Database Administrator's Guide*.

See "About Common Users and Local Users", "Managing Common Roles and Local Roles" and "Managing Commonly and Locally Granted Privileges" in *Oracle Database Security Guide* for information on users, roles, and privileges in a CDB and PDB.

- Support for NFS Version in Direct NFS Client

Starting with Oracle Database 12*c*, you can specify the NFS protocol version to be used by Direct NFS Client.

See "Enabling Direct NFS Client" on page 8-12.

Deprecated Features

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

- Change for Standalone Deinstallation Tool

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

  See "Deinstallation Tool Integrated with Installation Media"

- Deprecation of `-cleanupOBase`

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

Desupported Features

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

- Oracle Enterprise Manager Database Control
- `CLEANUP_ORACLE_BASE` Property Removed

Other Changes

The following are additional changes in the release:

- Document Structure Changes

  Some preinstallation tasks, and Oracle Preinstallation RPM information is redesigned into category topics. For more details, continue to read this section in its entirety, and refer to the chapters that subdivide preinstallation tasks into category topics.

- Oracle Preinstallation RPM

  Information on configuring Oracle Linux with Oracle Preinstallation RPMs is now available in a separate chapter:

  Chapter 3, "Automatically Configuring Oracle Linux with Oracle Preinstallation RPM"

  Similarly, for information about installing a supported Linux distribution, see "Guidelines for Linux Operating System Installation" on page 4-2.

- Preinstallation Task Changes

  - Fixup scripts do not replace system tuning, but they do reduce the amount of manual system configuration required for an initial deployment. For this reason, some manual tasks that fixup scripts perform are now moved to an appendix. If you choose to, see Appendix D, "How to Complete Preinstallation Tasks Manually" to continue to configure your servers manually.

  - Oracle Database, and Oracle Grid Infrastructure for a standalone server hardware and system requirements are now subdivided into category topics:

    "Configuring Servers for Oracle Database" on page 4-7

    "Operating System Requirements for x86-64 Linux Platforms" on page 4-11

    "Configuring Servers for Oracle Grid Infrastructure for a Standalone Server" on page 6-2

- Configuring users, groups, and environments for Oracle Database are now moved to a separate chapter:

  Chapter 5, "Configuring Users, Groups and Environments for Oracle Database"

- Configuring networks for Oracle Database is now moved to a separate appendix:

  Appendix E, "Configuring Networks for Oracle Database"

1

Oracle Database Installation Checklist

The following checklists provide a list of required preinstallation steps:

- Hardware Checklist for Oracle Database Installation
- Operating System Checklist for Oracle Database Installation on Linux
- Oracle User Environment Configuration Checklist for Oracle Database Installation
- Server Environment Configuration Checklist for Oracle Database Installation
- Storage and Recovery Checklist for Oracle Database Installation
- OUI Checklist for Single Instance Oracle Database Installation
- Planning Checklist for Oracle Database Installation

Use these checklists to coordinate tasks to help ensure that all system and storage preparation, and configuration tasks are completed before starting an Oracle Database or an Oracle Grid Infrastructure for a standalone server installation.

Hardware Checklist for Oracle Database Installation

Review the following hardware checklist for all installations:

Table 1–1   Server Hardware Checklist for Oracle Database

Check	Task
Confirm Server Make and Architecture	Confirm that server make, model, core architecture, and host bus adaptors (HBA) are supported to run with Oracle Database and Oracle Grid Infrastructure for a standalone server.
Check Runlevel, Display and Network Setup	■ Server should be started in runlevel 3 or runlevel 5. ■ Server display cards provide at least 1024 x 768 display resolution. ■ Server is connected to the network, contains a display monitor and DVD drive.
Allocate Local Disk Space for Oracle Software	■ Based on the Oracle Database edition you intend to install, server meets the disk space requirements listed in Table 4–1. ■ At least 6.9 GB of disk space for an Oracle Grid Infrastructure for a standalone server installation.
Random Access Memory (RAM)	■ At least 1 GB RAM for Oracle Database installations. See Table 4–3. ■ At least 4 GB of RAM for Oracle Grid Infrastructure for a standalone server including installations where you plan to install Oracle Database. See Table 6–1.
Temporary Disk Space Allocation	At least 1 GB allocated to /tmp.
Check Storage Hardware	Either Storage Area Network (SAN) or Network-Attached Storage (NAS). See Appendix C, "Using NAS Devices".

Operating System Checklist for Oracle Database Installation on Linux

Review the following software checklist for all installations:

Table 1–2   Operating System Checklist for Oracle Database on Linux

Check	Task
Operating System	■ Supported in the list of supported distribution, releases, kernels, and packages listed in "Operating System Requirements for x86-64 Linux Platforms" on page 4-11. ■ OpenSSH installed manually, if you do not have it installed already as part of a default Linux installation.
Oracle Preinstallation RPM for Oracle Linux	On Oracle Linux systems, see Chapter 3, "Automatically Configuring Oracle Linux with Oracle Preinstallation RPM" to configure your operating system for Oracle Database and Oracle Grid Infrastructure installations.
Kernel Parameters	Check "Configuring Kernel Parameters for Linux" on page D-1 to review the minimum kernel parameter requirements for installation. If your parameter settings do not meet the minimum requirements for installation, then Oracle Universal Installer (OUI) generates a fixup script that can change these settings to the minimum required levels for installation.

Oracle User Environment Configuration Checklist for Oracle Database Installation

Review the following user environment checklist for all installations:

Table 1–3   User Environment Configuration for Oracle Database

Check	Task
Create Groups and Users	Review "Creating Required Operating System Groups and Users" on page 5-1 for information about the groups and users you need to create for the kind of deployment you want to do. Installation owners have resource limits settings and other requirements described in "Checking Resource Limits for Oracle Software Installation Users" on page 5-9. Group and user names must use only ASCII characters.
Review Oracle Inventory (oraInventory) and OINSTALL Group Requirements	The Oracle Inventory directory is the central inventory of Oracle software installed on your system. Users who have the Oracle Inventory group as their primary group are granted the OINSTALL privilege to read and write to the central inventory. ■ **If you have an existing installation**, then OUI detects the existing oraInventory directory from the `oraInst.loc` file, and uses this location. ■ **If you are installing Oracle software for the first time**, and your system does not have an oraInventory directory, then the installer creates an Oracle inventory that is one directory level up from the Oracle base for the Oracle Grid Infrastructure install, and designates the installation owner's primary group as the Oracle Inventory group. Ensure that this group is available as a primary group for all planned Oracle software installation owners.
Unset Oracle Software Environment Variables	If you have had an existing installation on your system, and you are using the same user account to install this installation, then unset the `ORACLE_HOME`, `ORACLE_BASE`, `ORACLE_SID`, `TNS_ADMIN` environment variables and any other environment variable set for the Oracle installation user that is connected with Oracle software homes.
Configure the Oracle Software Owner Environment	Configure the environment of the `oracle` or `grid` user by performing the following tasks: ■ Set the default file mode creation mask (`umask`) to 022 in the shell startup file. ■ Set the `DISPLAY` environment variable. See, "Configuring Oracle Software Owner Environment" on page 5-13.

Server Environment Configuration Checklist for Oracle Database Installation

Review the following environment checklist for all installations:

Table 1–4   Environment Configuration for Oracle Database

Check	Task
Create Mount Point Paths for the Software Binaries	Oracle recommends that you follow the guidelines for an Optimal Flexible Architecture configuration, as described in Appendix F, "Optimal Flexible Architecture".
Check Shared Memory File System Mount	Review "Checking Shared Memory File System Mount on Linux" on page 4-32 to ensure that the /dev/shm mount area is of type tmpfs and is mounted with the appropriate options.
Oracle Users and Home Paths Must Use ASCII Characters Only	Ensure Oracle home paths and other Oracle directory names use ASCII characters only. Also ensure Oracle installation owner user names are ASCII characters only, because installation owner names are used for some default Oracle softwarse paths.
Enable Remote Display Configuration	If you are installing the software from an X Window System workstation or X terminal, then log in as root and enable remote display. See, "Logging In to the System as root" on page 4-6.

Storage and Recovery Checklist for Oracle Database Installation

Review the following storage configuration task checklist for all installations:

Table 1–5   Storage Configuration Checks for Oracle Database

Check	Task
Decide the Database Storage Option	During the database installation, you must specify one of the following storage options for database files: ■ **File System** Oracle Database Configuration Assistant (Oracle DBCA) creates the database files in a directory on a file system mounted on the computer. Oracle recommends that the file system be separate from the file systems used by the operating system or the Oracle software. The file system can be any of the following: - A file system on a disk that is physically attached to the system - A file system on a logical volume manager (LVM) volume or a RAID device - A network file system (NFS) mounted from a certified network-attached storage (NAS) device. ■ **Oracle Automatic Storage Management** Oracle Automatic Storage Management (Oracle ASM) is installed as part of an Oracle Grid Infrastructure installation. If you plan to use Oracle ASM, then you must install Oracle Grid Infrastructure before you install and create the database. See "Database Storage Options" on page 2-9.
Decide If You Want to Enable Recovery	If you decide to enable recovery during the database installation, then select one of the following options: ■ **File System:** Oracle Universal Installer (OUI) provides you with an option to configure the fast recovery area location. ■ **Oracle Automatic Storage Management:** This option allows you only to use the same disk group for both Oracle Database files and recovery files.. See "Identifying Storage Requirements for Oracle Automatic Storage Management" and "Creating a Fast Recovery Area Disk Group".

OUI Checklist for Single Instance Oracle Database Installation

Review the following task checklist before you start Oracle Universal Installer (OUI) for single instance Oracle Database installations:

Table 1–6   Oracle Database Checks Before Starting OUI

Check	Task
Verify if Oracle Grid Infrastructure Exists	If you want to use Oracle ASM or Oracle Restart, then you must install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart.
Check Running Oracle Processes, and Shut Down if Necessary	■ **On a standalone database not using Oracle ASM**: You do not need to shut down the database while you install Oracle Grid Infrastructure. ■ **On a standalone database using Oracle ASM**: Stop the existing Oracle ASM instances. The Oracle ASM instances are restarted during installation. See "Stopping Existing Oracle Processes" on page 5-12.
Ensure cron Jobs Do Not Run During Installation	If the installer is running when daily cron jobs start, then you may encounter unexplained installation problems if your cron job is performing cleanup, and temporary files are deleted before the installation is finished. Oracle recommends that you complete installation before daily cron jobs are run, or disable daily cron jobs that perform cleanup until after the installation is completed.
Decide if You Want to Install Other Languages	During installation, you are asked if you want translation of user interface text into languages other than the default, which is English. If the language set for the operating system is not supported by the installer, then by default the installer runs in the English language. See *Oracle Database Globalization Support Guide* for detailed information about character sets and language configuration.
Decide Your Management Option	By default, Oracle Database 12*c* is managed by Oracle Enterprise Manager Database Express. However, if you have an existing Oracle Management Agent, and decide to use Oracle Enterprise Manager Cloud Control to centrally manage your database, then obtain the following information to enter during the database installation: ■ **OMS host** ■ **OMS port** ■ **EM admin username** ■ **EM admin password** **See Also:** ■ *Oracle Database 2 Day DBA* for configuring Oracle Enterprise Manager Database Express. ■ *Oracle Enterprise Manager Cloud Control Administrator's Guide* for information about how to use Oracle Enterprise Manager Cloud Control to discover targets
Determine Root Script Execution Plan	See, "Determining Root Script Execution Plan" on page 5-16 to determine how you plan to run root scripts during an Oracle Grid Infrastructure installation.
Consider Memory Allocation and Automatic Memory Management	You can enable automatic memory management either during, or after the database installation. Enabling automatic memory management after installation involves a shutdown and restart of the database. With automatic memory management, the Oracle Database instances automatically manage and tune memory for you. With automatic memory management, you choose a memory target, and the instance automatically distributes memory between the system global area (SGA) and the instance program global area (instance PGA). As memory requirements change, the instance dynamically redistributes memory between the SGA and instance PGA. **See Also:** *Oracle Database Administrator's Guide*
Disable Transparent HugePages	Oracle recommends that you disable Transparent HugePages and use standard HugePages for enhanced performance. See, "Disabling Transparent HugePages" on page 4-33.

Planning Checklist for Oracle Database Installation

Review the following general tasks checklist for all installations:

Table 1–7   Recommended Planning Checks for Oracle Database

Check	Task
Read the Release Notes	Read *Oracle Database Release Notes for Linux* before you begin the installation. The release notes are available with the platform-specific documentation.
Review the Licensing Information	Although the installation media in the media pack contain many Oracle components, you are permitted to use only those components for which you have purchased licenses.
	Oracle Support Services does not provide support for components for which licenses have not been purchased.
	Read *Oracle Database Licensing Information*.
Obtain Your My Oracle Support Information	You will need your My Oracle Support username and password to perform various installation related tasks such as configuring security updates, downloading software updates, and reviewing certification.
	The My Oracle Support website is available at:
	https://support.oracle.com/
Review Oracle Support Certification Matrix	New platforms and operating system software versions might be certified after this guide is published, review the certification matrix on the My Oracle Support website for the most up-to-date list of certified hardware platforms and operating system versions:
	https://support.oracle.com/
	You must register online before using My Oracle Support. After logging in, from the menu options, select the **Certifications** tab. On the **Certifications** page, use the **Certification Search** options to search by Product, Release, and Platform. You can also search using the **Certification Quick Link** options such as Product Delivery, and Lifetime Support.
Run CVU and Download and Install the ORAchk Health Check Tool	Run CVU before installation to check your system for its compliance with minimum installation requirements, and download and install the ORAchk utility to perform proactive heath checks for the Oracle software stack.
	See, "Downloading and Installing the ORAchk Health Check Tool" on page 8-6.
Review the Oracle Universal Installer Installation Tutorial	Review the following step-by-step installation tutorial to understand OUI installation options:
	https://apexapps.oracle.com/pls/apex/f?p=44785:24:0::NO:24:P24_CONTENT_ID,P24_PREV_PAGE:6281,1
	Note: you can also run OUI up to the Summary screen, and either save a response file for review, or cancel the installation and run it at another time.

2

Overview of Oracle Database Installation

This chapter describes the different installation types of Oracle Database and issues to consider before you install Oracle Database:

- New Oracle Products and Features Installed with This Release
- Planning the Installation
- Installation Considerations
- Oracle Database Installation Methods
- Oracle Database Editions
- Database Security Notification Options
- Database Configuration Options
- Database Storage Options
- Database Management Options
- Database Backup and Recovery Options
- Upgrade Considerations

New Oracle Products and Features Installed with This Release

Refer to Changes in This Release for Oracle Database Installation Guide for information about the new features and products installed with this release.

Planning the Installation

The Oracle Database installation process consists of the following steps:

1. **Read the release notes:** Read *Oracle Database Release Notes for Linux* 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://docs.oracle.com

2. **Review the licensing information:** Although the installation media in the media pack contain many Oracle components, you are permitted to use only those components for which you have purchased licenses.

    Oracle Support Services does not provide support for components for which licenses have not been purchased.

> **See Also:** *Oracle Database Licensing Information*

3. **Plan the installation:** This chapter describes the Oracle products that you can install and issues that you must consider before starting the installation.

    You can also refer to Appendix J, which covers frequently asked questions about installing Oracle Database components, such as how to install Oracle Database if the site uses Oracle applications or if you need multiple Oracle Database connections.

4. **Configure Oracle Linux with Oracle Preinstallation RPMs:** Chapter 3 describes how to use Oracle Preinstallation RPMs to configure your operating system for Oracle Database and Oracle Grid Infrastructure installations.

5. **Complete preinstallation tasks:** Chapter 4 describes preinstallation tasks that you must complete before installing the product. Additionally, see Chapter 5 for information on configuring users, groups and environments, Chapter 6 for Oracle Restart preinstallation tasks, and Appendix E for checking the network setup.

6. **Install the software:** Use the following sections to install Oracle Database and Oracle Grid Infrastructure for a standalone server:

    - Chapter 6 describes how to install Oracle Grid Infrastructure for a standalone server.

    - Chapter 7 describes how to use Oracle Universal Installer to install Oracle Database and how to clone an Oracle home.

    - Appendix A provides information about performing silent, or response file installations, which you may want to use if you must perform multiple installations of Oracle Database.

    - Appendix B provides information about cloning an Oracle home.

    - Appendix D provides information about completing installation related configuration tasks manually.

    - Appendix H describes globalization support information.

    - Appendix I provides troubleshooting advice in case you encounter problems with the installation.

7. **Complete postinstallation tasks:** Chapter 8 describes recommended and required postinstallation tasks.

8. **Get started using Oracle Database:** Use the following sections to get started with Oracle Database:

    - Chapter 9 describes how to verify the contents of the installed Oracle Database, how to start various tools, and how to locate various files.

    - Appendix C describes the network-attached storage (NAS) devices, which you can use to store Oracle database files and Oracle software.

    - Appendix F describes the Optimal Flexible Architecture, which is a set of guidelines that ensures reliable Oracle installations that require little maintenance.

    - Appendix G explains the method to manage Oracle Database port numbers.

9. **Remove Oracle database software:** Chapter 10 describes how to remove Oracle Database software.

Installation Considerations

This section contains information that you must consider before deciding how to install this product. It contains the following sections:

- Hardware and Software Certification
- Multiple Oracle Homes Support
- Oracle Grid Infrastructure for a Standalone Server
- Oracle Cluster Synchronization Services
- Consider Memory Allocation and Automatic Memory Management
- Restrictions for HugePages Configurations

Hardware and Software Certification

The platform-specific hardware and software requirements included in this guide were current when this guide was published. However, because new platforms and operating system software versions might be certified after this guide is published, review the certification matrix on the My Oracle Support website for the most up-to-date list of certified hardware platforms and operating system versions. The My Oracle Support website is available at:

https://support.oracle.com/

You must register online before using My Oracle Support. After logging in, from the menu options, select the **Certifications** tab. On the **Certifications** page, use the **Certification Search** options to search by Product, Release, and Platform. You can also search using the **Certification Quick Link** options such as Product Delivery, and Lifetime Support.

Third-Party Database Certification for Oracle SQL Developer

You can use Oracle SQL Developer to view metadata and data of several non-Oracle databases. Refer to "Database Certification for SQL Developer (Oracle and Third-Party)" in *Oracle SQL Developer Installation Guide* for more information.

Multiple Oracle Homes Support

This product supports multiple Oracle homes. So, you can install this release or earlier releases of the software more than once on the same system, in different Oracle home directories.

Installing Oracle Database on a System with an Existing Oracle Installation

You must install Oracle Database into a new Oracle home directory. You cannot install products from one release of Oracle Database into an Oracle home directory of a different release. For example, you cannot install Oracle Database 12*c* software into an existing Oracle9*i* Oracle home directory.

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

Oracle Grid Infrastructure for a Standalone Server

The Oracle Grid Infrastructure for a standalone server provides the infrastructure to include your single-instance database in an enterprise grid architecture. Oracle Database 12*c* combines these infrastructure products into one software installation

called the Oracle Grid Infrastructure home. On a single-instance database, the Oracle Grid Infrastructure home includes Oracle Restart and Oracle Automatic Storage Management (Oracle ASM) software.

To use Oracle ASM or Oracle Restart, you must first install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart.

> **See Also:** Chapter 6, "Oracle Grid Infrastructure for a Standalone Server" for more information about installing Oracle Grid Infrastructure for a standalone server

Oracle Cluster Synchronization Services

When you install Oracle Grid Infrastructure for a standalone server, Oracle Universal Installer configures the single-node version of Oracle Cluster Synchronization Services (CSS). Specifically, CSS is a daemon process that is configured by the `root.sh` script.

The CSS service is required to enable synchronization between an Oracle ASM instance and the database instances that rely on it for database file storage. Because the service must be running before an Oracle ASM instance or database instance starts, it is configured to start automatically by Oracle Restart before the Oracle ASM instance is started. It must be running if an Oracle database is using Oracle ASM for database file storage.

For single-instance installations, the CSS daemon is installed in and runs from the Oracle Grid Infrastructure home which is the same home that runs Oracle ASM.

> **Note:** On cluster systems with Oracle RAC installations, the CSS daemon is configured during the Oracle Clusterware installation. If the system is running Oracle Clusterware, then see *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for information about removing Oracle RAC or Oracle Clusterware.

> **See Also:**
>
> "Oracle Automatic Storage Management" on page 2-10

Consider Memory Allocation and Automatic Memory Management

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

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

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

> **See Also:** "Managing Memory" in *Oracle Database Administrator's Guide*

Restrictions for HugePages Configurations

HugePages is a feature integrated into the Linux kernel with release 2.6 and can provide enhanced performance. You can choose to configure HugePages. However, this feature is an advanced configuration option and is not a requirement. Refer to your distribution documentation and to Oracle Technology Network and My Oracle Support for more information.

The HugePages feature allocates non-swappable memory for large page tables using memory-mapped files. If you enable HugePages, then you should deduct the memory allocated to HugePages from the available RAM before calculating swap space.

> **See Also:** *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems* for a complete list of HugePages restrictions

Oracle Database Installation Methods

You can choose different installation methods to install Oracle Database, as follows:

- Interactive Installation Types
- Automated Installation Methods Using Response Files

Interactive Installation Types

When you use the interactive method to install Oracle Database by selecting the **Create and configure a database** option, Oracle Universal Installer displays a series of screens that enable you to specify all the required information to install the Oracle Database software and create a database.

Oracle Universal Installer provides you the following options:

- Desktop Class: Select this option if you are installing on a laptop or desktop class system. This option includes a starter database and allows minimal configuration. This option is designed for those who want to quickly set up a database.

- Server Class: Select this option if you are installing on a server class system, such as what you would use when deploying Oracle Database in a production data center. This option allows for more advanced configuration options. Advanced configuration options available with this option include Oracle RAC, Oracle ASM, backup and recovery configuration, integration with Oracle Enterprise Manager Cloud Control, and more fine-grained memory tuning, among others.

  Furthermore, the Server Class option provides you with the following installation types:

  - Typical: Select this installation method to quickly install Oracle Database. This installation type requires minimal user input. Oracle Universal Installer installs the software and optionally creates a general-purpose database using the information that you specify on the screen. It is the default installation type.

  - Advanced: Select this installation type to complete any of the following tasks:

- – Select a database character set or different product languages.

- – Create the EXAMPLE tablespace during the installation.

- – Create a database on a different file system from the software.

- – Specify different passwords for administrative schemas.

- – Configure recovery options.

- – Configure Oracle Configuration Manager.

- – In the Select Database Edition screen, if you select **Enterprise Edition**, then Oracle Universal Installer automatically selects the components most customers need for their Oracle Database installation.

  See Also: "Reviewing Component-Specific Installation Guidelines" on page 7-1 for additional information about Oracle database installation

Automated Installation Methods Using Response Files

By creating a response file and specifying this file when you start Oracle Universal Installer, you can automate some or all of the Oracle Database installation. These automated installation methods are useful if you must perform multiple installations on similarly configured systems or if the system where you want to install the software does not have X Window System software installed.

When you use a response file, you can run Oracle Universal Installer in the following modes, depending on whether you specify all of the required information:

- Silent Mode: Oracle Universal Installer runs in silent mode if you use a response file that specifies all required information, and specify the `-silent` option when starting Oracle Universal Installer. None of the Oracle Universal Installer screens are displayed.

- Response File Mode: Oracle Universal Installer runs in response file mode if you do not specify all required information in the response file.

For more information about these modes and about how to complete an installation using response files, see Appendix A.

Oracle Database Editions

You can choose one of the following database editions when installing Oracle Database 12*c*:

- Enterprise Edition: Installs licensable Oracle Database options and database configuration and management tools in addition to all of the products that are installed during a Standard Edition installation. It also installs products most commonly used for data warehousing and transaction processing. This option also allows you to enable or disable individual components from a components list.

- Standard Edition: This installation type is designed for department-level or workgroup-level applications and for small and medium-sized enterprises (SMEs). It is engineered to provide core relational database management services and options. It installs an integrated set of management tools, full distribution, replication, web features, and it helps build business-critical applications.

- Standard Edition One: This installation type is designed for department-level, workgroup-level, or web applications. From single-instance environments for small business to highly distributed branch environments, Oracle Database

Standard Edition One includes all the features necessary to build business-critical applications.

- Standard Edition 2: This installation type is designed for department-level or workgroup-level applications and for small and medium-sized enterprises (SMEs). It is engineered to provide core relational database management services and options. It installs an integrated set of management tools, full distribution, replication, web features, and it helps build business-critical applications.

> **Note:**
>
> - You must install Oracle Database Client separately. You cannot install it during an Oracle Database installation. See *Oracle Database Client Installation Guide for Linux* for installation instructions.
>
> - The installation process is the same for all the database editions.
>
> - Ensure that you install only those products for which you have a valid license.

See Also: *Oracle Database Licensing Information* for more information about the features available with each Oracle Database edition and for information about licensing

Database Security Notification Options

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

During the database installation, in the Configure Security Updates screen Oracle Universal Installer prompts you to provide a security contact. Select one of the following options:

- Provide an email address to receive security information for your installation.

- Provide a My Oracle Support email address or account name to receive security information for your installation, and to enroll your system for Security Updates. You can receive information about alerts through My Oracle Support.

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

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

To choose not to receive security notifications, leave all fields in the Configure Security Updates screen blank, and click **Next** to continue.

If you provide your My Oracle Support credentials, then Security Updates automatically gathers configuration information regarding your installed Oracle products and uploads it to Oracle's support systems. You can access the information it collects through your My Oracle Support account, and review health check recommendations, patch recommendations and other recommendations for your system in addition to security alerts.

> **See Also:** The Oracle Security Policies page, which is available from the following URL:
>
> https://www.oracle.com/support/assurance/vulnerability-remed
> iation/security-fixing.html

Database Configuration Options

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

This section describes the following database configuration options:

- Preconfigured Database Types
- Installation Choices that Affect Database Creation

Preconfigured Database Types

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

- General Purpose/Transaction Processing
- Data Warehousing

See the online help provided by either Oracle Universal Installer or Oracle Database Configuration Assistant for a description of these preconfigured database types.

Installation Choices that Affect Database Creation

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

- Noninteractive Mode

  During an Oracle Database installation, if you choose to create a preconfigured database type, then Oracle Universal Installer prompts you for the minimum amount of information required to create a database of the type you choose. It then runs Oracle Database Configuration Assistant in silent or response file 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

  Install the database using Oracle Universal Installer and start Oracle Database Configuration Assistant from the Oracle home. Oracle Database Configuration Assistant runs in interactive mode. Using the screens in Oracle Database Configuration Assistant, you can either modify one of the preconfigured database types or customize the database.

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

Database Storage Options

If you create a database during the installation, you can specify one of the following storage options for database files:

- File System
- Oracle Automatic Storage Management

> **Note:** Installing files on raw devices is no longer an option during installation. You must use a file system or Oracle Automatic Storage Management (Oracle ASM).

File System

If you use the file system option, then Oracle Database Configuration Assistant creates the database files in a directory on a file system mounted on the computer. Oracle recommends that the file system be separate from the file systems used by the operating system or the Oracle software. The file system can be any of the following:

- A file system on a disk that is physically attached to the system

  If you are creating a database on basic disks that are not logical volumes or RAID devices, then Oracle recommends that you follow the Optimal Flexible Architecture (OFA) recommendations and distribute the database files over many disks.

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

  If you are using multiple disks in an 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 have to specify multiple file system mount points for the database storage.

- A network file system (NFS) mounted from a certified network-attached storage (NAS) device. You also have the option to use Direct NFS Client, which simplifies the administration of NFS configurations and also improves performance.

  If the NAS device is certified by Oracle, then you can store the database files on them.

  > **See Also:**
  >
  > - "General Configuration Guidelines for NAS Devices" on page C-1 for NAS certification information
  > - "Configuring Direct NFS Client" on page 8-10

If you use the Advanced database creation option, then you can also use the Oracle Managed Files feature with the new database. If you use this feature, then you must specify only the database object name instead of file names when creating or deleting database files.

> **See Also:** "Specifying Oracle Managed Files at Database Creation" in *Oracle Database Administrator's Guide*

Oracle Automatic Storage Management

Oracle Automatic Storage Management (Oracle ASM) is a high-performance storage management solution. For Oracle Database files, it simplifies the management of a dynamic database environment, for example, creating and laying out databases and managing disk space.

Oracle ASM can be used with single database installations, multiple database installations, and in Oracle RAC environments. It can be used with databases created in Oracle Database 10*g* Release 1 (10.1.0.3 or later). However, Oracle Database 12*c* databases must use Oracle ASM from Oracle Grid Infrastructure 12*c* or later. Oracle ASM is installed as part of the Oracle Grid Infrastructure installation. If you plan to use Oracle ASM, then you must install Oracle Grid Infrastructure before you install and create the database. If you want to upgrade an existing Oracle ASM installation, then you must upgrade Oracle ASM by running an Oracle Grid Infrastructure upgrade.

> **See Also:** Chapter 6, "Oracle Grid Infrastructure for a Standalone Server" for more information about installing the Oracle Grid Infrastructure for a standalone server

Oracle ASM manages the storage of all database files, such as redo logs, control files, and data pump export files.

Oracle ASM can manage the Oracle Database executable binary files and any other database files by creating a file system with Oracle Automatic Storage Management Cluster File System. Although Oracle Automatic Storage Management Cluster File System is cluster-aware, it also works as a file system on a single-instance database. Oracle Automatic Storage Management provides storage management for both cluster environments and single instance database environments.

> **See Also:** "Introducing Oracle ACFS and Oracle ADVM" in *Oracle Automatic Storage Management Administrator's Guide* for information about Oracle Automatic Storage Management Cluster File System

At a high level, implementing Oracle ASM involves allocating partitioned disks for Oracle Database with preferences for striping and mirroring. Oracle ASM manages the disk space for you. This helps avoid the need for traditional disk management tools, such as Logical Volume Managers (LVM), file systems, and the numerous commands necessary to manage both. The synchronization between Oracle ASM and the database instance is handled by CSS.

The following are components of an Oracle ASM installation:

- Oracle Automatic Storage Management Disk Groups
- Oracle Automatic Storage Management Instance

Oracle Automatic Storage Management Disk Groups

A disk group is a set of disk devices that Oracle ASM manages as a single unit. Each disk device can be an individual physical disk, a multiple disk device, such as a RAID storage array or logical volume, or a partition on a physical disk. In most cases, disk groups consist of one or more individual physical disks. To enable Oracle ASM to balance I/O operations and storage efficiently within the disk group, you must ensure

that all devices in the disk group have similar, if not identical, storage capacity and performance.

You can set the redundancy and striping attributes of individual file types within a disk group by using Oracle ASM disk group templates. When you create a disk group, Oracle ASM creates a set of default templates for that disk group. Default template settings depend on the disk group type. For example, the default template for control files for both normal and high redundancy disk groups is set to three-way mirroring. All other files are configured with two-way mirroring for normal redundancy, or three-way mirroring when configured for high redundancy. For a high redundancy disk group, the default mirroring cannot be changed, which implies that all files are always three-way mirrored in a high redundancy disk group. You can modify the default templates to suit your site's needs. See *Oracle Automatic Storage Management Administrator's Guide* for more information.

Oracle ASM spreads data evenly across all the devices in the disk group to optimize performance and utilization. You can add or remove disk devices from a disk group without shutting down the database. When you add or remove disks, Oracle ASM rebalances the files across the disk group. You can create multiple disk groups to do specific tasks, such as backup and recovery operations, in addition to regular file storage activities.

When you add a device to a disk group, you can specify a failure group for that device. Failure groups identify disk devices that have common failure characteristics, for example, devices that are attached to the same controller. If the controller fails, then all devices attached to it become unavailable. By default, each device also belongs to its own failure group. By using the failure groups you specify, Oracle ASM can distribute data among the devices in the disk group to minimize the risk of data loss caused by component failures.

Oracle Automatic Storage Management Instance

The Oracle ASM instance is a special Oracle instance that manages Oracle ASM disk groups. The Oracle ASM instance and the ASMSNMP account are created and started, if necessary, when you install Oracle Grid Infrastructure. Oracle Enterprise Manager uses this account to monitor Oracle ASM instances to retrieve data from Oracle ASM-related data dictionary views. The ASMSNMP account status is set to OPEN upon creation, and it is granted the SYSDBA privilege.

Oracle recommends that you have the Oracle ASM instance in its own Oracle home. Oracle also recommends that you run this instance before you start a database instance that uses Oracle ASM.

For an Oracle Database installation, you only need one Oracle ASM instance, regardless of the number of database instances on the computer.

> **See Also:** "Managing Oracle ASM Users with Oracle Enterprise Manager" in *Oracle Automatic Storage Management Administrator's Guide* for information about the ASMSNMP user

Database Management Options

To simplify database administration, Oracle provides a web-based management tool called Oracle Enterprise Manager. There are different ways to deploy Oracle Enterprise Manager:

- Deploy Oracle Enterprise Manager centrally in the environment

To deploy Oracle Enterprise Manager centrally, you must install at least one Oracle Management Repository and at least one Oracle Management Service within the environment, then install an Oracle Enterprise Management Agent on every computer to manage. You can then use a single HTML interface to manage and monitor software and hardware targets on all of those systems. Targets can include Oracle databases, application servers, net listeners, and third-party software. This single interface is called Oracle Enterprise Manager Cloud Control.

> **Note:**
>
> - Oracle Enterprise Manager Cloud Control is available separately on the Oracle Enterprise Manager Cloud Control installation media, and on the Oracle Technology Network website:
>
>   http://www.oracle.com/technetwork/oem/enterprise-manager/overview/index.html
>
> - For the latest Oracle Enterprise Manager certification information see the My Oracle Support certification page at:
>
>   https://support.oracle.com/

> **See Also:** *Oracle Enterprise Manager Cloud Control Basic Installation Guide*

- Oracle Enterprise Manager Database Express locally on the database system

  Oracle Enterprise Manager Database Express 12*c* is a web-based management tool built into Oracle Database without any need for special installation or management.

  Oracle Enterprise Manager Database Express can manage only a single database. If you want to administer multiple databases on a system, then you must either configure a separate Oracle Enterprise Manager Database Express for each database, or you must install Oracle Enterprise Manager Cloud Control.

This section contains the following topics:

- Management Options for Preconfigured Databases

- Management Options for Custom Databases

- Features Provided by Oracle Enterprise Manager Database Express 12c

Management Options for Preconfigured Databases

When you create a preconfigured database during the installation, you must select the Oracle Enterprise Manager interface to manage the database. The following options are available:

- Use Oracle Enterprise Manager Cloud Control for central database management

  This option is available only if an Oracle Management Agent is installed on the system. When Oracle Universal Installer detects an Oracle Management Agent on the system, you can choose this option and specify the Oracle Management Service to use to manage the database.

  If an Oracle Management Agent is not installed, then you must use Oracle Enterprise Manager Database Express to manage the database. However, if Oracle

Management Agent is installed after Oracle Database, then you can use Oracle Enterprise Manager Cloud Control to manage this database.

■ Use Oracle Enterprise Manager Database Express for local database management

Oracle Enterprise Manager Database Express is built into Oracle Database without any need for special installation or management. Oracle Enterprise Manager Database Express is available and configured by default during the database installation, and is not displayed as an option in Oracle Universal Installer during the database installation.

Management Options for Custom Databases

Install the database using Oracle Universal Installer and start Oracle Database Configuration Assistant from the Oracle home. Oracle Database Configuration Assistant runs in interactive mode. Using a screen in Oracle Database Configuration Assistant, you can specify the Oracle Enterprise Manager interface to use to manage the database. You can also choose not to configure the database with Oracle Enterprise Manager.

Oracle recommends that you configure the database to use Oracle Enterprise Manager during the database installation. However, if you do not do this, then see *Oracle Database 2 Day DBA* for configuring EM Express, or see *Oracle Enterprise Manager Cloud Control Administrator's Guide* for information about how to use Oracle Enterprise Manager Cloud Control to discover targets.

Features Provided by Oracle Enterprise Manager Database Express 12c

Oracle Enterprise Manager Database Express provides a web-based user interface that enables you to monitor, administer, and maintain an Oracle database.

You can use Oracle Enterprise Manager Database Express to perform basic database administration tasks, such as:

■ Configuration and Administration

– Initialization Parameters

– Memory Management

– Database Feature Usage

– Database Properties

■ Storage

– Tablespaces Management

– Undo Management

– Redo Log Groups

– Archive Logs

– Control Files

■ Security

– Users Management

– Roles Management

– Profiles Management

You can also use Oracle Enterprise Manager Database Express to complete Performance Monitoring and Tuning tasks, such as:

- Performance Hub

  - Real Time SQL Monitoring

  - ASH (Active Session History) Analytics

  - ADDM (Automatic Database Diagnostic Monitor)

  - AWR (Automatic Workload Repository) Browser

  - Historical performance monitoring and tuning

- SQL Tuning Advisor

Database Backup and Recovery Options

To simplify the management of backup and recovery files, you can create a fast recovery area for your database. During the database installation, Oracle Universal Installer provides you with options to configure the fast recovery area location. However, to configure backups, and to implement a backup and recovery strategy, you must use either Recovery Manager (RMAN) or Oracle Enterprise Manager Cloud Control.

> **See Also:**
>
> - *Oracle Database Backup and Recovery User's Guide* for information about Oracle backup and recovery solutions using RMAN or Oracle Enterprise Manager Cloud Control
>
> - *Oracle Database 2 Day DBA* for information about configuring customized backups and recovering a backed up database

Configuring Recovery

You can provide the location of the fast recovery area during the database installation.

You can use either a file system directory or an Oracle ASM disk group for the fast recovery area. To set the default values for fast recovery area and data file location, use Oracle base as the starting point. See "Oracle Base Directory" for more information on Oracle base.

- Default fast recovery area: `$ORACLE_BASE/recovery_area`

- Default data file location: `$ORACLE_BASE/oradata`

The default disk quota configured for the fast recovery area is 2 GB. For Oracle ASM disk groups, the required disk space depends on the redundancy level of the disk group that you choose.

> **See Also:**
>
> - "Identifying Storage Requirements for Oracle Automatic Storage Management"
>
> - "Creating a Fast Recovery Area Disk Group"

Upgrade Considerations

For information about upgrading an earlier release of Oracle Database to Oracle Database 12*c*, see *Oracle Database Upgrade Guide*. The following sections provide

additional platform-specific upgrade information that you must review before upgrading an existing database:

- Upgrading Your Operating System Before a Database Upgrade
- Upgrading Oracle Automatic Storage Management

Upgrading Your Operating System Before a Database Upgrade

When you upgrade to a new release of Oracle Database, the operating system requirements may have changed. If required, upgrade your operating system before upgrading Oracle Database. See Chapter 4, "Oracle Database Preinstallation Tasks" for a list of supported operating systems.

To upgrade the operating system and then perform a database upgrade, perform one of the following procedures:

- Upgrading the Operating System
- Migrating to a New Computer

Upgrading the Operating System

Upgrade the operating system. Then, upgrade the database either manually or by using Oracle Database Upgrade Assistant.

Migrating to a New Computer

Migrate to a new computer using one of the following methods:

- To upgrade the database on the new computer:
  1. Copy the database files from the computer running the previous operating system to the one running the supported operating system.
  2. Re-create the control files on the computer running the supported operating system.
  3. Manually upgrade the database using the method described in *Oracle Database Upgrade Guide*.

    Note: You cannot use Oracle Database Upgrade Assistant if you use this method. However, this method lets you easily revert to the earlier database.

- You can also upgrade the database using the Export/Import utilities method described in *Oracle Database Upgrade Guide*.

    See Also: The table on "Supported Upgrade Paths for Upgrading Oracle Database" in *Oracle Database Upgrade Guide* for information about upgrading your current database release

Upgrading Oracle Automatic Storage Management

In releases prior to Oracle Database 11*g* Release 2 (11.2), Oracle ASM was installed as part of the Oracle Database installation. In Oracle Database 11*g* Release 2 (11.2) and later releases, Oracle ASM is part of an Oracle Grid Infrastructure installation, either for a cluster or for a standalone server.

If you want to upgrade an existing Oracle ASM installation, then you must upgrade Oracle ASM by running an Oracle Grid Infrastructure upgrade. If you do not have Oracle ASM installed and you want to use Oracle ASM as your storage option, then you must complete an Oracle Grid Infrastructure installation before you start your Oracle Database installation.

See Also:

■ *Oracle Database Upgrade Guide*

■ *Oracle Automatic Storage Management Administrator's Guide*

3

Automatically Configuring Oracle Linux with Oracle Preinstallation RPM

Oracle recommends that you install Oracle Linux 7, Oracle Linux 6 or Oracle Linux 5 and use Oracle RPMs to configure your operating systems for Oracle Database and Oracle Grid Infrastructure installations. For Oracle Linux 7 and Oracle Linux 6, run the Oracle Preinstallation RPM. For Oracle Linux 5, run the Oracle Validated RPM.

This chapter contains the following contents:

- Overview of Oracle Linux Configuration with Oracle RPMs
- Installing a New Oracle Linux Installation from DVDs or Images
- Installing the Oracle Preinstallation RPM with ULN support
- Installing the Oracle Preinstallation RPM From Unbreakable Linux Network
- Installing the Oracle Preinstallation RPM From DVDs or Images
- Installing Oracle Linux with Oracle Linux Yum Server Support
- Additional Optional Operating System Configuration Tasks

Overview of Oracle Linux Configuration with Oracle RPMs

The Oracle RPMs for your Oracle Linux distributions and Oracle RDBMS releases automatically install any additional packages needed for installing Oracle Grid Infrastructure and Oracle Database, and configure your server operating system automatically, including setting kernel parameters and other basic operating system requirements for installation. For more information about what the Oracle RPMs do, refer to the following URL:

https://linux.oracle.com

Configuring a server using Oracle Linux and the Oracle Preinstallation RPM consists of the following steps:

1. Install Oracle Linux

2. Register your Linux distribution with Oracle Unbreakable Linux Network (ULN) or download and configure the yum repository for your system using the Oracle Linux yum server for your Oracle Linux release

3. Install the Oracle Preinstallation RPM or Oracle Validated RPM with the RPM for your Oracle Grid Infrastructure and Oracle Database releases, and update your Linux release.

4. Create role-allocated groups and users with identical names and ID numbers

5. Complete network interface configuration for each cluster node candidate

6. Complete system configuration for shared storage access as required for each standard or Core node cluster candidate.

After these steps are complete, you can proceed to install Oracle Grid Infrastructure and Oracle RAC

Installing a New Oracle Linux Installation from DVDs or Images

Use the following procedure to install a new Oracle Linux installation and to perform system configuration with the Oracle Preinstallation RPM:

1. Obtain Oracle Linux either by ordering the Oracle Linux media pack from Oracle Store, or by downloading disk images from the Oracle Software Delivery Cloud website for Oracle Linux and Oracle VM.

   Oracle Shop:

   https://shop.oracle.com

   Oracle Software Delivery Cloud website:

   https://edelivery.oracle.com/linux

2. Start the Oracle Linux installation, and respond to installation screens with values appropriate for your environment.

3. Review the first software selection screen, which lists task-specific software options. At the bottom of the screen, there is an option to customize now or customize later. Select **Customize now**, and click **Next**.

4. On Oracle Linux 7 and Oracle Linux 6, select **Servers** on the left hand side of the screen, and **System administration tools** on the right hand side of the screen (options may vary between releases)

   The Packages in System Tools window opens.

5. Select the Oracle Preinstallation RPM or Oracle Validated RPM package box from the package list.

   If you do not have an Oracle Preinstallation RPM or Oracle Validated package option that is current for your Oracle Database release, because you are using an Oracle Linux installation that is previous to your Oracle Database and Oracle Grid Infrastructure release, then install the RPM for your release manually after completing the operating system install.

6. Close the optional package window and click **Next**.

7. Complete the other screens to finish installing Oracle Linux.

   Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

8. Repeat steps 2 through 6 on all other cluster member nodes.

Installing the Oracle Preinstallation RPM with ULN support

Use the following procedure to subscribe to Unbreakable Linux Network (ULN) Oracle Linux channels, and to add the Oracle Linux channel that distributes the Oracle Preinstallation RPM:

1. Register your server with Unbreakable Linux Network (ULN). By default, you are registered for the Oracle Linux Latest channel for your operating system and hardware.

2. Log in to Unbreakable Linux Network:

    https://linux.oracle.com

3. Click the **Systems** tab, and in the System Profiles list, select a registered server. The System Details window opens and displays the subscriptions for the server.

4. Click **Manage Subscriptions**. The System Summary window opens.

5. From the **Available Channels** list, select the Linux installation media copy and update patch channels corresponding to your Oracle Linux distribution. For example, if your distribution is Oracle Linux 5 Update 6 for x86_64, then select the following:

    -   Oracle Linux 5 Update 6 installation media copy (x86_64)

    -   Oracle Linux 5 Update 6 Patch (x86_64)

6. Click **Subscribe**.

7. Start a terminal session and enter the following command as root, depending on your platform. For example:

    Oracle Linux 6 and Oracle Linux 7:

    ```
    # yum install oracle-rdbms-server-12cR1-preinstall
    ```

    Oracle Linux 5:

    ```
    # yum install oracle-validated
    ```

    You should see output indicating that you have subscribed to the Oracle Linux channel, and that packages are being installed. For example:

    ```
    el5_u6_i386_base
    el5_u6_x86_64_patch
    ```

    Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

8. Repeat steps 1 through 8 on all other servers in your cluster.

    > **Note:** Check the RPM log file to review the system configuration changes. For example, on Oracle Linux 5:
    >
    > ```
    > /var/log/oracle-validated/results/orakernel.log
    > ```

Installing the Oracle Preinstallation RPM From Unbreakable Linux Network

Use the following procedure to subscribe to Oracle Linux channels, and to add the Oracle Linux channel that distributes the Oracle Preinstallation RPM:

1. Complete a default Oracle Linux workstation installation.

    You can download Oracle Linux from the Oracle Software Delivery Cloud:

    https://edelivery.oracle.com/linux

2. Register your server with Unbreakable Linux Network (ULN). By default, you are registered for the Oracle Linux Latest channel for your operating system and hardware.

3. Log in to Unbreakable Linux Network:

https://linux.oracle.com

4. Click the **Systems** tab, and in the System Profiles list, select a registered server. The System Details window opens and displays the subscriptions for the server.

5. Click **Manage Subscriptions**. The System Summary window opens.

6. From the **Available Channels** list, select the Enterprise Linux installation media copy and update patch channels corresponding to your Oracle Linux distribution. For example, if your distribution is Oracle Linux 5 Update 5 for x86_64, then select the following:

- Oracle Linux 5 Update 5 installation media copy (x86_64)
- Oracle Linux 5 Update 5 Patch (x86_64)

7. Click **Subscribe**.

8. Start a terminal session and enter the following command as `root`, depending on your platform:

Oracle Linux 6 and Oracle Linux 7:

```
yum install oracle-rdbms-server-12cR1-preinstall
```

Oracle Linux 5:

```
# yum install oracle-validated
```

You should see output indicating that you have subscribed to the Oracle Linux channel. For example:

```
el5_u5_i386_base
el5_u5_x86_64_patch
```

Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

9. Enter the following command as `root` to update the `sysctl.conf` settings:

```
# sysctl -p
```

> **Note:** The RPM packages set the Oracle software user to `oracle` by default. Before installing Oracle Database, you can update the Oracle user name in the `etc/security/limits.d/oracle-rdbms-server-12cR1-preinstall.conf` file and other configuration files.

10. Repeat steps 1 through 9 on all other servers in your cluster.

> **Note:** Check the RPM log file to review the system configuration changes. For example, on Oracle Linux 5:
>
> ```
> /var/log/oracle-validated/results/orakernel.log
> ```

Installing the Oracle Preinstallation RPM From DVDs or Images

Use the following procedure to install the Oracle Preinstallation RPM from the Oracle Linux distribution:

1. Obtain Oracle Linux disks either by ordering the Oracle Linux media pack from Oracle Store, or by downloading disk images from the Oracle Software Delivery Cloud website for Oracle Linux and Oracle VM.

    Oracle Store:

    https://shop.oracle.com

    Oracle Software Delivery Cloud website:

    https://edelivery.oracle.com/linux

2. Start the Oracle Linux installation.

3. Review the first software selection screen, which lists task-specific software options. At the bottom of the screen, there is an option to customize now or customize later. Select **Customize now**, and click **Next**.

4. Select **Base System** on the Customization page, in the list on the left side of the screen, and then select **System Tools** on the right side of the screen. Then click **Optional Packages**.

    The Packages in System Tools window opens.

5. Select the Oracle Preinstallation RPM package box from the package list, and click **Next**.

6. Complete the other screens to finish installing Oracle Linux.

    Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

7. Repeat steps 2 through 6 on all other cluster member nodes.

Installing Oracle Linux with Oracle Linux Yum Server Support

Use the following procedure to install Oracle Linux and configure your Linux installation for security errata or bug fix updates using the Oracle Linux yum server:

1. Obtain Oracle Linux DVDs from Oracle Store, or download Oracle Linux from the Oracle Software Delivery Cloud:

    Oracle Store:

    https://shop.oracle.com

    Oracle Software Delivery Cloud website:

    https://edelivery.oracle.com/linux

2. Install Oracle Linux from the ISO or DVD image.

3. Log in as root

4. Download the yum repository file for your Linux distribution from http://yum.oracle.com/, using the instructions you can find on the Oracle Linux yum server website. For example:

    ```
    # cd /etc/yum.repos.d/
    ```

```
# wget http://yum.oracle.com/public-yum-ol6.repo
```

Ensure that the ol*release*_latest file (for example, `ol6_latest` for Oracle Linux 6) is enabled, as this is the repository that contains the Oracle Preinstallation RPM.

5. (Optional) Edit the `repo` file to enable other repositories. For example, enable the repository `ol6_UEK_latest` by setting `enabled=1` in the file with a text editor.

6. Run the command `yum repolist` to verify the registered channels.

7. Start a terminal session and enter the following command as `root`, depending on your platform. For example:

Oracle Linux 6 and Oracle Linux 7:

```
yum install oracle-rdbms-server-12cR1-preinstall
```

Oracle Linux 5:

```
# yum install oracle-validated
```

You should see output indicating that you have subscribed to the Oracle Linux channel, and that packages are being installed. For example:

```
el5_u6_i386_base
el5_u6_x86_64_patch
```

Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

After installation, run the command `yum update` as needed to obtain the most current security errata and bug fixes for your Oracle Linux installation.

Additional Optional Operating System Configuration Tasks

Complete the following optional configuration tasks:

- Configure Oracle Ksplice Repository for Oracle Linux
- Configure Additional Operating System Features

Configure Oracle Ksplice Repository for Oracle Linux

You can use Oracle Ksplice if you have Oracle Linux Premier support subscription and an access key, which is available on ULN. For more information about Ksplice (including trial versions, see http://www.ksplice.com/.

Complete the following task to register your system with Ksplice

1. Check for your kernel distribution at the following URL:

http://www.ksplice.com/uptrack/supported-kernels#

2. Log in as root.

3. Ensure that you have access to the Internet on the server where you want to use Ksplice. For example, if you are using a proxy server, then set the proxy server and port values in the shell with commands similar to the following:

```
# export http_proxy=http://proxy.example.com:port
# export https_proxy=http://proxy.example.com:port
```

4. Download the Ksplice Uptrack repository RPM package:

```
https://www.ksplice.com/yum/uptrack/ol/ksplice-uptrack-release.noarch.r
pm
```

5. Run the following commands:

```
rpm -i ksplice-uptrack-release.noarch.rpm
yum -y install uptrack
```

Configure Additional Operating System Features

As needed, configure the operating system for additional features, such as IPMI or additional programming environments, then review Chapter 4, "Oracle Database Preinstallation Tasks"

4

Oracle Database Preinstallation Tasks

This chapter describes the tasks that you must complete before you start Oracle Universal Installer.

Note: To use Oracle Automatic Storage Management (Oracle ASM) or Oracle Restart, you must first install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart.

See Also:

- "Configuring Servers for Oracle Grid Infrastructure for a Standalone Server" on page 6-2 before you proceed with the database preinstallation tasks

- Chapter 5, "Configuring Users, Groups and Environments for Oracle Database"

- Appendix E, "Configuring Networks for Oracle Database"

This chapter includes information about the following topics:

- Guidelines for Linux Operating System Installation

- Logging In to the System as root

- Configuring Servers for Oracle Database

- Reviewing Operating System Security Common Practices

- Using Installation Fixup Scripts

- Using Oracle RPM Checker on IBM: Linux on System z

- About Operating System Requirements

- Operating System Requirements for x86-64 Linux Platforms

- Operating System Requirements for IBM: Linux on System z

- Additional Drivers and Software Packages for Linux

- Checking the Software Requirements

- Installing the cvuqdisk RPM for Linux

- Checking Shared Memory File System Mount on Linux

- Confirming Host Name Resolution

- Disabling Transparent HugePages

- Identifying Required Software Directories

- Identifying or Creating an Oracle Base Directory

- Setting Disk I/O Scheduler on Linux

- Choosing a Storage Option for Oracle Database and Recovery Files

- Creating Directories for Oracle Database or Recovery Files

> **See Also:**
>
> - "Configuring Servers for Oracle Grid Infrastructure for a Standalone Server"
>
> - "Preinstallation Requirements" section in *Oracle Configuration Manager Installation and Administration Guide* and *Oracle Configuration Manager Prerequisites*
>
> - Appendix A, "Country Codes", in *Oracle Configuration Manager Installation and Administration Guide* for a list of valid country codes that can be used while installing Oracle Configuration Manager

Guidelines for Linux Operating System Installation

This section provides information about installing a supported Linux distribution. Complete the minimum hardware configuration before you install the operating system.

This section contains the following topics:

- Completing a Minimal Linux Installation

- Completing a Default Linux Installation

- About Oracle Linux and the Unbreakable Enterprise Kernel

- About the Oracle Preinstallation RPM

Completing a Minimal Linux Installation

Review the following sections regarding minimal Linux installation requirements:

- About Minimal Linux Installations

- RPM Packages for Completing Operating System Configuration

- Open SSH Requirement for Minimal Installation

About Minimal Linux Installations

To complete a minimal Linux installation, select one of the minimal installation options (either a custom installation where you select the Minimal option from Package Group Selection, or where you deselect all packages except for the Base pack). This installation lacks many RPMs required for database installation, so you must use an RPM package for your Oracle Linux release to install the required packages. The package you use depends on your Linux release, and your support status with Unbreakable Linux Network (ULN).

> **Note:** The Oracle Preinstallation RPM installs the X11 client libraries, but it does not install the X Window System server packages. To use graphical user interfaces such as OUI, configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

Refer to the following URL for documentation regarding installation of a reduced set of packages:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=728346.1

> **Note:** If you are not a member of Unbreakable Linux Network or Red Hat Support network, and you are a My Oracle Support customer, then you can download instructions to configure a script that documents installation of a reduced set of packages:
>
> https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=579101.1
>
> You can also search for "Linux reduced set of packages" to locate the instructions.

RPM Packages for Completing Operating System Configuration

Oracle Linux 6 Preinstallation RPM With ULN Support

Oracle Preinstallation RPM for your Oracle Linux 6 kernel (`oracle-rdbms-server-12cR1-preinstall`).

Unbreakable Linux Network (ULN) customers can obtain the Oracle Preinstallation RPM by using `yum`.

Oracle Linux 5 Oracle Validated RPM With ULN Support

Oracle Validated RPM (`oracle-validated`) for your Oracle Linux 5 kernel.

Unbreakable Linux Network (ULN) customers can obtain the Oracle Validated RPM by using `up2date`, or using `yum` (5.5 and later releases).

> **See Also:** Chapter 3, "Automatically Configuring Oracle Linux with Oracle Preinstallation RPM"

Oracle Linux 6 Preinstallation RPM Without ULN Support

http://yum.oracle.com/repo/OracleLinux/OL6/latest/x86_64/

Oracle Linux 5 Oracle Validated RPM Without ULN Support

http://yum.oracle.com/repo/OracleLinux/OL5/latest/x86_64/

Open SSH Requirement for Minimal Installation

SSH is required for Oracle Grid Infrastructure installation. OpenSSH should be included in the Linux distribution minimal installation. To confirm that SSH packages are installed, enter the following command:

```
# rpm -qa | grep ssh
```

If you do not see a list of SSH packages, then install those packages for your Linux distribution.

Completing a Default Linux Installation

If you do not install the Oracle Preinstallation RPM, then Oracle recommends that you install your Linux operating system with the default software packages (RPMs). This installation includes most of the required packages and helps you limit manual verification of package dependencies. Oracle recommends that you do not customize the RPMs during installation.

For information about a default installation, log on to My Oracle Support:

https://support.oracle.com

Search for "default rpms linux installation," and look for your Linux distribution. For example:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=401167.1

After installation, review system requirements for your distribution to ensure that you have all required kernel packages installed, and complete all other configuration tasks required for your distribution and system configuration.

About Oracle Linux and the Unbreakable Enterprise Kernel

Oracle's Unbreakable Enterprise Kernel delivers the latest innovations from upstream development to customers who run Oracle Linux in the data center. The Unbreakable Enterprise Kernel is included and enabled by default starting with Oracle Linux 5 Update 6.

The Unbreakable Enterprise Kernel is based on a recent stable mainline development Linux kernel, and also includes optimizations developed in collaboration with Oracle Database, Oracle middleware, and Oracle hardware engineering teams to ensure stability and optimal performance for the most demanding enterprise workloads.

Oracle highly recommends deploying the Unbreakable Enterprise Kernel in your Oracle Linux environment, especially if you are running enterprise applications. However, using Oracle Linux with the Unbreakable Enterprise Kernel is optional. If you require strict Red Hat Enterprise Linux kernel (RHEL) compatibility, then Oracle Linux also includes a kernel compatible with the RHEL Linux kernel, compiled directly from the RHEL source code.

You can obtain more information about the Unbreakable Enterprise Kernel for Oracle Linux at the following URL:

https://www.oracle.com/linux/index.html

The Unbreakable Enterprise Kernel for Oracle Linux is the standard kernel used with Oracle products. The build and QA systems for Oracle Database and other Oracle products use the Unbreakable Enterprise Kernel for Oracle Linux exclusively. The Unbreakable Enterprise Kernel for Oracle Linux is also the kernel used in Oracle Exadata and Oracle Exalogic systems. Unbreakable Enterprise Kernel for Oracle Linux is used in all benchmark tests on Linux in which Oracle participates, as well as in the Oracle Preinstallation RPM program for x86-64.

Oracle Ksplice, which is part of Oracle Linux, updates the Linux operating system (OS) kernel, while it is running, without requiring restarts or any interruption. Ksplice is available only with Oracle Linux.

About the Oracle Preinstallation RPM

If your Linux distribution is Oracle Linux, or Red Hat Enterprise Linux, and you are an Oracle Linux customer, then you can complete most preinstallation configuration tasks by using the Oracle Preinstallation RPM, available from the Oracle Linux Network, or available on the Oracle Linux DVDs. Using the Oracle Preinstallation RPM is not required, but Oracle recommends you use it to save time in setting up your cluster servers.

When installed, the Oracle Preinstallation RPM does the following:

- Automatically downloads and installs any additional RPM packages needed for installing Oracle Grid Infrastructure and Oracle Database, and resolves any dependencies

- Creates an `oracle` user, and creates the oraInventory (`oinstall`) and OSDBA (`dba`) groups for that user

- As needed, sets `sysctl.conf` settings, system startup parameters, and driver parameters to values based on recommendations from the Oracle Preinstallation RPM program

- Sets hard and soft resource limits

- Sets other recommended parameters, depending on your kernel version

To become an Oracle Linux Network customer, contact your sales representative, or purchase a license from the Oracle Linux store:

https://shop.oracle.com/product/oraclelinux

To register your server on the Unbreakable Linux Network, or to find out more information, see the following URL:

https://linux.oracle.com

If you are using Oracle Linux 5.2 and higher, then the Oracle Preinstallation RPM is included on the install media.

> **Note:** The Oracle Preinstallation RPM designated for each Oracle Database release sets kernel parameters and resource limits only for the user account `oracle`. To use multiple software account owners, you must perform system configuration for other accounts manually.

> **See Also:** Chapter 3, "Automatically Configuring Oracle Linux with Oracle Preinstallation RPM"

Using Oracle Ksplice to Perform a Zero Downtime Update

Oracle Ksplice Uptrack updates provide Linux security and bug fix updates, repackaged in a form that allows these updates to be applied without restarting the kernel.

To use Ksplice Uptrack:

1. Obtain or verify your Oracle Linux premium support subscription from Unbreakable Linux Network:

   https://linux.oracle.com

2. Log in as root.

3. Ensure that you have access to the Internet on the server where you want to use Ksplice. For example, if you are using a proxy server, then set the proxy server and port values in the shell with commands similar to the following:

```
# export http_proxy=http://proxy.example.com:port
# export https_proxy=http://proxy.example.com:port
```

4. Download the Ksplice Uptrack repository RPM package:

https://www.ksplice.com/yum/uptrack/ol/ksplice-uptrack-release.noarch.rpm

5. Run the following commands:

```
rpm -i ksplice-uptrack-release.noarch.rpm
yum -y install uptrack
```

6. Open /etc/uptrack/uptrack.conf with a text editor, enter your premium support access key, and save the file. You must use the same access key for all of your systems.

7. Run the following command to carry out a zero downtime update of your kernel:

```
uptrack-upgrade -y
```

> **See Also:**
>
> - The Oracle Ksplice Uptrack website for more information:
>
>   http://www.ksplice.com
>
> - Oracle Ksplice for Oracle Linux
>
>   https://oss.oracle.com/ksplice/docs/ksplice-quickstart.pdf

Logging In to the System as root

During installation, you must perform tasks as root or as other users on remote terminals. Complete the following procedure for user accounts that you want to enable for remote display.

> **Note:** If you log in as another user (for example, grid), then repeat this procedure for that user as well.

To log in as the root user and enable remote display, complete one of the following procedures:

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

  1. Start a new X terminal session (xterm).

  2. If you are installing the software on another system and using the system as an X11 display, then enter a command using the following syntax to enable remote hosts to display X applications on the local X server:

     ```
     $ xhost + RemoteHost
     ```

     where RemoteHost is the fully qualified remote host name. For example:

     ```
     $ xhost + somehost.example.com
     ```

```
somehost.example.com being added to the access control list
```

3. If you are not installing the software on the local system, then use the `ssh`, `rlogin`, or `telnet` command to connect to the system where you want to install the software:

```
$ ssh -Y RemoteHost
```

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

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

■ To install the software from a PC or other system with X server software:

> **Note:** If necessary, see the X server documentation, or contact your X server vendor or system administrator for more information about completing this procedure. Depending on the X server software that you are using, you may have to complete the tasks in a different order.

1. Start the X Window System software.

2. Configure the security settings of the X Window System software to permit remote hosts to display X applications on the local system.

3. Connect to the remote system where you want to install the software as the oracle software installation owner (`oracle`) and start an X terminal session (`xterm`) on that system.

4. Open another terminal on the remote system, and log in as the `root` user on the remote system, so you can run scripts as `root` when prompted.

Configuring Servers for Oracle Database

This section describes the following operating system tasks you must complete before you install Oracle Database:

■ Checking Server Hardware and Memory Configuration

■ General Server Minimum Requirements

■ Server Storage Minimum Requirements

■ Server Memory Minimum Requirements

Checking Server Hardware and Memory Configuration

Run the following commands to check your current system information:

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

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

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

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

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

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

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

```
# df -h /tmp
```

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

```
# df -h
```

5. To determine the amount of free RAM and disk swap space on the system, enter the following command:

```
# free
```

6. To determine if the system architecture can run the software, enter the following command:

```
# uname -m
```

Verify that the processor architecture matches the Oracle software release to install. For example, you should see the following for a x86-64 bit system:

```
x86_64
```

If you do not see the expected output, then you cannot install the software on this system.

General Server Minimum Requirements

Ensure the following general minimum requirements on your system:

- Ensure that the system is started with runlevel 3 or runlevel 5.

- Ensure display cards provide at least 1024 x 768 display resolution, so that Oracle Universal Installer displays correctly while performing a system console-based installation

Server Storage Minimum Requirements

Ensure that your system meets the following minimum storage requirements, depending on your system architecture:

- Disk Space Requirements for Linux x86-64

- Disk Space Requirements for IBM: Linux on System z

- Disk Space Requirements for the Temporary Directory

Disk Space Requirements for Linux x86-64

Ensure that your Linux x86-64 system meets the disk space requirements for software files as described in Table 4–1

Table 4–1   Disk Space Requirements on Linux x86-64

Installation Type	Disk Space for Software Files
Enterprise Edition	6.4 GB
Standard Edition	6.1 GB
Standard Edition One	6.1 GB
Standard Edition 2	6.1 GB

Additional disk space, either on a file system or on an Oracle ASM disk group is required for the fast recovery area.

Disk Space Requirements for IBM: Linux on System z

Ensure that your system meets the disk space requirements for software files as described in Table 4–2

Table 4–2   Disk Space Requirements on IBM: Linux on System z:

Installation Type	Disk Space for Software Files
Enterprise Edition	5.5 GB
Standard Edition	5.4 GB
Standard Edition One	5.5 GB
Standard Edition 2	5.4 GB

Additional disk space, either on a file system or on an Oracle ASM disk group is required for the fast recovery area.

Disk Space Requirements for the Temporary Directory

1 GB of space in the `/tmp` directory on your Linux system.

If the free space available in the `/tmp` directory is less than what is required, then complete one of the following steps:

- Delete unnecessary files from the `/tmp` directory to meet the disk space requirement.

- Set the `TMP` and `TMPDIR` environment variables when setting the `oracle` user's environment.

     See Also: "Configuring Oracle Software Owner Environment" on page 5-13 for more information about setting `TMP` and `TMPDIR`

- Extend the file system that contains the `/tmp` directory.

Server Memory Minimum Requirements

Ensure that your system meets the following memory requirements:

Minimum: 1 GB of RAM

Recommended: 2 GB of RAM or more

Table 4–3 describes the relationship between the installed RAM and the configured swap space recommendation:

Table 4–3   Swap Space Requirement for Linux

RAM	Swap Space
Between 1 GB and 2 GB	1.5 times the size of the RAM
Between 2 GB and 16 GB	Equal to the size of the RAM
More than 16 GB	16 GB

See Also:

- Restrictions for HugePages Configurations

- Configuring Servers for Oracle Grid Infrastructure for a Standalone Server

- Consider Memory Allocation and Automatic Memory Management

Reviewing Operating System Security Common Practices

Secure operating systems are an important basis for general system security. Ensure that your operating system deployment is in compliance with common security practices as described in your operating system vendor security guide.

See Also: `http://docs.oracle.com/en/operating-systems/`

Using Installation Fixup Scripts

Oracle Universal Installer detects when the minimum requirements for an installation are not met, and creates shell scripts, called fixup scripts, to finish incomplete system configuration steps. If Oracle Universal Installer detects an incomplete task, then it generates fixup scripts (`runfixup.sh`). You can run the fixup script and click **Fix and Check Again**. The fixup script modifies both persistent parameter settings and parameters in memory, so you do not have to restart the system.

See Also: "Cluster Verification Utility Reference" in *Oracle Clusterware Administration and Deployment Guide* for information about using the `cluvfy` command

The Fixup script does the following tasks:

- If necessary sets kernel parameters to values required for successful installation, including:

  - Shared memory parameters.

  - Open file descriptor and UDP send/receive parameters.

- Creates and sets permissions on the Oracle Inventory (central inventory) directory.

- Creates or reconfigures primary and secondary group memberships for the installation owner, if necessary, for the Oracle Inventory directory and the operating system privileges groups.

- Sets shell limits if necessary to required values.

Oracle recommends that you do not modify the contents of the generated fixup script.

> **Note:** Using fixup scripts does not ensure that all the prerequisites for installing Oracle Database are met. You must still verify that all the preinstallation requirements are met to ensure a successful installation.

> **See Also:** Appendix D, "How to Complete Preinstallation Tasks Manually"

Using Oracle RPM Checker on IBM: Linux on System z

Use the Oracle RPM Checker utility to verify that you have the required Red Hat Enterprise Linux or SUSE packages installed on the operating system before you start the Oracle Database or Oracle Grid Infrastructure installation.

Download the Oracle RPM Checker utility from the link in the My Oracle Support note 1574412.1 available at the following URL:

https://support.oracle.com/epmos/faces/DocumentDisplay?id=1574412.1&displayIndex=1

Download the Oracle RPM Checker utility for your IBM: Linux on System z distribution, unzip the RPM, and install the RPM as `root`. Then, run the utility as `root` to check your operating system packages. For example:

```
# rpm -ivh ora-val-rpm-EL6-DB-12.1.0.1-1.s390x.rpm
```

On Red Hat Enterprise Linux, the utility checks and also installs all required RPMs. For example:

```
yum install ora-val-rpm-EL6-DB-12.1.0.1-1.s390x.rpm
```

About Operating System Requirements

Depending on the products that you intend to install, verify that you have the required operating system kernel and packages installed.

Requirements listed in this document are current as of the date listed on the title page. To obtain the most current information about kernel requirements, see the online version at the following URL:

http://docs.oracle.com

Oracle Universal Installer performs checks your system to verify that it meets the listed operating system package requirements. To ensure that these checks complete successfully, verify the requirements before you start OUI.

> **Note:** Oracle does not support running different operating system versions on cluster members, unless an operating system is being upgraded. You cannot run different operating system version binaries on members of the same cluster, even if each operating system is supported.

Operating System Requirements for x86-64 Linux Platforms

The Linux distributions and packages listed in this section are supported for this release on x86-64.

Identify operating system requirements for Oracle Grid Infrastructure, and identify additional operating system requirements for Oracle Database and Oracle RAC installations.

- Supported Oracle Linux 7 and Red Hat Enterprise Linux 7 Distributions for x86-64
- Supported Oracle Linux 6 and Red Hat Enterprise Linux 6 Distributions for x86-64
- Supported Oracle Linux 5 and Red Hat Enterprise Linux 5 Distributions for x86-64
- Supported SUSE Distributions for x86-64
- Supported NeoKylin Linux Advanced Server Distributions for x86-64

> **Note:** The platform-specific hardware and software requirements included in this guide were current when this guide was published. However, because new platforms and operating system software versions might be certified after this guide is published, review the certification matrix on the My Oracle Support website for the most up-to-date list of certified hardware platforms and operating system versions:
>
> https://support.oracle.com/

> **Note:**
>
> - The Unbreakable Enterprise Kernel for Oracle Linux can be installed on x86-64 servers running either Oracle Linux 5 Update 5, or Red Hat Enterprise Linux 5 Update 5. As of Oracle Linux 5 Update 6, the Unbreakable Enterprise Kernel for Oracle Linux is the default system kernel. An x86 (32-bit) release of Oracle Linux including the Unbreakable Enterprise Kernel is available with Oracle Linux 5 update 7 and later.
>
> - The 32-bit packages listed in the following sections are required only for 32-bit client installs.
>
> - Oracle Universal Installer requires an X Window System (for example, `libx`). The `libx` packages are part of a default Linux installation. If you install Linux using an Oracle Preinstallation RPM, then the `libx` packages are installed as part of that RPM. If you perform an install on a system with a reduced set of packages, then you must ensure that `libx` is installed.

Supported Oracle Linux 7 and Red Hat Enterprise Linux 7 Distributions for x86-64

Use the following information to check supported Oracle Linux 7 and Red Hat Linux 7 distributions:

> **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), Oracle Linux 7 and Red Hat Enterprise Linux 7 are supported on Linux x86-64 systems.

See Also: If you currently use, or plan to upgrade to, Red Hat Enterprise Linux 7.2 or Oracle Linux 7.2, then see information about the RemoveIPC settings:

- My Oracle Support Note 2081410.1:

  https://support.oracle.com/epmos/faces/DocumentDisplay?id
  =2081410.1&displayIndex=1

- Oracle Linux 7 Update 2 Release Notes:

  http://docs.oracle.com/en/operating-systems/

Table 4–4   x86-64 Supported Linux 7 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.

Table 4–4   (Cont.) x86-64 Supported Linux 7 Minimum Operating System Requirements

Item	Requirements
Oracle Linux 7	Subscribe to the Oracle Linux 7 channel on the Unbreakable Linux Network, or configure a yum repository from the Oracle Linux yum server website, and then install the Oracle Preinstallation RPM. This RPM installs all required kernel packages for Oracle Grid Infrastructure and Oracle Database installations, and performs other system configuration.
	Supported distributions:
	■ Oracle Linux 7 with the Unbreakable Enterprise kernel: 3.8.13-33.el7uek.x86_ 64 or later
	■ Oracle Linux 7 with the Red Hat Compatible kernel: 3.10.0-123.el7.x86_64 or later
Red Hat Enterprise Linux 7	Supported distribution:
	■ Red Hat Enterprise Linux 7: 3.10.0-123.el7.x86_64 or later

Packages for Oracle Linux 7 and Red Hat Enterprise Linux 7	The following packages (or later versions) must be installed:
	`binutils-2.23.52.0.1-12.el7.x86_64`
	`compat-libcap1-1.10-3.el7.x86_64`
	`compat-libstdc++-33-3.2.3-71.el7.i686`
	`compat-libstdc++-33-3.2.3-71.el7.x86_64`
	`gcc-4.8.2-3.el7.x86_64`
	`gcc-c++-4.8.2-3.el7.x86_64`
	`glibc-2.17-36.el7.i686`
	`glibc-2.17-36.el7.x86_64`
	`glibc-devel-2.17-36.el7.i686`
	`glibc-devel-2.17-36.el7.x86_64`
	`ksh`
	`libaio-0.3.109-9.el7.i686`
	`libaio-0.3.109-9.el7.x86_64`
	`libaio-devel-0.3.109-9.el7.i686`
	`libaio-devel-0.3.109-9.el7.x86_64`
	`libgcc-4.8.2-3.el7.i686`
	`libgcc-4.8.2-3.el7.x86_64`
	`libstdc++-4.8.2-3.el7.i686`
	`libstdc++-4.8.2-3.el7.x86_64`
	`libstdc++-devel-4.8.2-3.el7.i686`
	`libstdc++-devel-4.8.2-3.el7.x86_64`
	`libXi-1.7.2-1.el7.i686`
	`libXi-1.7.2-1.el7.x86_64`
	`libXtst-1.2.2-1.el7.i686`
	`libXtst-1.2.2-1.el7.x86_64`
	`make-3.82-19.el7.x86_64`
	`sysstat-10.1.5-1.el7.x86_64`

Supported Oracle Linux 6 and Red Hat Enterprise Linux 6 Distributions for x86-64

Use the following information to check supported Oracle Linux 6 and Red Hat Linux 6 distributions:

Table 4–5   x86-64 Supported Linux 6 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Oracle Linux 6	Subscribe to the Oracle Linux 6 channel on the Unbreakable Linux Network, or configure a yum repository from the Oracle Linux yum server website, and then install the Oracle Preinstallation RPM. This RPM installs all required kernel packages for Oracle Grid Infrastructure and Oracle Database installations, and performs other system configuration.
	Supported distributions:
	■ Oracle Linux 6 with the Unbreakable Enterprise Kernel (UEK):
	Update 2 or higher, 2.6.39-200.24.1.el6uek.x86_64 or later UEK2 kernels
	Update 4 or higher, 3.8.13-16 or later UEK3 kernels
	Update 7 or higher, 4.1.12-32 or later UEK4 kernels
	■ Oracle Linux 6 with the Red Hat Compatible kernel: 2.6.32-71.el6.x86_64 or later
Red Hat Enterprise Linux 6	Supported distributions:
	■ Red Hat Enterprise Linux 6: 2.6.32-71.el6.x86_64 or later
Packages for Oracle Linux 6 and Red Hat Enterprise Linux 6	The following packages (or later versions) must be installed:

```
binutils-2.20.51.0.2-5.11.el6 (x86_64)
compat-libcap1-1.10-1 (x86_64)
compat-libstdc++-33-3.2.3-69.el6 (x86_64)
compat-libstdc++-33-3.2.3-69.el6 (i686)
gcc-4.4.4-13.el6 (x86_64)
gcc-c++-4.4.4-13.el6 (x86_64)
glibc-2.12-1.7.el6 (i686)
glibc-2.12-1.7.el6 (x86_64)
glibc-devel-2.12-1.7.el6 (x86_64)
glibc-devel-2.12-1.7.el6 (i686)
ksh
libgcc-4.4.4-13.el6 (i686)
libgcc-4.4.4-13.el6 (x86_64)
libstdc++-4.4.4-13.el6 (x86_64)
libstdc++-4.4.4-13.el6 (i686)
libstdc++-devel-4.4.4-13.el6 (x86_64)
libstdc++-devel-4.4.4-13.el6 (i686)
libaio-0.3.107-10.el6 (x86_64)
libaio-0.3.107-10.el6 (i686)
libaio-devel-0.3.107-10.el6 (x86_64)
libaio-devel-0.3.107-10.el6 (i686)
libXext-1.1 (x86_64)
libXext-1.1 (i686)
libXtst-1.0.99.2 (x86_64)
libXtst-1.0.99.2 (i686)
libX11-1.3 (x86_64)
libX11-1.3 (i686)
libXau-1.0.5 (x86_64)
libXau-1.0.5 (i686)
libxcb-1.5 (x86_64)
libxcb-1.5 (i686)
libXi-1.3 (x86_64)
libXi-1.3 (i686)
make-3.81-19.el6
sysstat-9.0.4-11.el6 (x86_64)
```

Supported Oracle Linux 5 and Red Hat Enterprise Linux 5 Distributions for x86-64

Use the following information to check supported Oracle Linux 5 and Red Hat Linux 5 distributions:

Table 4–6   x86-64 Supported Linux 5 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Oracle Linux 5	Subscribe to the Oracle Linux 5 channel on the Unbreakable Linux Network, and then install the Oracle Validated RPM. This RPM installs all required kernel packages for Oracle Grid Infrastructure and Oracle Database installations, and performs other system configuration.
	Supported distributions:
	■ Oracle Linux 5 Update 6 with the Unbreakable Enterprise kernel: 2.6.32-100.0.19 or later
	■ Oracle Linux 5 Update 6 with the Red Hat compatible kernel: 2.6.18-238.0.0.0.1.el5 or later
Red Hat Enterprise Linux 5	Supported distributions:
	■ Red Hat Enterprise Linux 5 Update 6: 2.6.18-238.0.0.0.1.el5 or later
Package requirements for Oracle Linux 5 and Red Hat Enterprise Linux 5	The following packages (or later versions) must be installed:

```
binutils-2.17.50.0.6
compat-libstdc++-33-3.2.3
compat-libstdc++-33-3.2.3 (32 bit)
coreutils-5.97-23.el5_4.1
gcc-4.1.2
gcc-c++-4.1.2
glibc-2.5-58
glibc-2.5-58 (32 bit)
glibc-devel-2.5-58
glibc-devel-2.5-58 (32 bit)
ksh
libaio-0.3.106
libaio-0.3.106 (32 bit)
libaio-devel-0.3.106
libaio-devel-0.3.106 (32 bit)
libgcc-4.1.2
libgcc-4.1.2 (32 bit)
libstdc++-4.1.2
libstdc++-4.1.2 (32 bit)
libstdc++-devel 4.1.2
libXext-1.0.1
libXext-1.0.1 (32 bit)
libXtst-1.0.1
libXtst-1.0.1 (32 bit)
libX11-1.0.3
libX11-1.0.3 (32 bit)
libXau-1.0.1
libXau-1.0.1 (32 bit)
libXi-1.0.1
libXi-1.0.1 (32 bit)
make-3.81
sysstat-7.0.2
```

Supported SUSE Distributions for x86-64

Use the following information to check supported SUSE distributions:

> **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), SUSE Linux Enterprise Server 12 is certified on Linux x86-64 systems.

Table 4–7   x86-64 Supported SUSE Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.

Table 4–7   (Cont.) x86-64 Supported SUSE Minimum Operating System Requirements

Item	Requirements
SUSE Linux Enterprise Server	Supported distributions: ■ SUSE Linux Enterprise Server 11 SP2: 3.0.13-0.27 or later ■ SUSE Linux Enterprise Server 12 SP1: 3.12.49-11 or later

SUSE 11	The following packages (or later versions) must be installed:
	
	`binutils-2.21.1-0.7.25`
	`gcc-4.3-62.198`
	`gcc-c++-4.3-62.198`
	`glibc-2.11.3-17.31.1`
	`glibc-devel-2.11.3-17.31.1`
	`ksh-93u-0.6.1`
	`libaio-0.3.109-0.1.46`
	`libaio-devel-0.3.109-0.1.46`
	`libcap1-1.10-6.10`
	`libstdc++33-3.3.3-11.9`
	`libstdc++33-32bit-3.3.3-11.9`
	`libstdc++43-devel-4.3.4_20091019-0.22.17`
	`libstdc++46-4.6.1_20110701-0.13.9`
	`libgcc46-4.6.1_20110701-0.13.9`
	`make-3.81`
	`sysstat-8.1.5-7.32.1`
	`xorg-x11-libs-32bit-7.4`
	`xorg-x11-libs-7.4`
	`xorg-x11-libX11-32bit-7.4`
	`xorg-x11-libX11-7.4`
	`xorg-x11-libXau-32bit-7.4`
	`xorg-x11-libXau-7.4`
	`xorg-x11-libxcb-32bit-7.4`
	`xorg-x11-libxcb-7.4`
	`xorg-x11-libXext-32bit-7.4`
	`xorg-x11-libXext-7.4`

Table 4–7 (Cont.) x86-64 Supported SUSE Minimum Operating System Requirements

Item	Requirements
SUSE 12	The following packages (or later versions) must be installed:

```
binutils-2.25.0-13.1
gcc-4.8-6.189
gcc48-4.8.5-24.1
glibc-2.19-31.9
glibc-32bit-2.19-31.9
glibc-devel-2.19-31.9.x86_64
glibc-devel-32bit-2.19-31.9.x86_64
mksh-50-2.13
libaio1-0.3.109-17.15
libaio-devel-0.3.109-17.15
libcap1-1.10-59.61
libstdc++48-devel-4.8.5-24.1.x86_64
libstdc++48-devel-32bit-4.8.5-24.1.x86_64
libstdc++6-5.2.1+r226025-4.1.x86_64
libstdc++6-32bit-5.2.1+r226025-4.1.x86_64
libstdc++-devel-4.8-6.189.x86_64
libstdc++-devel-32bit-4.8-6.189.x86_64
libgcc_s1-5.2.1+r226025-4.1.x86_64
libgcc_s1-32bit-5.2.1+r226025-4.1.x86_64
make-4.0-4.1.x86_64
sysstat-10.2.1-3.1.x86_64
xorg-x11-driver-video-7.6_1-14.30.x86_64
xorg-x11-server-7.6_1.15.2-36.21.x86_64
xorg-x11-essentials-7.6_1-14.17.noarch
xorg-x11-Xvnc-1.4.3-7.2.x86_64
xorg-x11-fonts-core-7.6-29.45.noarch
xorg-x11-7.6_1-14.17.noarch
xorg-x11-server-extra-7.6_1.15.2-36.21.x86_64
xorg-x11-libs-7.6-45.14.noarch
xorg-x11-fonts-7.6-29.45.noarch
```

Supported NeoKylin Linux Advanced Server Distributions for x86-64

Use the following information to check supported NeoKylin Linux Advanced Server distributions:

Note: Starting with Oracle Database 12*c* Release 1 (12.1.0.2), NeoKylin Linux Advanced Server 6 is certified on Linux x86-64 systems.

Table 4–8   x86-64 Supported NeoKylin Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
NeoKylin Linux Advanced Server	Supported distribution: ■ NeoKylin Linux Advanced Server 6: 2.6.32-431.el6.x86_64 or later

NeoKylin 6.0	The following packages (or later versions) must be installed:
	
	`binutils-2.20.51.0.2-5.36.el6 (x86_64)`
	`compat-libcap1-1.10-1 (x86_64)`
	`compat-libstdc++-33-3.2.3-69.el6 (x86_64)`
	`compat-libstdc++-33-3.2.3-69.el6 (i686)`
	`gcc-4.4.7-4.el6 (x86_64)`
	`gcc-c++-4.4.7-4.el6 (x86_64)`
	`glibc-2.12-1.132.el6 (i686)`
	`glibc-2.12-1.132.el6 (x86_64)`
	`glibc-devel-2.12-1.132.el6 (x86_64)`
	`glibc-devel-2.12-1.132.el6 (i686)`
	`ksh`
	`libgcc-4.4.7-4.el6 (i686)`
	`libgcc-4.4.7-4.el6 (x86_64)`
	`libstdc++-4.4.7-4.el6 (x86_64)`
	`libstdc++-4.4.7-4.el6 (i686)`
	`libstdc++-devel-4.4.7-4.el6 (x86_64)`
	`libstdc++-devel-4.4.7-4.el6 (i686)`
	`libaio-0.3.107-10.el6 (x86_64)`
	`libaio-0.3.107-10.el6 (i686)`
	`libaio-devel-0.3.107-10.el6 (x86_64)`
	`libaio-devel-0.3.107-10.el6 (i686)`
	`libXext-1.3.1-2.el6 (x86_64)`
	`libXext-1.3.1-2.el6 (i686)`
	`libXtst-1.2.1-2.el6 (x86_64)`
	`libXtst-1.2.1-2.el6 (i686)`
	`libX11-1.5.0-4.el6 (x86_64)`
	`libX11-1.5.0-4.el6 (i686)`
	`libXau-1.0.6-4.el6 (x86_64)`
	`libXau-1.0.6-4.el6 (i686)`
	`libxcb-1.8.1-1.el6 (x86_64)`
	`libxcb-1.8.1-1.el6 (i686)`
	`libXi-1.6.1-3.el6 (x86_64)`
	`libXi-1.6.1-3.el6 (i686)`
	`make-3.81-20.el6`
	`sysstat-9.0.4-22.el6 (x86_64)`

Operating System Requirements for IBM: Linux on System z

The distributions and packages listed in this section are supported for this release on IBM: Linux on System z.

Identify operating system requirements for Oracle Grid Infrastructure, and identify additional operating sytem requirements for Oracle Database and Oracle RAC installations.

■   Supported Red Hat Enterprise Linux 7 Distributions for IBM: Linux on System z

■   Supported Red Hat Enterprise Linux 6 Distributions for IBM: Linux on System z

- Supported Red Hat Enterprise Linux 5 Distributions for IBM: Linux on System z
- Supported SUSE Distributions for IBM: Linux on System z

> **Note:**
>
> - For the 12.1 release, starting with Oracle Database 12*c* Release 1 (12.1.0.2), Red Hat Enterprise Linux 7.1 and SUSE Linux Enterprise Server 12 SP1 are certified on IBM: Linux on System z.
>
> - The 32-bit packages listed in the following sections are required only for 32-bit client installs.

Supported Red Hat Enterprise Linux 7 Distributions for IBM: Linux on System z

Use the following information to check the supported Red Hat Linux 7 distributions:

Table 4–9   Linux 7 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Red Hat Enterprise Linux 7	Red Hat Enterprise Linux 7.1: 3.10.0-229.el7.s390x or later
	Note: You can install on Red Hat Enterprise Linux 7 Update 1, but Oracle recommends that you install on Red Hat Enterprise Linux 7 Update 2 for seamless security enhancements.
	See My Oracle Support Note 2213265.1 for more information:
	https://support.oracle.com/rs?type=doc&id=2213265.1
Packages for Red Hat Enterprise Linux 7	The following packages (or later versions) must be installed:

	`binutils-2.23.52.0.1-30.el7.s390x`
	`compat-libcap1-1.10-7.el7.s390x`
	`compat-libstdc++-33-3.2.3-71.el7 (s390)`
	`compat-libcap1-1.10-1 (s390x)`
	`cpp-4.8.2-16.el7.s390x`
	`gcc-4.8.3-9.el7.s390x`
	`gcc-4.8.3-9.el7.s390x`
	`glibc-2.17-78.el7 (s390)`
	`glibc-devel-2.17-78.el7 (s390x)`
	`glibc-devel-2.17-78.el7 (s390)`
	`glibc-devel-2.17-78.el7 (s390x)`
	`glibc-headers-2.17-78.el7 (s390x)`
	`ksh-20120801-22.el7 (s390x)`
	`libaio-0.3.109-12.el7 (s390)`
	`libaio-0.3.109-12.el7 (s390)`
	`libaio-devel-0.3.109-12.el7 (s390x)`
	`libgcc-4.8.3-9.el7 (s390)`
	`libgcc-4.8.3-9.el7 (s390x)`
	`libstdc++-4.8.3-9.el7 (s390)`
	`libstdc++-4.8.3-9.el7 (s390x)`
	`libstdc++-devel-4.8.3-9.el7 (s390)`
	`libstdc++-devel-4.8.3-9.el7 (s390x)`
	`libXtst-1.2.2-2.1.el7 (s390)`
	`libXtst-1.2.2-2.1.el7 (s390x)`
	`libXi-1.7.2-2.1.el7 (s390x)`
	`libXi-1.7.2-2.1.el7 (s390x)`
	`libxcb-1.9-5.1.el7 (s390)`
	`libxcb-1.9-5.el7 (s390x)`
	`llibX11-1.6.0-2.el7 (s390)`
	`libX11-1.6.0-2.el7 (s390x)`
	`libXau-1.0.8-2.1.el7 (s390)`
	`libXau-1.0.8-2.1.el7 (s390x)`
	`libXext-1.3.2-2.1.el7 (s390)`
	`libXext-1.3.2-2.1.el7 (s390x)`
	`make-3.82-21.el7 (s390x)`
	`mpfr-3.1.1-4.el7.s390x`
	`sysstat-10.1.5-7.el7 (s390x)`

Supported Red Hat Enterprise Linux 6 Distributions for IBM: Linux on System z

Use the following information to check the supported Red Hat Linux 6 distributions:

Table 4–10   Linux 6 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Red Hat Enterprise Linux 6	Red Hat Enterprise Linux 6.3: 2.6.32-279.el6.s390x or later
	Note: You can install on Red Hat Enterprise Linux 6 Update 3, but Oracle recommends that you install on Red Hat Enterprise Linux 6 Update 4 as RHEL 6.4 includes significant I/O performance gains on Open Storage.
	See My Oracle Support Note 1574412.1 for more information:
	`https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=157441` `2.1`
Packages for Red Hat Enterprise Linux 6	The following packages (or later versions) must be installed:

	`binutils-2.20.51.0.2-5.34.el6 (s390x)`
	`compat-libstdc++-33-3.2.3-69.el6 (s390)`
	`compat-libstdc++-33-3.2.3-69.el6 (s390x)`
	`compat-libcap1-1.10-1 (s390x)`
	`gcc-4.4.6-4.el6 (s390x)`
	`gcc-c++-4.4.6-4.el6 (s390x)`
	`glibc-2.12-1.80.el6 (s390)`
	`glibc-2.12-1.80.el6 (s390x)`
	`glibc-devel-2.12-1.80.el6 (s390)`
	`glibc-devel-2.12-1.80.el6 (s390x)`
	`ksh-20100621-16.el6 (s390x)`
	`libaio-0.3.107-10.el6 (s390)`
	`libaio-0.3.107-10.el6 (s390x)`
	`libaio-devel-0.3.107-10.el6 (s390x)`
	`libgcc-4.4.6-4.el6 (s390)`
	`libgcc-4.4.6-4.el6 (s390x)`
	`libstdc++-4.4.6-4.el6 (s390x)`
	`libstdc++-devel-4.4.6-4.el6 (s390x)`
	`libXtst-1.0.99.2-3.el6 (s390)`
	`libXtst-1.0.99.2-3.el6 (s390x)`
	`libXi-1.3-3.el6 (s390)`
	`libXi-1.3-3.el6 (s390x)`
	`libXmu-1.0.5-1.el6 (s390)`
	`libXaw-1.0.6-4.1.el6 (s390)`
	`libXft-2.1.13-4.1.el6 (s390)`
	`libXp-1.0.0-15.1.el6 (s390)`
	`make-3.81-20.el6 (s390x)`
	`sysstat-9.0.4-20.el6 (s390x)`

Supported Red Hat Enterprise Linux 5 Distributions for IBM: Linux on System z

Use the following information to check supported Red Hat Linux 5 distributions:

Table 4–11   Linux 5 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Red Hat Enterprise Linux 5	Red Hat Enterprise Linux 5.8: 2.6.18-308.el5 s390x or later
Package requirements for Red Hat Enterprise Linux 5	The following packages (or later versions) must be installed: ``` binutils-2.17.50.0.6-20.el5 (s390x) compat-libstdc++-33-3.2.3-61 (s390) compat-libstdc++-33-3.2.3-61 (s390x) gcc-c++-4.1.2-52.el5 (s390x) glibc-2.5-81 (s390) glibc-2.5-81 (s390x) glibc-devel-2.5-81 (s390) glibc-devel-2.5-81 (s390x) ksh-20100621-5.el5 (s390x) libaio-0.3.106-5 (s390) libaio-0.3.106-5 (s390x) libaio-devel-0.3.106-5 (s390) libaio-devel-0.3.106-5 (s390x) libgcc-4.1.2-52.el5 (s390) libgcc-4.1.2-52.el5 (s390x) libstdc++-4.1.2-52.el5 (s390) libstdc++-4.1.2-52.el5 (s390x) libstdc++-devel-4.1.2-52.el5 (s390x) libstdc++44-devel-4.4.6-3.el5.1 (s390) libstdc++44-devel-4.4.6-3.el5.1 (s390x) libXtst-1.0.1-3.1 (s390) libXtst-1.0.1-3.1 (s390x) libXi-1.0.1-4.el5_4 (s390) libXi-1.0.1-4.el5_4 (s390x) make-3.81-3.el5 (s390x) sysstat-7.0.2-11.el5 (s390x) ```

Supported SUSE Distributions for IBM: Linux on System z

Use the following information to check supported SUSE distributions:

Table 4–12   SUSE Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.

Table 4–12   (Cont.) SUSE Minimum Operating System Requirements

Item	Requirements
SUSE Linux Enterprise Server	▪ SUSE Linux Enterprise Server 11 SP2: 3.0.13-0.27-default s390x or later ▪ SUSE Linux Enterprise Server 12 SP1: 3.12.53-60.30.1-default s390x or later
SUSE 11	The following packages (or later versions) must be installed:

```
binutils-2.21.1-0.7.25 (s390x)
gcc-4.3-62.198 (s390x)
gcc-c++-4.3-62.198 (s390x)
glibc-2.11.3-17.31.1 (s390x)
glibc-32bit-2.11.3-17.31.1 (s390x)
glibc-devel-2.11.3-17.31.1 (s390x)
glibc-devel-32bit-2.11.3-17.31.1 (s390x)
ksh-93u-0.6.1 (s390x)
make-3.81-128.20 (s390x)
libaio-0.3.109-0.1.46 (s390x)
libaio-32bit-0.3.109-0.1.46 (s390x)
libaio-devel-0.3.109-0.1.46 (s390x)
libaio-devel-32bit-0.3.109-0.1.46 (s390x)
libcap1-1.10-6.10 (s390x)
libgcc46-4.6.1_20110701-0.13.9 (s390x)
libstdc++33-3.3.3-11.9 (s390x)
libstdc++33-32bit-3.3.3-11.9 (s390x)
libstdc++43-devel-32bit-4.3.4_20091019-0.22.17 (s390x)
libstdc++43-devel-4.3.4_20091019-0.22.17 (s390x)
libstdc++46-32bit-4.6.1_20110701-0.13.9 (s390x)
libstdc++46-4.6.1_20110701-0.13.9 (s390x)
sysstat-8.1.5-7.32.1 (s390x)
xorg-x11-libs-32bit-7.4-8.26.32.1 (s390x)
xorg-x11-libs-7.4-8.26.32.1 (s390x)
xorg-x11-libX11-32bit-7.4-5.9.1 (s390x)
xorg-x11-libX11-7.4-5.9.1 (s390x)
xorg-x11-libXau-32bit-7.4-1.15 (s390x)
xorg-x11-libXau-7.4-1.15 (s390x)
xorg-x11-libxcb-7.4-1.20.34 (s390x)
xorg-x11-libxcb-32bit-7.4-1.20.34 (s390x)
xorg-x11-libXext-32bit-7.4-1.16.21 (s390x)
xorg-x11-libXext-7.4-1.16.21 (s390x)
```

Table 4–12   (Cont.) SUSE Minimum Operating System Requirements

Item	Requirements
SUSE 12	The following packages (or later versions) must be installed:

```
binutils-2.25.0-13.1 (s390x)
gcc-32bit-4.8-6.189 (s390x)
gcc48-4.8.5-24.1 (s390x)
gcc48-32bit-4.8.5-24.1 (s390x)
gcc48-c++-4.8.5-24.1 (s390x)
gcc48-info-4.8.5-24.1 (s390x)
gcc48-locale-4.8.5-24.1 (s390x)
gcc-c++-4.8-6.189 (s390x)
gcc-c++-32bit-4.8-6.189 (s390x)
gcc-info-4.8-6.189 (s390x)
gcc-locale-4.8-6.189 (s390x)
glibc-2.19-31.9 (s390x)
glibc-32bit-2.19-31.9 (s390x)
glibc-devel-2.19-31.9 (s390x)
glibc-devel-32bit-2.19-31.9 (s390x)
libaio1-0.3.109-17.15 (s390x)
libaio1-32bit-0.3.109-17.15 (s390x)
libaio-devel-0.3.109-17.15 (s390x)
libcap1-1.10-59.61 (s390x)
libcap1-32bit-1.10-59.61 (s390x)
libcap2-2.22-11.709 (s390x)
libcap2-32bit-2.22-11.709 (s390x)
libcap-ng0-0.7.3-4.125 (s390x)
libcap-ng0-32bit-0.7.3-4.125 (s390x)
libcap-ng-utils-0.7.3-4.125 (s390x)
libcap-progs-2.22-11.709 (s390x)
libgcc_s1-5.2.1+r226025-4.1 (s390x)
libgcc_s1-32bit-5.2.1+r226025-4.1 (s390x)
libgomp1-32bit-5.2.1+r226025-4.1 (s390x)
libstdc++48-devel-4.8.5-24.1 (s390x)
libstdc++48-devel-32bit-4.8.5-24.1 (s390x)
libstdc++6-5.2.1+r226025-4.1 (s390x)
libstdc++6-32bit-5.2.1+r226025-4.1 (s390x)
libstdc++-devel-4.8-6.189 (s390x)
libstdc++-devel-32bit-4.8-6.189 (s390x)
libXtst6-1.2.2-3.60 (s390x)
libXtst6-32bit-1.2.2-3.60 (s390x)
make-4.0-4.1 (s390x)
mksh-50-2.13 (s390x)
sysstat-10.2.1-3.1 (s390x)
xorg-x11-7.6_1-14.17 (s390x)
xorg-x11-essentials-7.6_1-14.17 (s390x)
xorg-x11-fonts-7.6-29.45 (s390x)
xorg-x11-fonts-core-7.6-29.45 (s390x)
xorg-x11-libs-7.6-45.14 (s390x)
xorg-x11-server-7.6_1.15.2-36.21 (s390x)
xorg-x11-server-extra-7.6_1.15.2-36.21 (s390x)
xorg-x11-Xvnc-1.4.3-7.2 (s390x)
```

Additional Drivers and Software Packages for Linux

You are not required to install additional drivers and packages, but you may choose to install or configure drivers and packages in the following list:

- [Installation Requirements for Open Database Connectivity](#)
- [Installation Requirements for PAM on Linux](#)
- [Installation Requirements for Oracle Messaging Gateway](#)
- [Installation Requirements for Lightweight Directory Access Protocol](#)
- [Installation Requirements for Programming Environments for Linux](#)
- [Installation Requirements for Web Browsers](#)

> **Note:** Oracle Database Smart Flash Cache is an Enterprise Edition only feature.

Installation Requirements for Open Database Connectivity

Review the following sections if you plan to install Open Database Connectivity (ODBC):

- [About ODBC Drivers and Oracle Database](#)
- [Installing Oracle ODBC Driver for Linux](#)

About ODBC Drivers and Oracle Database

Open Database Connectivity (ODBC) is a set of database access APIs that connect to the database, prepare, and then run SQL statements on the database. An application that uses an ODBC driver can access non-uniform data sources, such as spreadsheets and comma-delimited files.

Installing Oracle ODBC Driver for Linux

If you intend to use ODBC, then install the most recent ODBC Driver Manager for Linux. Download and install the ODBC Driver Manager and Linux RPMs from the following website:

http://www.unixodbc.org

Review the minimum supported ODBC driver, and install the following ODBC driver, or later releases, for all distributions of Linux:

```
unixODBC-2.3.1 or later
```

Installation Requirements for PAM on Linux

Review the following sections to install PAM:

- [About PAM and Login Authentication](#)
- [Installing PAM Library](#)

About PAM and Login Authentication

Pluggable Authentication Modules (PAM) is a system of libraries that handle user authentication tasks for applications. On Linux, external scheduler jobs require PAM. Oracle strongly recommends that you install the latest Linux-PAM library for your Linux distribution.

Installing PAM Library

Use a package management system (`yum`, `up2date`, `YaST`) for your distribution to install the latest `pam` library.

Installation Requirements for Oracle Messaging Gateway

Review the following sections to install Oracle Messaging Gateway:

- About Oracle Messaging Gateway
- Installing Oracle Messaging Gateway

About Oracle Messaging Gateway

Oracle Messaging Gateway is a feature of the Oracle database that enables communication between applications based on non-Oracle messaging systems and Oracle Streams Advanced Queuing.

Oracle Messaging Gateway supports the integration of Oracle Streams Advanced Queuing (AQ) with applications based on WebSphere MQ and TIBCO Rendezvous. For information on supported versions see *Oracle Database Advanced Queuing User's Guide*.

> **Note:** Oracle Messaging Gateway does not support the integration of Advanced Queuing with TIBCO Rendezvous on IBM: Linux on System z.

Installing Oracle Messaging Gateway

Oracle Messaging Gateway is installed with the Enterprise Edition of Oracle Database.

If you require a CSD or Fix Pack for IBM WebSphere MQ, then see the following website for more information:

http://www.ibm.com

Installation Requirements for Lightweight Directory Access Protocol

Review the following sections to install Lightweight Directory Access Protocol:

- About LDAP and Oracle Plug-ins
- Installing the LDAP Package

About LDAP and Oracle Plug-ins

Lightweight Directory Access Protocol (LDAP) is an application protocol for accessing and maintaining distributed directory information services over IP networks. You require the LDAP package if you want to use features requiring LDAP, including the Oracle Database scripts `odisrvreg` and `oidca` for Oracle Internet Directory, or `schemasync` for third-party LDAP directories.

Installing the LDAP Package

LDAP is included in a default Linux operating system installation.

If you did not perform a default Linux installation, and you intend to use Oracle scripts requiring LDAP, then use a package management system (`up2date`, `YaST`) for your distribution to install a supported LDAP package for your distribution, and install any other required packages for that LDAP package.

Installation Requirements for Programming Environments for Linux

Review the following sections to install programming environments:

- About Programming Environments and Oracle Database
- Configuring Support for Programming Environments

About Programming Environments and Oracle Database

Oracle Database supports multiple programming languages for application development in different environments. Some languages require that you install additional compiler packages for the operating system.

Programming environments are options. They are not required for Oracle Database.

> **See Also :** *Oracle Database Development Guide* for an overview of programming environments

Configuring Support for Programming Environments

Ensure that your system meets the requirements for the programming environment you want to configure:

- Requirements for Programming Environments for Linux x86-64
- Requirements for Programming Environments for IBM: Linux on System z

Table 4–13   Requirements for Programming Environments for Linux x86-64

Programming Environments	Support Requirements
Java Database Connectivity (JDBC) / Oracle Call Interface (OCI)	JDK 6 (Java SE Development Kit release 1.6.0_37 or later updates of 1.6) with the JNDI extension with Oracle Java Database Connectivity. JDK 1.6 is installed with this release.
Oracle C++ Oracle C++ Call Interface Pro*C/C++ Oracle XML Developer's Kit (XDK)	Intel C/C++ Compiler 12.0.5 or later, and the version of GNU C and C++ compilers listed in the software requirements section in this document for your platform. Oracle C++ Call Interface (OCCI) applications can be built only with Intel C++ Compiler 12.0.5 used with the standard template libraries of the gcc versions listed in the software requirements section in this document for your platform. Oracle XML Developer's Kit is supported with the same compilers as OCCI.
Pro*COBOL	Micro Focus Server Express 5.1

Table 4–14   Requirements for Programming Environments for IBM: Linux on System z

Programming Environments	Support Requirements
Java Database Connectivity (JDBC) / Oracle Call Interface (OCI)	- JDK 6 (1.6.0 SR12) - JDK 7 (1.7.0) JDK 1.6 is installed with this release.
Pro*COBOL	Micro Focus Server Express 5.1

Installation Requirements for Web Browsers

Web browsers are required only if you intend to use Oracle Enterprise Manager Database Express and Oracle Enterprise Manager Cloud Control. Web browsers must support JavaScript, and the HTML 4.0 and CSS 1.0 standards. For a list of browsers

that meet these requirements see the Enterprise Manager certification matrix on My Oracle Support:

https://support.oracle.com

> **See Also:** *Oracle Enterprise Manager Cloud Control Basic Installation Guide* for steps on how to access the Enterprise Manager certification matrix

Checking the Software Requirements

To check the software requirements, perform the following steps:

1. To determine the distribution and version of Linux installed, enter one of the following commands:

   ```
   # cat /etc/oracle-release
   # cat /etc/redhat-release
   # cat /etc/SuSE-release
   # lsb_release -id
   ```

2. To determine whether the required kernel errata is installed, enter the following command:

   ```
   # uname -r
   ```

   The following is sample output displayed by running this command on an Oracle Linux 6 system:

   ```
   2.6.39-100.7.1.el6uek.x86_64
   ```

   Review the required errata level for your distribution. If the errata level is previous to the required minimum errata update, then obtain and install the latest kernel update from your Linux distributor.

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

   ```
   # rpm -q package_name
   ```

   Alternatively, if you require specific system architecture information, then enter the following command:

   ```
   # rpm -qa --queryformat "%{NAME}-%{VERSION}-%{RELEASE} (%{ARCH})\n" | grep package_name
   ```

   You can also combine a query for multiple packages, and review the output for the correct versions. For example:

   ```
   # rpm -q binutils compat-libstdc++ gcc glibc libaio libgcc libstdc++ \
   make sysstat unixodbc
   ```

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

Installing the cvuqdisk RPM for Linux

If you do not use an Oracle Preinstallation RPM, then you must install the cvuqdisk RPM. Without cvuqdisk, the Cluster Verification Utility cannot find shared disks, and

you receive a "Package cvuqdisk not installed" error when you run the Cluster Verification Utility. Use the `cvuqdisk` RPM for your hardware (for example, `x86_64`, or `i386`).

To install the `cvuqdisk` RPM, complete the following procedure:

1. Locate the `cvuqdisk` RPM package, which is in the directory `rpm` on the Oracle Database installation media. If you installed Oracle Grid Infrastructure, then it is in the directory *oracle_home1*/cv/rpm.

2. Log in as `root`.

3. Use the following command to find if you have an existing version of the `cvuqdisk` package:

   ```
   # rpm -qi cvuqdisk
   ```

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

   ```
   # rpm -e cvuqdisk
   ```

4. Set the environment variable `CVUQDISK_GRP` to point to the group that owns `cvuqdisk`, typically `oinstall`, for example:

   ```
   # CVUQDISK_GRP=oinstall; export CVUQDISK_GRP
   ```

5. In the directory where you have saved the `cvuqdisk` RPM, use the following command to install the `cvuqdisk` package:

   ```
   rpm -iv package
   ```

   For example:

   ```
   # rpm -iv cvuqdisk-1.0.9-1.rpm
   ```

Checking Shared Memory File System Mount on Linux

Ensure that the `/dev/shm` mount area is of type `tmpfs` and is mounted with the following options:

- With `rw` and `execute` permissions set on it

- With `noexec` or `nosuid` not set on it

Use the following procedure to check the shared memory file system:

1. Check the current mount settings. For example:

   ```
   $ more /etc/fstab |grep "tmpfs"
   tmpfs           /dev/shm        tmpfs     defaults    0 0
   ```

2. If necessary, change the mount settings. For example, log in as `root`, open the `/etc/fstab` file with a text editor, and modify the `tmpfs` line:

   ```
   tmpfs     /dev/shm     tmpfs     rw,exec     0 0
   ```

   > **See Also:**   *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems* for more information about shared memory mounts

Confirming Host Name Resolution

Typically, the computer on which you want to install Oracle Database is connected to a network. Ensure that the computer host name is resolvable through a Domain Name System (DNS), a network information service (NIS), or a centrally-maintained TCP/IP host file, such as /etc/hosts. Use the ping command to ensure that your computer host name is resolvable. For example:

```
ping myhostname
pinging myhostname.example.com [192.0.2.2] with 32 bytes of data:
Reply from 192.0.2.2: bytes=32 time=138ms TTL=56
```

If your computer host name does not resolve, then contact your system administrator.

> **See Also:** [Appendix E, "Configuring Networks for Oracle Database"](#)

Disabling Transparent HugePages

Transparent HugePages memory is enabled by default with Red Hat Enterprise Linux 6, Red Hat Enterprise Linux 7, SUSE 11, Oracle Linux 6, and Oracle Linux 7 with earlier releases of Oracle Linux Unbreakable Enterprise Kernel 2 (UEK2) kernels. Transparent HugePages memory is disabled by default in UEK2 and later releases of UEK kernels.

Transparent HugePages can cause memory allocation delays at runtime. To avoid performance issues, Oracle recommends that you disable Transparent HugePages on all Oracle Database servers. Oracle recommends that you instead use standard HugePages for enhanced performance.

Transparent HugePages memory differs from standard HugePages memory because the kernel khugepaged thread allocates memory dynamically during runtime. Standard HugePages memory is pre-allocated at startup, and does not change during runtime.

> **See Also:** *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems* for information about HugePages

To check if Transparent HugePages is enabled run one of the following commands as the root user:

Red Hat Enterprise Linux kernels:

```
# cat /sys/kernel/mm/redhat_transparent_hugepage/enabled
```

Other kernels:

```
# cat /sys/kernel/mm/transparent_hugepage/enabled
```

The following is a sample output that shows Transparent HugePages is being used as the [always] flag is enabled.

```
[always] never
```

> **Note:** If Transparent HugePages is removed from the kernel then the /sys/kernel/mm/transparent_hugepage or /sys/kernel/mm/redhat_transparent_hugepage files do not exist.

To disable Transparent HugePages perform the following steps:

1. On Oracle Linux 6 and earlier releases, add the following entry to the kernel boot line in the `/etc/grub.conf` file:

```
transparent_hugepage=never
```

For example:

```
title Oracle Linux Server (2.6.32-300.25.1.el6uek.x86_64)
        root (hd0,0)
        kernel /vmlinuz-2.6.32-300.25.1.el6uek.x86_64 ro root=LABEL=/
transparent_hugepage=never
        initrd /initramfs-2.6.32-300.25.1.el6uek.x86_64.img
```

The grub file name may vary for Oracle Linux 7 and later operating systems. Check your operating system documentation for the correct file name and the steps to disable Transparent HugePages.

For example, on Oracle Linux 7.3, to disable Transparent HugePages edit the `/etc/default/grub` file and then run the `grub2-mkconfig` command.

2. Restart the system to make the changes permanent.

Identifying Required Software Directories

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

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

Note:

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

- Ensure that all paths used by the database software, such as the Oracle home path and the Oracle base path, use characters only from the following set: "# % & ' () * + , - . / : ; < = > ? @ _ A-Z a-z 0-9. This includes user names, file names, and directory names. At the time of this release, the use of other characters for an Oracle Grid Infrastructure home or Oracle Database home is not supported. The set of characters provided is further restricted by user and file naming rules of the operating system.

Oracle Base Directory

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

/mount_point/app/*software_owner*

In this example:

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

  The examples in this guide use /u01 for the mount point directory.

- *software_owner* is the operating system user name of the software owner installing the Oracle software, for example oracle or grid.

  > **Note:** If you start a database instance using the server parameter file (spfile) with the ORACLE_BASE environment variable set, then its value is automatically stored in spfile. If you unset the ORACLE_BASE environment variable and start the instance again, then the database uses the value of the Oracle base directory stored in spfile.

You must specify the Oracle base directory that contains all Oracle products.

> **Note:** If you have an existing Oracle base directory, then you can select it from the Oracle Base list during the database installation. If you do not have an Oracle base, then you can create one by editing the text in the list box. By default, the list contains the existing value for the Oracle base. See "Installing the Oracle Database Software" on page 7-8 for more information.

You can use the same Oracle base directory for multiple installations or you can create separate Oracle base directories for different installations. If different operating system users install Oracle software on the same system, then each user must create a separate Oracle base directory. The following are examples of Oracle base directories that can exist on the same system:

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

See "Creating the Oracle Base and Oracle Inventory Directories" on page 4-38.

Oracle Inventory Directory

The Oracle Inventory directory (oraInventory) stores an inventory of all software installed on the system. It is required and shared by all Oracle software installations on a single system. If you have an existing Oracle Inventory path, then Oracle Universal Installer continues to use that Oracle Inventory.

The first time you install Oracle software on a system, Oracle Universal Installer provides an Optimal Flexible Architecture-compliant path in the format /u[01-09]/app, such as /u01/app. The user running the installation has permissions to write to that path. If this is true, then Oracle Universal Installer creates the Oracle Inventory directory in the path /u[01-09]/app/oraInventory. For example:

```
/u01/app/oraInventory
```

If you have set ORACLE_BASE for the oracle user during installation, then Oracle Universal Installer creates the Oracle Inventory directory one level above the ORACLE_BASE in the path ORACLE_BASE/../oraInventory. For example, if ORACLE_BASE is set to /u01/app/oracle, then the Oracle Inventory directory is created in the path /u01/app/oraInventory.

If you have neither created an OFA-compliant path nor set `ORACLE_BASE`, then the Oracle Inventory directory is placed in the home directory of the user that is performing the installation. For example:

```
/home/oracle/oraInventory
```

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

> **Note:**
>
> - All Oracle software installations rely on the Oracle Inventory directory. Ensure that you back it up regularly.
>
> - Do not delete this directory unless you have completely removed all Oracle software from the system.
>
> - By default, the Oracle Inventory directory is not installed under the Oracle Base directory. This is because all Oracle software installations share a common Oracle Inventory, so there is only one Oracle Inventory for all users. Whereas, there is a separate Oracle Base for each user.

Oracle Home Directory

The Oracle home directory is the directory where you 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 and a name that identifies it. In accordance with the OFA guidelines, Oracle strongly recommends that the Oracle home directory you specify is a subdirectory of the Oracle base directory for the user account performing the installation. Oracle recommends that you specify a path similar to the following for the Oracle home directory:

```
oracle_base/product/11.2.0/db_home_1
oracle_base/product/12.1.0/dbhome_1
oracle_base/product/12.1.0/grid
```

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

> **Note:** During the installation, you must not specify an existing directory that has predefined permissions applied to it as the Oracle home directory. If you do, then you may experience installation failure due to file and group ownership permission errors.

Identifying or Creating an Oracle Base Directory

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

- Identifying an Existing Oracle Base Directory
- Creating the Oracle Base and Oracle Inventory Directories

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

Identifying an Existing Oracle Base Directory

Existing Oracle base directories may not have paths that follow the Optimal Flexible Architecture (OFA) guidelines. However, if you identify an existing Oracle Inventory directory or existing Oracle home directories, then you can usually identify the Oracle base directories, as follows:

- Identifying an existing Oracle Inventory directory. See "Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist" on page 5-2 for more information.

  > **Note:** Oracle recommends that you do not put the `oraInventory` directory under the Oracle base directory for a new installation. If you have an existing installation, then follow the steps in this section.

- Identifying an existing Oracle home directory

  Enter the following command to display the contents of the `oratab` file:

  On Linux x86-64 and IBM: Linux on System z:

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

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

  ```
  *:/u03/app/oracle/product/12.1.0/dbhome_1:N
  *:/opt/orauser/infra_904:N
  *:/oracle/9.2.0:N
  ```

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

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

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

- Identifying an existing Oracle base directory

  After you locate the Oracle home directory, run a similar command to confirm the location of Oracle base:

  ```
  cat /u01/app/oraInventory/ContentsXML/inventory.xml
  ```

Before deciding to use an existing Oracle base directory for this installation, ensure that it meets the following conditions:

- It is not on the same file system as the operating system.

■ It has sufficient free disk space, as follows:

Requirement	Free Disk Space
The Oracle base directory contains only software files.	Up to 4 GB
The Oracle base directory contains both software and database files (not recommended for production databases).	Up to 6 GB

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

```
# df -h oracle_base_path
```

See the following sections for more information:

■ If an Oracle base directory exists and you want to use it, then see "Choosing a Storage Option for Oracle Database and Recovery Files" section on page 4-39.

When you configure the `oracle` user's environment later in this chapter, set the `ORACLE_BASE` environment variable to specify the directory you chose.

■ If an Oracle base directory does not exist on the system or to create an Oracle base directory, see "Creating the Oracle Base and Oracle Inventory Directories" on page 4-38.

Creating the Oracle Base and Oracle Inventory Directories

Before you create an Oracle base directory, you must identify an appropriate file system with sufficient free disk space.

To identify an appropriate file system, perform the following:

1. Determine the free disk space on each mounted file system, using the following command:

```
# df -h
```

2. Identify a file system that has the appropriate amount of free space, from the display:

The file system that you identify can be a local file system, a cluster file system, or an NFS file system on a certified NAS device.

3. Note the name of the mount point directory for the file system that you identified.

To create the Oracle base directory and specify the correct owner, group, and permissions for it, perform the following:

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

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

For example:

```
# mkdir -p /u01/app/oracle
```

```
# mkdir -p /u01/app/oraInventory
# chown -R oracle:oinstall /u01/app/oracle
# chown -R oracle:oinstall /u01/app/oraInventory
# chmod -R 775 /u01/app
```

2. When you configure the `oracle` user's environment later in this chapter, set the `ORACLE_BASE` environment variable to specify the Oracle base directory that you created.

Setting Disk I/O Scheduler on Linux

Disk I/O schedulers reorder, delay, or merge requests for disk I/O to achieve better throughput and lower latency. Linux has multiple disk I/O schedulers available, including Deadline, Noop, Anticipatory, and Completely Fair Queuing (CFQ). For best performance for Oracle ASM, Oracle recommends that you use the Deadline I/O Scheduler.

Enter the following command to ensure that the Deadline disk I/O scheduler is configured for use:

```
# echo deadline > /sys/block/${ASM_DISK}/queue/scheduler
```

Choosing a Storage Option for Oracle Database and Recovery Files

Oracle Database files include data files, control files, redo log files, the server parameter file, and the password file. For all installations, you must choose the storage option to use for Oracle Database files. During the database installation, you must choose the storage option to use for recovery files (the fast recovery area). You do not have to use the same storage option for each file type.

> **Note:**   Database files and recovery files are supported on file systems and Oracle ASM.

Use the following guidelines when choosing the storage options for each file type:

- Choose any combination of the supported storage options for each file type.

- Determine whether you want to use Oracle ASM for Oracle Database files, recovery files, or both. See "Identifying Storage Requirements for Oracle Automatic Storage Management" on page 6-8 for more information.

- See "Database Storage Options" on page 2-9 for more information about these storage options

For information on how to configure disk storage before you start the installation, see one of the following sections depending on your choice:

- To use a file system for database or recovery file storage, see the "Creating Directories for Oracle Database or Recovery Files" section on page 4-40.

- To use Oracle ASM for database or recovery file storage, see the "Configuring Storage for Oracle Automatic Storage Management" section.

- To identify existing disk groups and determine the free disk space that they contain, see the "Using an Existing Oracle Automatic Storage Management Disk Group" section on page 7-3.

Creating Directories for Oracle Database or Recovery Files

This section contains the following topics:

- Guidelines for Placing Oracle Database Files on a File System
- Creating Required Directories

Guidelines for Placing Oracle Database Files on a File System

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

- The default path suggested by Oracle Universal Installer for the database file directory is a subdirectory of the Oracle base directory.

- You can choose either a single file system or more than one file system to store the database files:

  - If you want to use a single file system, then choose a file system on a physical device that is 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.

  - If you want to use more than one file system, then choose file systems on separate physical devices that are dedicated to the database.

    This method enables you to distribute physical input-output operations and create separate control files on different devices for increased reliability. It also enables you to fully implement the OFA guidelines. You can choose the Advanced database creation option to implement this method.

- If you intend to create a preconfigured database during the installation, then the file system (or file systems) that you choose must have at least 2 GB of free disk space.

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

- For optimum performance, the file systems that you choose must be on physical devices that are used only by the database.

- 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 to place the Oracle Database or recovery files on a separate file system from the Oracle base directory.

To create directories for the Oracle database or recovery files on separate file systems from the Oracle base directory:

1. Use the following command to determine the free disk space on each mounted file system:

   ```
   # df -h
   ```

2. Identify the file systems to use, from the display:

File Type	File System Requirements
Database files	Choose either: ■ A single file system with at least 2 GB of free disk space ■ Two or more file systems with at least 2 GB of free disk space in total
Recovery files	Choose a file system with at least 2.4 GB of free disk space

If you are using the same file system for many file types, then add the disk space requirements for each type to determine the total disk space requirement.

3. Note the names of the mount point directories for the file systems that you identified.

4. Enter commands similar to the following to create the recommended subdirectories in each of the mount point directories and set the appropriate owner, group, and permissions on them:

■ Database file directory:

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

The default location for the database file directory is `$ORACLE_BASE/oradata`.

■ Recovery file directory (fast recovery area):

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

The default fast recovery area is `$ORACLE_BASE/fast_recovery_area`. Oracle recommends that you keep the fast recovery area on a separate physical disk than that of the database file directory. This method enables you to use the fast recovery area to retrieve data if the disk containing `oradata` is unusable for any reason.

5. See "Configuring Servers for Oracle Grid Infrastructure for a Standalone Server" and "Stopping Existing Oracle Processes" on page 5-12 for information about using Oracle ASM for storage.

5

Configuring Users, Groups and Environments for Oracle Database

This chapter describes the users and groups user environment and management environment settings to complete before you install Oracle Database and Grid Infrastructure for a standalone server. It contains the following topics:

- Creating Required Operating System Groups and Users
- Checking Resource Limits for Oracle Software Installation Users
- Setting Remote Display and X11 Forwarding Configuration
- Stopping Existing Oracle Processes
- Configuring Oracle Software Owner Environment
- Determining Root Script Execution Plan

Creating Required Operating System Groups and Users

Depending on if this is the first time Oracle software is being installed on your system and on the products that you are installing, you may have to create several operating system groups and users.

However, if you use Oracle Preinstallation RPM to provision your Linux operating system for an Oracle Grid Infrastructure or Oracle Database installation, then it configures an Oracle database installation owner (`oracle`), an Oracle Inventory group (`oinstall`), and an Oracle administrative privileges group (`dba`).

> **See Also:** Chapter 3, "Automatically Configuring Oracle Linux with Oracle Preinstallation RPM"

If you prefer to allocate operating system user privileges so that you can use one administrative user and one group for operating system authentication for all administrative privileges, then you can use the `oracle` user as the installation owner, and use one group as the primary group for any user requiring administrative privileges for Oracle ASM, and Oracle Database administration. This group must also be the Oracle Inventory group. To simplify using the defaults for Oracle tools the group name should be `oinstall`.

You can also create custom configuration groups and users based on job role separation that divide access privileges.

Log in as `root`, and use the instructions in the following sections to locate or create the Oracle Inventory group and a Oracle software owner user:

- [Determining If the Oracle Inventory and Oracle Inventory Group Exists](#)
- [Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist](#)
- [About Oracle Installations with Job Role Separation](#)
- [Descriptions of Job Role Separation Groups and Users](#)
- [Creating Job Role Separation Database Operating System Groups and Users](#)

Determining If the Oracle Inventory and Oracle Inventory Group Exists

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

You can configure one group to be the access control group for Oracle Inventory, for database administrators (OSDBA), and for all other access control groups used by Oracle software for operating system authentication. However, if you use one group to provide operating system authentication for all system privileges, then this group must be the primary group for all users to whom you want to grant administrative system privileges.

If you have an existing central Oracle Inventory, then ensure that you use the same Oracle Inventory for all Oracle software installations, and ensure that all Oracle software users you intend to use for installation have permissions to write to this directory.

To determine if the Oracle Inventory group exists, perform the following steps:

1. An `oraInst.loc` file has content similar to the following:

   ```
   inventory_loc=central_inventory_location
   inst_group=group
   ```

   In the preceding example, *central_inventory_location* is the location of the Oracle Central Inventory, and *group* is the name of the group that has permissions to write to the central inventory.

   To determine if the `oraInst.loc` file exists, enter the following command:

   On Linux x86-64 and IBM: Linux on System z:

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

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

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

2. Use the command `grep groupname /etc/group` to confirm that the group specified as the Oracle Inventory group still exists on the system. For example:

   ```
   # grep oinstall /etc/group
   oinstall:x:1000:grid,oracle
   ```

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

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

```
# /usr/sbin/groupadd -g 54321 oinstall
```

About Oracle Installations with Job Role Separation

A job role separation configuration of Oracle Database and Oracle ASM is a configuration with groups and users to provide separate groups for operating system authentication.

Review the following restrictions for users created to own Oracle software:

- Oracle recommends that you create one software owner to own each Oracle software installation. See "Oracle Software Owner For Each Oracle Software Product" on page 5-3 for more information.

- To create separate Oracle software owners and separate operating system privileges groups for different Oracle software installations, note that each of these users must have the Oracle central inventory group (`oraInventory`) as their primary group. Members of this group have write privileges to the Oracle central inventory (`oraInventory`) directory, and are also granted permissions for various Oracle Restart resources and directories in the Oracle Restart home to which DBAs need write access, and other necessary privileges. In Oracle documentation, this group is represented as `oinstall` in code examples. See "Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist" on page 5-2.

- Oracle software installation owner users must also have the OSDBA group of the database, the OSDBA group of the Oracle Grid Infrastructure home (if you create it), and (if you create them) the OSOPER, OSBACKUPDBA, OSDGDBA, and OSKMDBA groups as secondary groups. Oracle software owners require membership to the OSDBA group of the Oracle Grid infrastructure home so that database instances can log on to Oracle ASM.

- The Oracle Database, and the Oracle Grid Infrastructure for a standalone server installation owner users (`oracle` and `grid` respectively) must belong to the Oracle Inventory group (`oinstall`).

- Each Oracle software owner must be a member of the same central inventory group. Oracle recommends that you do not have more than one central inventory for Oracle installations. If an Oracle software owner has a different central inventory group, then you may corrupt the central inventory.

- The Oracle Grid Infrastructure for a standalone server installation owner user (`grid`) must be in the OSDBA group of every database home.

Descriptions of Job Role Separation Groups and Users

The following sections provide an overview about users and groups to divide access privileges by job roles:

- Oracle Software Owner For Each Oracle Software Product
- Standard Oracle Database Groups for Job Role Separation
- Extended Oracle Database Groups for Job Role Separation
- Oracle Automatic Storage Management Groups for Job Role Separation

Oracle Software Owner For Each Oracle Software Product

You can create a single user (for example, `oracle`) to own both Oracle Database, and Oracle Grid Infrastructure for a standalone server installations. However, Oracle recommends that you create one software owner to own each Oracle software installation (typically, `oracle`, for the database software and `grid` for the Oracle Restart owner user).

You must create at least one software owner the first time you install Oracle software on the system.

> **Note:**   In Oracle documentation, a user created to own only Oracle Grid Infrastructure software installations is called the `grid` user. A user created to own either all Oracle installations, or only Oracle database installations, is called the `oracle` user.

Standard Oracle Database Groups for Job Role Separation

Create the following operating system groups, if you are installing Oracle Database:

- The OSDBA group (typically, `dba`)

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

- The OSOPER group for Oracle Database (typically, `oper`)

  This is an optional group. Create this group if you want a separate group of operating system users to have a limited set of database administrative privileges for starting up and shutting down the database (the `SYSOPER` privilege). This group cannot directly connect as `SYSOPER`, unless explicitly granted. However, they have the privileges granted by the `SYSOPER` privilege. By default, members of the OSDBA group have all privileges granted by the `SYSOPER` privilege.

Extended Oracle Database Groups for Job Role Separation

Starting with Oracle Database 12*c* release 1 (12.1), in addition to the OSOPER privilege to start and shut down the database, you can create new administrative privileges that are more task-specific and less privileged than the OSDBA/SYSDBA system privileges to support specific administrative privileges tasks required for everyday database operation. Users granted these system privileges are also authenticated through operating system group membership.

You do not have to create these specific group names, but during installation you are prompted to provide operating system groups whose members are granted access to these system privileges. You can assign the same group to provide authentication for these privileges, but Oracle recommends that you provide a unique group to designate each privilege.

The OSDBA subset job role separation privileges and groups consist of the following:

- The OSBACKUPDBA group for Oracle Database (typically, `backupdba`)

  Create this group if you want a separate group of operating system users to have a limited set of database backup and recovery related administrative privileges (the `SYSBACKUP` privilege).

  Add the Oracle software installation owner to the OSBACKUPDBA group.

- The OSDGDBA group for Oracle Data Guard (typically, `dgdba`)

  Create this group if you want a separate group of operating sytem users to have a limited set of privileges to administer and monitor Oracle Data Guard (the SYSDG privilege).

  Add the Oracle software installation owner to the OSDGDBA group

- The OSKMDBA group for encyption key management (typically, `kmdba`)

Create this group if you want a separate group of operating sytem users to have a limited set of privileges for encryption key management such as Oracle Wallet Manager management (the SYSKM privilege).

If you want to have an OSKMDBA group for Oracle Database, then the Oracle software installation owner must be a member of this group.

Oracle Automatic Storage Management Groups for Job Role Separation

Create the following operating system groups if you are installing Oracle Grid Infrastructure:

- The OSDBA group for Oracle ASM (typically, `asmdba`)

  The OSDBA group for Oracle ASM can be the same group used as the OSDBA group for the database, or you can create a separate OSDBA group for Oracle ASM to provide administrative access to Oracle ASM instances.

  The Oracle Grid Infrastructure software owner (typically, `grid`) must be a member of the OSDBA group. Membership in the OSDBA group enables access to the files managed by Oracle ASM. If you have a separate OSDBA group for Oracle ASM, then the Oracle Restart software owner must be a member of the OSDBA group for each database and the OSDBA group for Oracle ASM.

- The OSASM group for Oracle ASM Administration (typically, `asmadmin`)

  Create this group as a separate group if you want to have separate administration privileges groups for Oracle ASM and Oracle Database administrators. Members of this group are granted the SYSASM system privileges to administer Oracle ASM. In Oracle documentation, the operating system group whose members are granted `SYSASM` privileges is called the OSASM group, and in command lines, is referred to as `asmadmin`.

  Oracle ASM can support multiple databases. If you have multiple databases on your system, and use multiple OSDBA groups so that you can provide separate SYSDBA privileges for each database, then you should create a group whose members are granted the OSASM/SYSASM administrative privileges, and create a grid infrastructure user (`grid`) that does not own a database installation, so that you separate Oracle Grid Infrastructure SYSASM administrative privileges from a database administrative privileges group.

  Members of the OSASM group can use SQL to connect to an Oracle ASM instance as `SYSASM` using operating system authentication. The `SYSASM` privileges permit mounting and dismounting of disk groups, and other storage administration tasks. `SYSASM` privileges provide no access privileges on an RDBMS instance.

  If you do not designate a separate group as the OSASM group, then the OSDBA group you define is also, by default, the OSASM group.

- The OSOPER group for Oracle ASM (typically, `asmoper`)

  This is an optional group. Create this group if you want a separate group of operating system users to have a limited set of Oracle instance administrative privileges (the `SYSOPER` for ASM privilege), including starting up and stopping the Oracle ASM instance. By default, members of the OSASM group also have all privileges granted by the SYSOPER for ASM privilege.

  If you want to have an OSOPER group for Oracle ASM, then the Oracle Grid Infrastructure owner must be a member of this group.

See Also:

- *Oracle Database Administrator's Guide* for more information about the OSDBA, OSASM, OSOPER, OSBACKUPDBA, OSDGDBA, and OSKMDBA groups, and the `SYSDBA`, `SYSASM`, `SYSOPER`, `SYSBACKUP`, `SYSDG`, and `SYSKM` privileges

- The "Managing Administrative Privileges" section in *Oracle Database Security Guide*

Creating Job Role Separation Database Operating System Groups and Users

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

- Creating the OSDBA Group for Database Installations

- Creating an OSOPER Group for Database Installations

- Creating the OSBACKUPDBA Group for Database Installations

- Creating the OSDGDBA Group for Database Installations

- Creating the OSKMDBA Group for Database Installations

- Creating the OSDBA Group for Oracle Automatic Storage Management

- Creating the OSOPER Group for Oracle Automatic Storage Management

- Creating the OSASM Group for Oracle Automatic Storage Management

- When to Create the Oracle Software Owner User

- Determining if an Oracle Software Owner User Exists

- Creating an Oracle Software Owner User

- Modifying an Existing Oracle Software Owner User

Note:

- After you create the required operating system groups described in this section, you must add the Oracle software owner user (typically, `oracle`) to these groups, otherwise these groups will not be available as an option in Oracle Universal Installer while performing the database installation.

- The UIDs and GIDs mentioned in this section are illustrative only. Oracle recommends that you do not use the UID and GID defaults. Instead, provide common assigned group and user IDs, and confirm that they are unused before you create or modify groups and users.

- If necessary, contact your system administrator before using or modifying an existing user or group.

Creating the OSDBA Group for Database Installations

You must create an OSDBA group in the following circumstances:

- An OSDBA group does not exist, for example, if this is the first installation of Oracle Database software on the system

- An OSDBA group exists, but you want to give a different group of operating system users database administrative privileges for a new Oracle Database installation

Create the OSDBA group using the group name `dba`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54322 dba
```

Creating an OSOPER Group for Database Installations

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

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

Create the OSOPER group using the group name `oper`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54323 oper
```

Creating the OSBACKUPDBA Group for Database Installations

Create the OSBACKUPDBA group using the group name `backupdba`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54324 backupdba
```

Creating the OSDGDBA Group for Database Installations

Create the OSDGDBA group using the group name `dgdba`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54325 dgdba
```

Creating the OSKMDBA Group for Database Installations

Create the OSKMDBA group using the groups name `kmdba` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54326 kmdba
```

Creating the OSDBA Group for Oracle Automatic Storage Management

If you require, create a new OSDBA group for Oracle ASM using the group name `asmdba` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54327 asmdba
```

Creating the OSOPER Group for Oracle Automatic Storage Management

If you require, create an OSOPER group for Oracle ASM with the group name `asmoper` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54328 asmoper
```

Creating the OSASM Group for Oracle Automatic Storage Management

If you require, create an OSASM group using the group name `asmadmin` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54329 asmadmin
```

When to Create the Oracle Software Owner User

Depending on whether you want to create a new user, or use an existing user, see the following sections:

- If an Oracle software owner user does not exist; for example, if this is the first installation of Oracle software on the system.

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

- If you have created an Oracle software owner for Oracle Grid Infrastructure, such as `grid`, and you want to create a separate Oracle software owner for Oracle Database software, such as `oracle`.

Determining if an Oracle Software Owner User Exists

To determine if an Oracle software owner user named `oracle` or `grid` exists, enter commands similar to the following:

```
# id oracle
# id grid
```

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

```
uid=54321(oracle) gid=54321(oinstall) groups=54322(dba),54323(oper)
```

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

```
uid=54322(grid) gid=54321(oinstall)
groups=54321(oinstall),54329(asmadmin),54327(asmdba),54322(dba)
```

Determine whether you want to use an existing user, or create a new user. To use the existing user, ensure that the user's primary group is the Oracle Inventory group and that it is a member of the appropriate OSDBA and OSOPER groups. Depending on your choice, see one of the following sections for more information:

- Creating an Oracle Software Owner User

- Modifying an Existing Oracle Software Owner User

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

Creating an Oracle Software Owner User

If the Oracle software owner user (`oracle` or `grid`) does not exist, or if you require a new Oracle software owner user, then create it as described in this section (in this case to create the `oracle` user).

In the following procedure, use the user name `oracle` unless a user with that name exists:

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

   ```
   # /usr/sbin/useradd -u 54321 -g oinstall -G dba,asmdba,backupdba,dgdba,kmdba
   oracle
   ```

   In the preceding command:

   - The -u option specifies the user ID. Using this command flag is optional because the system can provide you with an automatically generated user ID number. However, Oracle recommends that you specify a number. You must note the user ID number because you need it during preinstallation.

   - The `-g` option specifies the primary group, which must be the Oracle Inventory group, for example `oinstall`.

   - The `-G` option specifies the secondary groups, which must include the OSDBA group, and, if required, the ASMDBA, OSOPER, OSBACKUPDBA, OSDGDBA, and OSKMDBA groups, for example, `dba`, `asmdba`, `oper`, `backupdba`, `dgdba`, and `kmdba`.

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

   ```
   # passwd oracle
   ```

Modifying an Existing Oracle Software Owner User

If the `oracle` user exists, but its primary group is not `oinstall`, or it is not a member of the appropriate OSDBA, OSOPER, or OSDBA for ASM groups, then modify the user group settings for the user `oracle`.

Specify the primary group using the `-g` option and any required secondary group using the `-G` option:

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

Oracle does not support modifying an existing installation owner. See "About Oracle Installations with Job Role Separation" on page 5-3 for a complete list of restrictions.

Checking Resource Limits for Oracle Software Installation Users

On Oracle Linux systems, Oracle recommends that you install Oracle Preinstallation RPMs to meet preinstallation requirements like configuring your operating system to set the resource limits in the `limits.conf` file. Oracle Preinstallation RPM only configures the `limits.conf` file for the `oracle` user. If you are implementing Oracle Grid Infrastructure job role separation, then copy the values from the `oracle` user to the `grid` user in the `limits.conf` file.

For each installation software owner, check the resource limits for installation, using the following recommended ranges:

Table 5–1   Installation Owner Resource Limit Recommended Ranges

Resource Shell Limit	Resource	Soft Limit	Hard Limit
Open file descriptors	nofile	at least 1024	at least 65536
Number of processes available to a single user	nproc	at least 2047	at least 16384
Size of the stack segment of the process	stack	at least 10240 KB	at least 10240 KB, and at most 32768 KB
Maximum Locked Memory Limit	memlock	at least 90 percent of the current RAM when HugePages memory is enabled and at least 3145728 KB (3 GB) when HugePages memory is disabled	at least 90 percent of the current RAM when HugePages memory is enabled and at least 3145728 KB (3 GB) when HugePages memory is disabled

To check resource limits:

1. Log in as an installation owner.

2. Check the soft and hard limits for the file descriptor setting. Ensure that the result is in the recommended range, for example:

```
$ ulimit -Sn
1024
$ ulimit -Hn
65536
```

3. Check the soft and hard limits for the number of processes available to a user. Ensure that the result is in the recommended range, for example:

```
$ ulimit -Su
2047
$ ulimit -Hu
16384
```

4. Check the soft limit for the stack setting. Ensure that the result is in the recommended range, for example:

```
$ ulimit -Ss
10240
$ ulimit -Hs
32768
```

5. Repeat this procedure for each Oracle software installation owner.

If necessary, update the resource limits in the `/etc/security/limits.conf` configuration file for the installation owner. However, note that the configuration file is distribution specific. Contact your system administrator for distribution specific configuration file information.

> **Note:** If the `grid` or `oracle` users are logged in, then changes in the
> `limits.conf` file do not take effect until you log these users out and
> log them back in. You must do this before you use these accounts for
> installation.

> **See Also:** "Configuring Oracle Software Owner Environment" on
> page 5-13

Setting Remote Display and X11 Forwarding Configuration

If you are on a remote terminal, and the local system has only one visual (which is
typical), then use the following syntax to set your user account `DISPLAY` environment
variable:

Bourne, Korn, and Bash shells:

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

C shell:

```
$ setenv DISPLAY hostname:0
```

For example, if you are using the Bash shell and if your host name is `local_host`, then
enter the following command:

```
$ export DISPLAY=local_host:0
```

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

1. Using any text editor, edit or create the software installation owner's
    `~/.ssh/config` file.

2. Ensure that the `ForwardX11` attribute in the `~/.ssh/config` file is set to `no`. For
    example:

    ```
    Host *
        ForwardX11 no
    ```

3. Ensure that the permissions on the `~/.ssh` are secured to the `oracle` or `grid` user.
    For example:

    ```
    $ ls -al .ssh
    total 28
    drwx------  2 oracle oinstall 4096 Jun 21 2012
    drwx------ 19 oracle oinstall 4096 Jun 21 2012
    -rw-r--r--  1 oracle oinstall 1202 Jun 21 2012 authorized_keys
    -rwx------  1 oracle oinstall  668 Jun 21 2012 id_dsa
    -rwx------  1 oracle oinstall  601 Jun 21 2012 id_dsa.pub
    -rwx------  1 oracle oinstall 1610 Jun 21 2012 known_hosts
    ```

Stopping Existing Oracle Processes

> **Note:** If you are installing additional Oracle Database 12*c* products in an existing Oracle home, then stop *all* processes, including the listener and database, running in the Oracle home. You must complete this task to enable Oracle Universal Installer to relink certain executables and libraries.

Consider the following before you install Oracle Grid Infrastructure for a standalone server, or Oracle Database:

- If you plan to use Oracle Restart, then you must install Oracle Grid Infrastructure for a standalone server before you install and create the database. When you perform a database installation, the database must use the same listener created during the Oracle Grid Infrastructure for a standalone server installation, thereafter you do not have to perform the steps listed in this section.

   The default listener and any additional listeners must run from the Oracle Grid Infrastructure home. See "Configuring Oracle Software Owner Environment" on page 5-13 to continue.

- If you have an existing Oracle Database 12*c* running on Oracle ASM, then stop any existing Oracle ASM instances. After you finish installing Oracle Grid Infrastructure for a standalone server, start the Oracle ASM instance again.

If you create a database during the software installation, then most installation types configure and start a default Oracle Net listener using TCP/IP port `1521` and the IPC key value `EXTPROC`. If an existing Oracle Net listener process is using the same port or key value, Oracle Universal Installer looks for the next available port (for example, `1522`) and configures and starts the new listener on this available port.

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

1. Switch user to `oracle`:

   ```
   # su - oracle
   ```

2. Enter the following command to determine if a listener process is running and to identify its name and the Oracle home directory in which it is installed:

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

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

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

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

   > **Note:** If no Oracle Net listeners are running, then see "Configuring Oracle Software Owner Environment" on page 5-13 to continue.

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

   - Bourne, Bash, or Korn shell:

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

- ■ C or tcsh shell:

```
% setenv ORACLE_HOME oracle_home1
```

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

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

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

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

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

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

Configuring Oracle Software Owner Environment

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

- ■ Set the default file mode creation mask (`umask`) to 022 in the shell startup file.
- ■ Set the `DISPLAY` environment variable.

> **Caution:** Use shell programs supported by your operating system vendor. If you use a shell program that is not supported by your operating system, then you can encounter errors during installation.

To set the Oracle software owners' environments, follow these steps, for each software owner (`oracle`, `grid`). The following procedure lists the steps for the `oracle` user only:

1. Start a new X terminal session (`xterm`).

2. Enter the following command to ensure that X Window applications can display on this system:

```
$ xhost + RemoteHost
```

where `RemoteHost` is the fully qualified remote host name. For example:

```
$ xhost + somehost.example.com
```

3. If you are not logged in as the user, then switch to the software owner user you are configuring. For example, as the `oracle` user.

```
$ su - oracle
```

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

```
$ echo $SHELL
```

5. Open the user's shell startup file in any text editor:

- Bash shell (`bash`):

  ```
  $ vi .bash_profile
  ```

- Bourne shell (`sh`) or Korn shell (`ksh`):

  ```
  $ vi .profile
  ```

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

  ```
  % vi .login
  ```

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

   ```
   umask 022
   ```

7. Save the file and exit from the text editor.

8. To run the shell startup script, enter one of the following commands:

   - Bash shell:

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

   - Bourne or Korn shell:

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

   - C shell:

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

9. If you are not installing the software on the local computer, then run the following command on the remote computer to set the `DISPLAY` variable:

   - Bourne, Bash or Korn shell:

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

   - C shell:

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

   In this example, `local_host` is the host name or IP address of the system (your workstation, or another client) on which you want to display the installer.

   Run the following command on the remote system to check if the `SHELL` and the `DISPLAY` environment variables are set correctly:

   ```
   echo $SHELL
   echo $DISPLAY
   ```

   To change the display location from the default display to a remote system display, run the following command on the local computer:

   ```
   $ xhost + RemoteHost
   ```

   To verify that the X applications display is set properly, run an X11-based program that comes with the operating system such as `xclock`.

   ```
   $ xclock
   ```

   In this example, you can find `xclock` at `/usr/X11R6/bin/xclocks`.

If the `DISPLAY` environment variable is set correctly, then you can see `xclock` on your computer screen. If you get any display errors see "X Window Display Errors" on page I-2. If `xclock` does not start, then contact your system administrator.

10. If the `/tmp` directory has less than 1 GB of free disk space, then identify a file system with at least 1 GB of free space and set the `TMP` and `TMPDIR` environment variables to specify a temporary directory on this file system:

    a. To determine the free disk space on each mounted file system use the following command:

    ```
    # df -h /tmp
    ```

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

    ```
    $ sudo - s
    # mkdir /mount_point/tmp
    # chmod 775 /mount_point/tmp
    # exit
    ```

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

    *   Bourne, Bash, or Korn shell:

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

    *   C shell:

        ```
        % setenv TMP /mount_point/tmp
        % setenv TMPDIR /mount_point/tmp
        ```

11. If you have had an existing installation on your system, and you are using the same user account to install this installation, then unset the `ORACLE_HOME`, `ORACLE_BASE`, `ORACLE_SID`, `TNS_ADMIN` environment variables and any other environment variable set for the Oracle installation user that is connected with Oracle software homes.

    Enter the following commands to ensure that the `ORACLE_HOME`, `ORACLE_BASE`, `ORACLE_SID` and `TNS_ADMIN` environment variables are not set:

    ■   Bourne, Bash, or Korn shell:

        ```
        $ unset ORACLE_HOME
        $ unset ORACLE_BASE
        $ unset ORACLE_SID
        $ unset TNS_ADMIN
        ```

    ■   C shell:

        ```
        % unsetenv ORACLE_HOME
        % unsetenv ORACLE_BASE
        % unsetenv ORACLE_SID
        % unsetenv TNS_ADMIN
        ```

    Use the following command to check the `PATH` environment variable:

    ```
    $ echo $PATH
    ```

Ensure that the `$ORACLE_HOME/bin` path is removed from your `PATH` environment variable.

> **Note:** If the `ORACLE_HOME` environment variable is set, then Oracle Universal Installer uses the value that it specifies as the default path for the Oracle home directory. If you set the `ORACLE_BASE` environment variable, then Oracle recommends that you unset the `ORACLE_HOME` environment variable and choose the default path suggested by Oracle Universal Installer.

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

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

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

> **See Also:** "Environment Requirements for Oracle Grid Infrastructure Software Owner" on page 6-3

Determining Root Script Execution Plan

During an Oracle Grid Infrastructure installation, Oracle Universal Installer prompts you to run scripts with superuser (or `root`) privileges to complete several system configuration tasks. You can either run these root scripts manually as `root` when prompted, or during installation you can provide configuration information and passwords using one of the following root privilege delegation options:

- **Use root user credentials**

  Provide the superuser (or `root`) password. This option runs the root scripts automatically as the `root` user.

- **Use Sudo**

  Sudo is a UNIX and Linux utility that allows members of the sudoers group privileges to run individual commands as `root`. To enable Sudo, have a system administrator with the appropriate privileges configure a user that is a member of the sudoers list, and provide the username and password when prompted during installation.

  > **See Also:** Step 9, "Root Script Execution Configuration" screen in the "Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation" section.

6

Oracle Grid Infrastructure for a Standalone Server

Oracle Grid Infrastructure for a standalone server, also known as Oracle Restart, provides system support for a single-instance Oracle Database. This support includes volume management, file system, and automatic restart capabilities. If you plan to use Oracle Automatic Storage Management (Oracle ASM), then you must install Oracle Restart before installing your database. Oracle Grid Infrastructure for a standalone server includes Oracle Restart and Oracle Automatic Storage Management. Oracle combined the two infrastructure products into a single set of binaries that is installed into an Oracle Restart home.

Oracle Automatic Storage Management is a volume manager and a file system for Oracle database files that supports single-instance Oracle Database and Oracle Real Application Clusters (Oracle RAC) configurations. Oracle Automatic Storage Management also supports a general purpose file system for your application needs, including Oracle Database binaries. Oracle Automatic Storage Management is Oracle's recommended storage management solution that provides an alternative to conventional volume managers, file systems, and raw devices.

> **Note:**
>
> - You can neither install Oracle Restart on an Oracle Grid Infrastructure cluster member node, nor add an Oracle Restart server to an Oracle Grid Infrastructure cluster member node. Oracle Restart supports single-instance databases on one server, while Oracle Grid Infrastructure for a Cluster supports single-instance or Oracle RAC databases on a cluster.
>
> - If you want to use Oracle ASM or Oracle Restart, then you must install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart.
>
> - Oracle Restart is used in single-instance (nonclustered) environments only.

Oracle Restart improves the availability of your Oracle database by providing the following:

- When there is a hardware or a software failure, Oracle Restart automatically starts all Oracle components, including the Oracle database instance, Oracle Net Listener, database services, and Oracle ASM.

- Oracle Restart starts components in the proper order when the database host is restarted.

- Oracle Restart runs periodic checks to monitor the status of Oracle components. If a check operation fails for a component, then the component is shut down and restarted.

This chapter contains the following information:

- Configuring Servers for Oracle Grid Infrastructure for a Standalone Server
- Oracle ACFS and Oracle ADVM
- Oracle Automatic Storage Management Storage Configuration
- Installing Oracle Grid Infrastructure for a Standalone Server Using a Software-Only Installation
- Installing and Configuring Oracle Grid Infrastructure for a Standalone Server
- Modifying Oracle Grid Infrastructure for a Standalone Server Binaries After Installation

Configuring Servers for Oracle Grid Infrastructure for a Standalone Server

This section describes the following operating system tasks that you must complete before you install Oracle Grid Infrastructure for a standalone server:

- Checking Hardware and Memory Configuration
- Server Memory Minimum Requirements
- Server Storage Minimum Requirements
- Environment Requirements for Oracle Grid Infrastructure Software Owner

Checking Hardware and Memory Configuration

Run the following commands to check your current system information:

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

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

   If necessary, see your operating system documentation for information about configuring additional swap space.

2. To determine the available RAM and swap space, enter the following command:

   ```
   # free
   ```

3. To determine the amount of free disk space on the system, enter one of the following commands:

   ```
   # df -h
   ```

4. To determine the amount of space available in the /tmp directory, enter one of the following commands:

   ```
   # df -h /tmp
   ```

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

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

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

Server Memory Minimum Requirements

Ensure that your system meets the following memory requirements for installing Oracle Grid Infrastructure for a standalone server:

At least 4 GB of RAM for Oracle Grid Infrastructure for a standalone server including installations where you plan to install Oracle Database.

The following table describes the relationship between installed RAM and the configured swap space recommendation:

Table 6–1   Swap Space Requirement for Oracle Restart

RAM	Swap Space
Between 4 GB and 16 GB	Equal to the size of the RAM
More than 16 GB	16 GB

> **Note:** On Linux, the HugePages feature allocates non-swappable memory for large page tables using memory-mapped files. If you enable HugePages, then you should deduct the memory allocated to HugePages from the available RAM before calculating swap space.

Server Storage Minimum Requirements

Ensure that your system meets the following minimum disk space requirements for installing Oracle Grid Infrastructure on a standalone server:

- For Linux x86-64: At least 6.9 GB of disk space

- For IBM: Linux on System z: 3.8 GB of disk space

- At least 1 GB of space in the `/tmp` directory.

  If there is less than 1 GB of free space in the `/tmp` directory, then complete one of the following steps:

  - Delete unnecessary files from the `/tmp` directory to meet the disk space requirement.

  - Set the `TMP` and `TMPDIR` environment variables to specify a temporary directory when setting the `oracle` user's environment.

    See Also: "Configuring Oracle Software Owner Environment" on page 5-13 for more information about setting `TMP` and `TMPDIR`

  - Extend the file system that contains the `/tmp` directory.

Environment Requirements for Oracle Grid Infrastructure Software Owner

Complete the following tasks to set the Oracle Restart software owner user's environment:

- Review the information in "Logging In to the System as root" on page 4-6 and "Configuring Oracle Software Owner Environment" on page 5-13.

- Ensure that you set the path to the Oracle base directory. Oracle recommends that you create the Oracle Restart home and the Oracle Database home under the same Oracle base directory. For example:

```
# ORACLE_BASE=/u01/app/oracle;
# export ORACLE_BASE
```

- Set the Oracle Restart software owner user default file mode creation mask (`umask`) to 022 in the shell startup file. Setting the mask to 022 ensures that the user performing the software installation creates files with 644 permissions.

- Set `ulimit` settings for file descriptors and processes for the Oracle Restart installation software owner.

- Set the `DISPLAY` environment variable in preparation for installation.

If you plan to install Oracle Database, then you must meet additional preinstallation requirements. See Chapter 4, "Oracle Database Preinstallation Tasks".

Oracle ACFS and Oracle ADVM

This section contains information about Oracle Automatic Storage Management Cluster File System (Oracle ACFS) and Oracle Automatic Storage Management Dynamic Volume Manager (Oracle ADVM). It contains the following topics:

- About Oracle ACFS and Oracle ADVM

- Oracle ACFS and Oracle ADVM Support on Linux

- Restrictions and Guidelines for Oracle ACFS

- Enabling Oracle ACFS on Oracle Restart Configurations

About Oracle ACFS and Oracle ADVM

Oracle ACFS extends Oracle ASM technology to support all of your application data in both single instance and cluster configurations. Oracle ADVM provides volume management services and a standard disk device driver interface to clients. Oracle Automatic Storage Management Cluster File System communicates with Oracle ASM through the Oracle Automatic Storage Management Dynamic Volume Manager interface.

Oracle ACFS and Oracle ADVM Support on Linux

Oracle ACFS and Oracle ADVM are supported on Oracle Linux, Red Hat Enterprise Linux, and SUSE Linux. Table 6–2 lists the platforms and kernel versions that support Oracle ACFS and Oracle ADVM.

Table 6–2   Platforms That Support Oracle ACFS and Oracle ADVM

Platform / Operating System	Kernel
Oracle Linux 7	■ Oracle Linux 7 with Red Hat Compatible Kernels: Update 0, RedHat Compatible Kernel 3.10.0-123 Update 1 and later, 3.10-0-229 and later RedHat Compatible kernels ■ Unbreakable Enterprise Kernel Release 3: All Updates, 3.8.13-35 and later UEK 3.8.13 kernels ■ Unbreakable Enterprise Kernel Release 4: All Updates, 4.1.12-32 and later UEK 4.1.12 kernels
Oracle Linux 6	■ Oracle Linux 6 with Red Hat Compatible Kernel ■ Unbreakable Enterprise Kernel Release 1: All Updates, 2.6.32-100 and later UEK 2.6.32 kernels ■ Unbreakable Enterprise Kernel Release 2: All Updates, 2.6.39-100 and later UEK 2.6.39 kernels ■ Unbreakable Enterprise Kernel Release 3: All Updates, 3.8.13 and later UEK 3.8.13 kernels ■ Unbreakable Enterprise Kernel Release 4: All Updates, 4.1.12-32 and later UEK 4.1.12 kernels
Oracle Linux 5	■ Oracle Linux 5 Update 3 with Red Hat Compatible Kernel: 2.6.18 or later ■ Unbreakable Enterprise Kernel Release 1: Update 3 and later, 2.6.32-100 and later UEK 2.6.32 kernels ■ Unbreakable Enterprise Kernel Release 2: Update 3 and later, 2.6.39-100 and later UEK 2.6.39 kernels
Red Hat Enterprise Linux 7	■ Red Hat Enterprise Linux 7: Update 0, 3.10.0-123 kernel Update 1 and later, 3.10.0-229 and later RedHat kernels
Red Hat Enterprise Linux 6	Red Hat Enterprise Linux 6
Red Hat Enterprise Linux 5	Red Hat Enterprise Linux 5 Update 3: 2.6.18 or later
SUSE Linux Enterprise Server 11	■ SUSE Linux Enterprise Server 11 Service Pack 2 (SP2) ■ SUSE Linux Enterprise Server 11 Service Pack 3 (SP3)

> **Note:** If you use Security Enhanced Linux (SELinux) in enforcing mode with Oracle ACFS, then ensure that you mount the Oracle ACFS file systems with an SELinux default context. Refer to your Linux vendor documentation for information about the context mount option.

See Also:

- My Oracle Support Note 1369107.1 for more information about platforms and releases that support Oracle ACFS and Oracle ADVM:

  https://support.oracle.com/CSP/main/article?cmd=show&type =NOT&id=1369107.1

- Patch Set Updates for Oracle Products (My Oracle Support Note 854428.1 for current release and support information:

  https://support.oracle.com/CSP/main/article?cmd=show&type =NOT&id=854428.1

Restrictions and Guidelines for Oracle ACFS

Oracle Restart does not support root-based Oracle Clusterware resources. For this reason, the following restrictions apply if you run Oracle ACFS on an Oracle Restart configuration:

- Oracle Restart does not support Oracle ACFS resources on all platforms.

- Starting with Oracle Database 12*c*, Oracle Restart configurations do not support the Oracle ACFS registry.

- You must manually load Oracle ACFS drivers after a system restart.

- You must manually mount an Oracle ACFS file system, and unmount it after the Oracle ASM instance has finished running.

- Creating Oracle datafiles on an Oracle ACFS file system is not supported in Oracle Restart configurations.

To enable Oracle ACFS on Oracle Restart see "Enabling Oracle ACFS on Oracle Restart Configurations" on page 6-6.

Note the following general restrictions and guidelines also about Oracle ACFS:

- Oracle ACFS can be used as an option only when Oracle ASM is configured.

- You can place Oracle database binaries, datafiles, and administrative files (for example, trace files) on Oracle ACFS. Oracle Database homes can be stored on Oracle ACFS only if the database version is Oracle Database 11*g* Release 2 or higher. Earlier releases of Oracle Database cannot be installed on Oracle ACFS.

- Oracle ACFS provides a general purpose file system for other files.

  See Also:

  - *Oracle Database Release Notes for Linux* for latest information about supported platforms and releases

  - "Introducing Oracle ACFS and Oracle ADVM" in *Oracle Automatic Storage Management Administrator's Guide*

Enabling Oracle ACFS on Oracle Restart Configurations

To use Oracle ACFS on Oracle Restart configurations, install Oracle Grid Infrastructure for a standalone server, and then enable root access for Oracle ACFS using the following command:

Log in as `root`

```
# cd Grid_home/crs/install
```

```
# roothas.sh -lockacfs
```

> **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), the `roothas.sh` script replaces the `roothas.pl` script in the Oracle Grid Infrastructure home.

Oracle Automatic Storage Management Storage Configuration

Review the following sections for information on Oracle Automatic Storage Management (Oracle ASM) storage configuration:

- Managing Disk Groups for Older Database Versions
- Oracle Automatic Storage Management Installation Considerations
- Configuring Storage for Oracle Automatic Storage Management
- About Oracle ASM with Oracle ASM Filter Driver
- Configuring Oracle ASM Disk Groups Manually using Oracle ASMCA
- Testing the Oracle Automatic Storage Management Installation
- Upgrading Existing Oracle Automatic Storage Management Instances

> **See Also:** "Using an Existing Oracle Automatic Storage Management Disk Group" on page 7-3

Managing Disk Groups for Older Database Versions

Use Oracle ASM Configuration Assistant (Oracle ASMCA) to create and modify disk groups when you install earlier Oracle databases on Oracle Grid Infrastructure installations.

Releases prior to Oracle Database 11*g* Release 2 used Oracle Database Configuration Assistant (Oracle DBCA) to perform administrative tasks on Oracle ASM. Starting with Oracle Database 11*g* Release 2 (11.2), Oracle ASM is installed as part of an Oracle Grid Infrastructure installation. You can no longer use Oracle DBCA to perform administrative tasks on Oracle ASM.

> **See Also:** *Oracle Automatic Storage Management Administrator's Guide* for details about configuring disk group compatibility for databases using Oracle Database 11*g* or earlier software with Oracle Grid Infrastructure 12*c*.

Oracle Automatic Storage Management Installation Considerations

In previous releases, Oracle Automatic Storage Management (Oracle ASM) was installed as part of the Oracle Database installation. Starting with Oracle Database 11*g* Release 2 (11.2), Oracle ASM is part of an Oracle Grid Infrastructure installation, either for a cluster, or for a standalone server.

If you want to upgrade an existing Oracle ASM installation, then you must upgrade Oracle ASM by running an Oracle Grid Infrastructure upgrade (upgrades of existing Oracle ASM installations). If you do not have Oracle ASM installed and you want to use Oracle ASM as your storage option, then you must complete an Oracle Grid Infrastructure for a standalone server installation before you start your Oracle Database installation.

Oracle Automatic Storage Management Configuration Assistant (Oracle ASMCA) is installed as part of the Oracle Grid Infrastructure installation. You must run Oracle ASMCA for installing and configuring Oracle ASM instances, disk groups, volumes, and Oracle ACFS. In addition, you can use the ASMCA command-line interface.

> **See Also:**
>
> - "Managing Oracle Flex ASM" in *Oracle Automatic Storage Management Administrator's Guide*
>
> - "Oracle ASM Configuration Assistant" in *Oracle Automatic Storage Management Administrator's Guide* for information about Oracle ASMCA

Configuring Storage for Oracle Automatic Storage Management

This section contains the following topics that describe how to configure storage for use with Oracle Automatic Storage Management:

- Identifying Storage Requirements for Oracle Automatic Storage Management
- Creating DAS or SAN Disk Partitions for Oracle Automatic Storage Management

> **See Also:** "Creating Disk Groups for a New Oracle Installation" in *Oracle Automatic Storage Management Administrator's Guide* for information about creating and managing disk groups

> **See Also:** "Configuring Storage Paths and Disk Devices" for information about using Oracle ASMLIB

> **Note:** Oracle does not recommend using identifiers for database object names that must be quoted. While these quoted identifiers may be valid as names in the `SQL CREATE` statement, such as `CREATE DISKGROUP "1data" ...`, the names may not be valid when using other tools that manage the database object.

Identifying Storage Requirements for Oracle Automatic Storage Management

To identify the storage requirements for using Oracle ASM, you must determine the number of devices and the amount of free disk space that you require. To complete this task, follow these steps:

1. Determine whether you want to use Oracle ASM for Oracle Database files, recovery files, or both. Oracle Database files include data files, control files, redo log files, the server parameter file, and the password file.

   During the database installation, you have the option to select either a file system or Oracle ASM as the storage mechanism for Oracle Database files. Similarly, you also have the option to select either a file system or Oracle ASM as the storage mechanism for your recovery files.

   > **Note:** You do not have to use the same storage mechanism for both Oracle Database files and recovery files. You can use a file system for one file type and Oracle ASM for the other.

If you select Oracle ASM as your storage option for Oracle Database files, then depending on your choice in the Specify Recovery Options screen, you have the following recovery options:

- If you select the Oracle ASM option for your recovery files, then Oracle Universal Installer provides you with only the option to use the same disk group for both Oracle Database files and recovery files.

- If you decide not to enable recovery during the database installation, then, after the database installation, you can modify the DB_RECOVERY_FILE_DEST parameter to enable the fast recovery area.

  See Also:

  - "Oracle ASM Configuration Assistant Command-Line Interface" section in *Oracle Automatic Storage Management Administrator's Guide*

  - "Creating a Fast Recovery Area Disk Group" on page 8-4

2. Choose the Oracle ASM redundancy level to use for each Oracle ASM disk group that you create.

   The redundancy level that you choose for the Oracle ASM disk group determines how Oracle ASM mirrors files in the disk group and determines the number of disks and amount of disk space that you require, as follows:

   - External redundancy

     This option does not allow Oracle ASM to mirror the contents of the disk group. Oracle recommends that you select this redundancy level either when the disk group contains devices, such as RAID devices, that provide their own data protection or when the database does not require uninterrupted access to data.

   - Normal redundancy

     To optimize performance and reliability in a normal redundancy disk group, Oracle ASM uses two-way mirroring for data files and three-way mirroring for control files, by default. In addition, you can choose the mirroring characteristics for individual files in a disk group. You can use two-way mirroring or no mirroring.

     A normal redundancy disk group requires a minimum of two failure groups (or two disk devices) if you are using two-way mirroring. The effective disk space in a normal redundancy disk group is half the sum of the disk space of all of its devices.

     For most installations, Oracle recommends that you use normal redundancy disk groups.

   - High redundancy

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

     Although high-redundancy disk groups provide a high level of data protection, you must consider the higher cost of additional storage devices before deciding to use this redundancy level.

3. Determine the total amount of disk space that you require for the database files and recovery files.

If an Oracle ASM instance is running on the system, then you can use an existing disk group to meet these storage requirements. If necessary, you can add disks to an existing disk group during the database installation.

Use the following table to determine the minimum number of disks and the minimum disk space requirements for the installation:

Redundancy Level	Minimum Number of Disks	Data Files	Recovery Files	Both File Types
External	1	1.8 GB	3.6 GB	5.4 GB
Normal	2	3.6 GB	7.2 GB	10.8 GB
High	3	5.4 GB	10.8 GB	16.2 GB

4. Optionally, identify failure groups for the Oracle ASM disk group devices.

If you intend to use a normal or high redundancy disk group, then you can further protect the database against hardware failure by associating a set of disk devices in a custom failure group. By default, each device is included in its failure group. However, if two disk devices in a normal redundancy disk group are attached to the same small computer system interface (SCSI) controller, then the disk group becomes unavailable if the controller fails. The controller in this example is a single point of failure.

For instance, to avoid failures of this type, you can use two SCSI controllers, each with two disks, and define a failure group for the disks attached to each controller. This configuration would enable the disk group to tolerate the failure of one SCSI controller.

> **Note:** Define custom failure groups after installation, using the GUI tool ASMCA, the command line tool `asmcmd`, or SQL commands. If you define custom failure groups, then you must specify a minimum of two failure groups for normal redundancy disk groups and three failure groups for high redundancy disk groups.

5. If you are sure that a suitable disk group does not exist on the system, then install or identify appropriate disk devices to add to a new disk group. Use the following guidelines when identifying appropriate disk devices:

   - The disk devices must be owned by the user performing the grid installation.

     > **See Also:** "Configuring a Permissions File for Disk Devices for Oracle ASM" on page D-12 for information about creating or modifying permissions

   - All the devices in an Oracle ASM disk group must be the same size and have the same performance characteristics.

   - Do not specify multiple partitions on a single physical disk as a disk group device. Oracle ASM expects each disk group device to be on a separate physical disk.

   - Although you can specify a logical volume as a device in an Oracle ASM disk group, Oracle does not recommend their use because it adds a layer of complexity that is unnecessary with Oracle ASM. Oracle recommends that if

you choose to use a logical volume manager, then use the logical volume manager to represent a single logical unit number (LUN) without striping or mirroring, so that you can minimize the effect on storage performance of the additional storage layer.

> **See Also:**
>
> - "About Oracle ASM with Oracle ASM Filter Driver" for information about completing this task
>
> - "Preparing Storage for ASM" in *Oracle Automatic Storage Management Administrator's Guide* for information about configuring Oracle ASM disk groups

Creating DAS or SAN Disk Partitions for Oracle Automatic Storage Management

In order to use a DAS or SAN disk in Oracle ASM, the disk must have a partition table. Oracle recommends creating exactly one partition for each disk.

> **Note:** You can use any physical disk for Oracle ASM, if it is partitioned.

> **See Also:** "Creating Files on a NAS Device for Use with Oracle Automatic Storage Management" on page C-4.

About Oracle ASM with Oracle ASM Filter Driver

Starting with Oracle Database 12*c* Release 1 (12.1.0.2), Oracle ASM Filter Driver (Oracle ASMFD) is installed with an Oracle Grid Infrastructure installation.

Oracle Automatic Storage Management Filter Driver (Oracle ASMFD) rejects write I/O requests that are not issued by Oracle software. This filter helps to prevent users with administrative privileges from inadvertently overwriting Oracle ASM disks, thus preventing corruption in Oracle ASM disks and files within the disk group. For disk partitions, the area protected is the area on the disk managed by Oracle ASMFD, assuming the partition table is left untouched by the user.

Oracle ASMFD simplifies the configuration and management of disk devices by eliminating the need to rebind disk devices used with Oracle ASM each time the system is restarted.

> **See Also:** *Oracle Automatic Storage Management Administrator's Guide* for more information about configuring and migrating disk devices to use Oracle ASM Filter Driver

Configuring Oracle ASM Disk Groups Manually using Oracle ASMCA

The Oracle Automatic Storage Management Configuration Assistant (Oracle ASMCA) utility creates a new Oracle Automatic Storage Management instance if there is no Oracle ASM instance currently configured on the computer.

After installing Oracle Grid Infrastructure for a standalone server, you can also use Oracle ASMCA to create and configure disk groups, Oracle ADVM and Oracle ACFS.

To create additional disk groups or manually configure Oracle ASM disks, you can run the Oracle ASMCA as follows:

```
$ cd Grid_home/bin
$ ./asmca
```

Grid_home is the path to the Oracle Grid Infrastructure for a standalone server home.

> **See Also:** *Oracle Automatic Storage Management Administrator's Guide*

Testing the Oracle Automatic Storage Management Installation

After installing Oracle Grid Infrastructure for a standalone server, as described in this chapter, to test the Oracle ASM installation, log in using the asmcmd command-line utility, which lets you manage Oracle ASM disk group files and directories. To do this:

1. Open a shell window, and temporarily set the ORACLE_SID and ORACLE_HOME environment variables to specify the appropriate values for the Oracle ASM instance to use.

   For example, if the Oracle ASM SID is named +ASM and the Oracle home is located in the grid subdirectory of the ORACLE_BASE directory, then enter the following commands to create the required settings:

   - Bourne, Bash, or Korn shell:

     ```
     $ ORACLE_SID=+ASM
     $ export ORACLE_SID
     $ ORACLE_HOME=/u01/app/oracle/product/12.1.0/grid
     $ export ORACLE_HOME
     ```

   - C shell:

     ```
     % setenv ORACLE_SID +ASM
     % setenv ORACLE_HOME /u01/app/oracle/product/12.1.0/grid
     ```

2. Use ASMCMD to list the disk groups for the Oracle ASM instance:

   ```
   $ORACLE_HOME/bin/asmcmd lsdg
   ```

   ASMCMD connects by default as the SYS user with SYSASM privileges.

   If the Oracle ASM instance is not running, start the instance with the following command:

   ```
   $ORACLE_HOME/bin/asmcmd startup
   ```

> **See Also:** "Oracle ASM Command-Line Utility" in *Oracle Automatic Storage Management Administrator's Guide*

Upgrading Existing Oracle Automatic Storage Management Instances

If you have an Oracle ASM installation from an earlier release installed on your server, or in an existing Oracle Restart installation, you can use Oracle Automatic Storage Management Configuration Assistant (Oracle ASMCA) to upgrade the existing Oracle ASM instance to 12*c*, and subsequently configure disk groups, Oracle ASM volumes and Oracle ASM file systems.

> **Note:** You must first shut down all databases and applications using an existing Oracle ASM instance before upgrading it.

During the installation, if you use Oracle ASM, and Oracle ASMCA detects that there is a prior Oracle ASM version installed in another Oracle ASM home, then after installing the Oracle ASM 12*c* binaries, you can start Oracle ASMCA to upgrade the existing Oracle ASM instance.

> **See Also:**
>
> - "Upgrade an Oracle ASM Instance" in *Oracle Automatic Storage Management Administrator's Guide*
>
> - "Upgrading an Oracle ASM Instance with Oracle ASM Configuration Assistant" in *Oracle Automatic Storage Management Administrator's Guide*
>
> - "Downgrading an Oracle ASM Instance in an Oracle Restart Configuration" in *Oracle Automatic Storage Management Administrator's Guide*

Installing Oracle Grid Infrastructure for a Standalone Server Using a Software-Only Installation

A software-only installation only copies the Oracle Grid Infrastructure for a standalone server binaries to the specified location. After the installation, manually configure Oracle Grid Infrastructure for a standalone server and Oracle ASM.

When you perform a software-only installation of Oracle Grid Infrastructure software, you must complete a few manual configuration steps to enable Oracle Restart after you install the software.

> **Note:** Oracle recommends that only advanced users perform the software-only installation because this installation method provides no validation of the installation, and this installation option requires manual postinstallation steps to enable the Oracle Grid Infrastructure for a standalone server software.

Performing a software-only installation involves the following steps:

1. Installing the Software Binaries
2. Configuring the Software Binaries

Installing the Software Binaries

1. Run the `runInstaller` command from the relevant directory on the Oracle Database 12*c* installation media or download directory.

2. Complete a software-only installation of Oracle Grid Infrastructure for a standalone server.

3. Run the `orainstRoot.sh` script if prompted.

4. Oracle Universal Installer (OUI) prompts you to run the `root.sh` script.

5. `root.sh` then prompts you to run `roothas.pl`. See "Configuring the Software Binaries" on page 6-14 for information about running `roothas.pl` and configuring Oracle Grid Infrastructure for a standalone server after performing a software-only installation.

6. Verify that the server meets the installation requirements using the command `runcluvfy.sh stage -pre hacfg`. Ensure that you complete all storage and server preinstallation requirements.

Configuring the Software Binaries

To configure and activate a software-only Oracle Grid Infrastructure for a standalone server installation for Oracle Restart, complete the following tasks:

1. Log in as the `root` user and run the `roothas.pl` script from *Grid_home* using the following syntax:

   ```
   Grid_home/perl/bin/perl -I Grid_home/perl/lib -I Grid_home/crs/install
   Grid_home/crs/install/roothas.pl
   ```

   For example, if your Oracle Grid Infrastructure for a standalone server home is `u01/app/oracle/product/12.1.0/grid`, then run the following script:

   ```
   # /u01/app/oracle/product/12.1.0/grid/perl/bin/perl -I
   /u01/app/oracle/product/12.1.0/grid/perl/lib -I /u01/app/oracle/product
   /12.1.0/grid/crs/install
   /u01/app/oracle/product/12.1.0/grid/crs/install/roothas.pl
   ```

2. Change the directory to *Grid_home*/oui/bin, where *Grid_home* is the path of the Oracle Grid Infrastructure for a standalone server home.

3. Log in as the Oracle Restart software owner user and enter the following command:

   ```
   ./runInstaller -updateNodeList ORACLE_HOME=Grid_home -defaultHomeName CLUSTER_
   NODES= CRS=TRUE
   ```

   For example:

   ```
   $ ./runInstaller -updateNodeList ORACLE_
   HOME=/u01/app/oracle/product/12.1.0/grid
   -defaultHomeName CLUSTER_NODES= CRS=TRUE
   ```

4. Use the `SRVCTL` utility along with Oracle Network Configuration Assistant and Oracle ASMCA to add the listener, the Oracle ASM instance, and all Oracle ASM disk groups to the Oracle Restart configuration.

   > **See Also:**
   >
   > - *Oracle Database Net Services Administrator's Guide* to configure a listener using Oracle Network Configuration Assistant
   > - *Oracle Automatic Storage Management Administrator's Guide* to create and add disk groups, and configure Oracle ASM using Oracle ASMCA
   > - *Oracle Database Administrator's Guide* to create and add an ASM instance using SRVCTL

Installing and Configuring Oracle Grid Infrastructure for a Standalone Server

If you install Oracle Grid Infrastructure for a standalone server and then create your database, the database is automatically added to the Oracle Restart configuration and is automatically restarted when required. However, if you install Oracle Grid

Infrastructure for a standalone server on a host computer on which a database exists, you must manually add the database, the listener, the Oracle ASM instance, and other components to the Oracle Restart configuration.

> **Note:** Oracle Grid Infrastructure for a standalone server can accommodate multiple single-instance databases on a single host computer.

This section includes the following topics:

- Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation
- Installing Oracle Grid Infrastructure for a Standalone Server for an Existing Database

Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation

Perform the following steps to install Oracle Grid Infrastructure for a standalone server and then create a database that is managed by Oracle Restart. First install Oracle Grid Infrastructure for a standalone server, which installs Oracle Restart and Oracle ASM, then configure Oracle ASM with at least one disk group, and then install Oracle Database that stores database files in Oracle ASM disk groups. Click the help button on the Oracle Universal Installer page for page level assistance.

You may have to shut down existing Oracle processes before you proceed with the Oracle Grid Infrastructure for a standalone server installation. See "Stopping Existing Oracle Processes" on page 5-12 for more information.

To install Oracle Grid Infrastructure for a standalone server with a new database installation:

1. Start Oracle Universal Installer as the Oracle Restart software owner user. Complete one of the following steps depending on the location of the installation files:

   - If the installation files are on installation media, enter commands similar to the following, where *directory_path* is the path of the Oracle Grid Infrastructure for a standalone server directory on the installation media:

     ```
     $ /directory_path/runInstaller
     ```

     > **Note:** You must install Oracle Grid Infrastructure for a standalone server from the Oracle Grid Infrastructure media.

   - If the installation files are on the hard disk, change the directory to the path of the Oracle Grid Infrastructure for a standalone server (clusterware) directory and enter the following command:

     ```
     $ ./runInstaller
     ```

> **Note:** Start Oracle Universal Installer from the terminal session where you logged in as the Oracle Restart software owner user and set the user's environment.

If Oracle Universal Installer is not displayed, see "X Window Display Errors" on page I-2 and "Remote Terminal Installation Error" on page I-2 for information about troubleshooting.

2. In the Select Installation Option screen, select the **Install and Configure Oracle Grid Infrastructure for a Standalone Server** option to install and configure Oracle Restart and Oracle ASM. Click **Next.**

3. In the Select Product Languages screen, select one or more languages. Move the languages from the Available Languages list to the Selected Languages list. Click **Next.**

4. The Create ASM Disk Group screen lists all the Oracle ASM disks.

   Click **Change Discovery Path** to select any devices to be used by Oracle ASM but are not listed in the screen. In the Change Disk Discovery Path window, enter a string to use to search for devices that Oracle ASM will use. If the disk string is set to `ORCL:*` or is left empty (""), then the installer discovers these disks. Click **OK.**

   After you finish selecting the disks to be used by Oracle ASM, click **Next**.

   > **Note:** During installation, disk paths mounted on Oracle ASM and registered on Oracle ASMFD with the string `ORCL:*` are listed as default database storage candidate disks.

   Consider the following information about disk devices while performing this step:

   - The default Disk Group Name is `DATA`. You can enter a new name for the disk group, or use the default name.

   - The disk devices must be owned by the user performing the grid installation.

     > **See Also:** "Configuring a Permissions File for Disk Devices for Oracle ASM" on page D-12 for information about creating or modifying permissions

   - Check with your system administrator to determine if the disks used by Oracle ASM are mirrored at the storage level. If so, select **External** for the redundancy. If the disks are not mirrored at the storage level, then select **Normal** for the redundancy.

     > **Note:** For normal redundancy, you require twice as much disk space to hold the same amount of data. For example, if your database is 100 GB, then you require approximately 200 GB of storage.

   - Every Oracle ASM disk is divided into allocation units (AU). An allocation unit is the fundamental unit of allocation within a disk group. You can select the **AU Size** value from 1, 2, 4, 8, 16, 32 or 64 MB, depending on the specific disk group compatibility level. The default value is set to 1 MB.

5. In the Specify ASM Password screen, enter the password required to connect to the Oracle ASM instance. The Oracle ASM instance is managed by a privileged role called `SYSASM`, which grants full access to Oracle ASM disk groups. Oracle recommends that you create a less privileged user, `ASMSNMP`, with `SYSDBA` privileges to monitor the Oracle ASM instance.

   Enter passwords for the `SYS` and `ASMSNMP` user accounts. The passwords should be at least eight characters in length and include at least 1 alphabetic, 1 numeric, and 1 of the following three punctuation mark characters: hyphens (-), underscores (_), or number sign (#). No other special characters are allowed in the password field.

   Optionally, you can use the same password for all accounts. However, Oracle recommends that you specify a different password for each account. You must remember the passwords that you specify.

6. The Specify Management Options screen gives you the option to manage Oracle Grid Infrastructure and Oracle Automatic Storage Management using Oracle Enterprise Manager Cloud Control. Select **Register with Enterprise Manager (EM) Cloud Control** and specify the following configuration information:

   - OMS Host: The system name where the Management repository is running.

   - OMS Port: The Oracle Enterprise Manager port number to receive requests from the Management service.

   - EM Admin User Name: The user name to log in to Oracle Enterprise Manager.

   - EM Admin Password: The password to log in to Oracle Enterprise Manager.

7. In the Privileged Operating System Groups screen, select the name of the operating system group you created for the `OSDBA` group, the `OSASM` group, and the Oracle ASM operator group `OSOPER`. If you create only the `dba` group, then you can use that group for all three privileged groups. If you created a separate `asmadmin` group, then use that value for the `OSASM` group. Click **Next**.

   > **See Also:** "Oracle Automatic Storage Management Groups for Job Role Separation" on page 5-5

8. In the Specify Installation Location screen, enter the following information, and click **Next:**

   - **Oracle Base**: Enter the location for the Oracle base directory. Do not include spaces in the path.

   - **Software Location:** Accept the default value or enter the directory path in which you want to install the software.The directory path must not contain spaces.

     > **See Also:**
     >
     > - "Identifying Required Software Directories" on page 4-34 for information about Oracle base directory and Oracle home directory
     >
     > - "Naming Directories" on page F-2 for directory naming conventions

9. The Root Script Execution Configuration screen allows you to select the configuration method to either automatically or manually run the root scripts during the grid infrastructure installation. To run root scripts automatically, select **Automatically run configuration scripts** and select one of the following options:

- Use "root" user credential: Provide the `root` user password.

- Use Sudo: Provide the program path, username, and password for the sudo program.

  See Also: "Determining Root Script Execution Plan" on page 5-16

10. The Create Inventory screen is displayed only if this is the first time you are installing Oracle software on your system.

Change the path for the Inventory Directory, if required. Select `oinstall` for the oraInventory Group Name, if required. Click **Next.**

11. The Perform Prerequisite Checks screen checks if the minimum system requirements are met to perform the Oracle Grid Infrastructure for a standalone server installation. If all the system requirements are met, then you are directed to the Summary screen. If an installation fails, you can review the error.

If you click **Check Again**, then you can run the prerequisite check again to see if the minimum requirements are met to carry on with the database installation.

Click **Fix & Check Again**, if you want the installer to fix the problem and check the system requirements again.

> **Note:** The Fix & Check Again option generates a script that you must run as the `root` user. This generated script sets some system parameter values. Oracle recommends that you do not modify the contents of this script. See "Using Installation Fixup Scripts" on page 4-10 for more information.

To get a list of failed requirements, select **Show Failed** from the list. To get a list of all the prerequisite checks run by Oracle Universal Installer, select **Show All**. To get a list of the prerequisites checks that are successful, select **Show Succeeded.**

> **Note:** Oracle recommends that you use caution when selecting the Ignore All option. If you select this option, then Oracle Universal Installer may not confirm that your system can install Oracle Database successfully.

See Also:

- "Configuring Servers for Oracle Grid Infrastructure for a Standalone Server" on page 6-2

- Chapter 4, "Oracle Database Preinstallation Tasks"

12. Review the contents of the Summary screen, and click **Install.**

You can click **Save Response File** to save all the installation steps into a response file. This file can be used for a silent installation.

13. The Install Product screen displays the progress of the Oracle Grid Infrastructure for a standalone server installation.

If you selected an option to automatically run the root scripts in the Root Script Execution Configuration screen, then Oracle Universal Installer automatically runs the root scripts. Click **Next.**

Else, if you did not select the option to automatically run the configuration scripts, then Oracle Universal Installer prompts you to run the `root.sh` script and, if required, the `orainstRoot.sh` script as the `root` user to complete the installation. During this process, the Execute Configuration Scripts window appears. Do *not* click **OK** until you run the scripts mentioned in this screen.

> **Note:** The `orainstRoot.sh` script must be run if this is the first time you are installing Oracle software on your system. However, if you selected an option to automatically run the root scripts, then Oracle Universal Installer automatically runs the `oraInstRoot.sh` script.

14. The Finish screen displays the installation status. Click **Close** to end the installation and exit Oracle Universal Installer.

If you encounter any errors, see the configuration log for information. The path to the configuration log is displayed on the Configuration Assistants window.

15. Oracle ASMCA is installed as part of the Oracle Grid Infrastructure for a standalone server installation. To create additional disk groups, run the Oracle ASMCA utility. For example, you can create another disk group named `RECOVERY` to store the fast recovery area.

> **See Also:**
>
> - "Configuring Oracle ASM Disk Groups Manually using Oracle ASMCA" on page 6-11
> - "Creating a Fast Recovery Area Disk Group" on page 8-4

> **Note:** To verify that the Oracle High Availability Service is installed properly, run `./crsctl check has` command from *Grid_home*/bin directory.
>
> *Grid_home* is the path to the Oracle Grid Infrastructure for a standalone server home. `ohasd` is a daemon installed with Oracle Grid Infrastructure that starts software services, such as Oracle ASM.

16. Install Oracle Database. See "Installing the Oracle Database Software" on page 7-8.

> **Note:**
>
> - If a new database is installed after a grid infrastructure installation, then the listener runs from the Oracle Grid Infrastructure for a standalone server home. Because Oracle ASM is installed as part of Oracle Grid Infrastructure for a standalone server, the default listener is created and runs from the Oracle Grid Infrastructure for a standalone server home. If you perform a database installation, then the database must use the same listener created during the Oracle Grid Infrastructure for a standalone server installation.
> - If you are using Oracle Restart, then the default listener and any additional listeners must run from the Oracle Grid Infrastructure for a standalone server home.

See "Troubleshooting and Deconfiguring Oracle Restart" on page I-4 to deconfigure Oracle Restart without removing installed binaries.

Installing Oracle Grid Infrastructure for a Standalone Server for an Existing Database

Follow the high-level instructions in this section to install Oracle Grid Infrastructure for a standalone server and configure it for an existing Oracle database. Oracle Restart can only manage existing release 12.1 resources and hence you can install Oracle Grid Infrastructure for a standalone server only for an existing release 12.1 database. However, Oracle database releases before 12.1 can coexist on the same server without being managed by Oracle Restart.

To install Oracle Grid Infrastructure for a standalone server for an existing database:

- On the same host computer as the database, use Oracle Universal Installer to install Oracle Grid Infrastructure for a standalone server, and select **Install and Configure Oracle Grid Infrastructure for a Standalone Server** as the installation option.

    The Oracle Grid Infrastructure for a standalone server components are installed in a separate Oracle home.

    See "Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation" on page 15 for detailed instructions.

- Go to the Grid home's `bin` directory.

    Use the `srvctl add database` command with the `-c SINGLE` flag to add the database in an Oracle Restart configuration. Also use the `srvctl add` command to add the listener, the Oracle ASM instance, all Oracle ASM disk groups, and any database services to the Oracle Restart configuration.

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

Modifying Oracle Grid Infrastructure for a Standalone Server Binaries After Installation

After the Oracle Grid Infrastructure for a standalone server installation, you must first stop the Oracle Restart stack to modify the software installed in your Grid home. For example, to apply a one-off patch or modify any of the DLLs used by Oracle Restart or Oracle ASM, you must follow these steps to stop and restart the Oracle Restart stack.

However, if you run the OPatch utility with the `auto` option, `opatchauto`, then do not stop and restart the Oracle Restart stack manually as OPatch stops and starts the software stack for you. If you run OPatch with the `apply` option, `opatch apply`, then you must follow the steps in this section to stop and restart the Oracle Restart stack manually.

> **See Also:** *Oracle OPatch User's Guide for Windows and UNIX* for information about using `opatchauto`

> **Caution:** Before relinking executables, you must shut down all executables that run in the Oracle home directory that you are relinking. In addition, shut down applications linked with Oracle shared libraries.

Prepare the Oracle Grid Infrastructure for a standalone server home for modification using the following procedure:

1. Log in as the Oracle Grid Infrastructure for a standalone server software owner user and change the directory to the path *Grid_home*/bin, where *Grid_home* is the path to the Oracle Grid Infrastructure for a standalone server home:

   ```
   $ cd Grid_home/bin
   ```

2. Shut down the Oracle Restart stack using the following command:

   ```
   $ crsctl stop has -f
   ```

3. Perform the updates to the software installed in the Oracle Grid Infrastructure for a standalone server home.

4. Enter the following command to restart the Oracle Restart stack:

   ```
   $ crsctl start has
   ```

Relink Oracle Grid Infrastructure for a standalone server using the following procedure:

1. Log in as `root`

   ```
   # cd Grid_home/crs/install
   # roothas.sh -unlock
   ```

2. Log in as the Oracle Grid Infrastructure for a standalone server owner:

   ```
   $ export ORACLE_HOME=Grid_home
   $ Grid_home/bin/relink
   ```

3. Log in as `root` again:

   ```
   # cd Grid_home/rdbms/install/
   # ./rootadd_rdbms.sh
   # cd Grid_home/crs/install
   # roothas.sh -patch
   ```

   You must relink the Oracle Restart and Oracle ASM binaries every time you apply an operating system patch or after an operating system upgrade.

   > **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), the `roothas.sh` script replaces the `roothas.pl` script in the Oracle Grid Infrastructure home.

   See Also:

   - "Deinstalling Previous Release Grid Home" on page 10-5
   - *Oracle Database Administrator's Guide*

7

Installing Oracle Database

The Oracle Database software is available on installation media, or you can download it from the Oracle Technology Network website, or the Oracle Software Delivery Cloud portal. 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 silent mode installations, without using the GUI.

- Preinstallation Considerations

- Reviewing Component-Specific Installation Guidelines

- Accessing the Installation Software

- Installing the Oracle Database Software

- Installing Oracle Database Examples

> **See Also:** Appendix A for information about silent mode installations

Preinstallation Considerations

Review the information in Chapter 2, "Overview of Oracle Database Installation" and complete the tasks listed in Chapter 4, "Oracle Database Preinstallation Tasks".

Performing Multiple Oracle Database Installations in Response File or Silent Mode

If you must perform multiple installations of Oracle Database, you may want to use silent mode or response file mode. In response file mode, at each node, you run Oracle Universal Installer from the command line using a response file. The response file is a text file that contains the settings you typically enter in the Oracle Universal Installer GUI dialog boxes.

> **See Also:** Appendix A for information about silent mode installations

Reviewing Component-Specific Installation Guidelines

Review the following guidelines before starting Oracle Universal Installer:

- Oracle Universal Installer

  Using Oracle Universal Installer from an earlier Oracle release to install components from this release is no longer allowed.

- Oracle Automatic Storage Management

In previous releases, Oracle Automatic Storage Management (Oracle ASM) was installed as part of the Oracle Database installation. Starting with Oracle Database 11*g* Release 2 (11.2), Oracle ASM is part of an Oracle Grid Infrastructure installation, either for a cluster, or for a standalone server.

To upgrade an existing Oracle ASM installation, upgrade Oracle ASM by running an Oracle Grid Infrastructure upgrade. If you do not have Oracle ASM installed and you want to use Oracle ASM as your storage option, then you must complete an Oracle Grid Infrastructure for a standalone server installation before you start your Oracle Database installation.

> **See Also:** Chapter 6, "Oracle Grid Infrastructure for a Standalone Server" for information about Oracle Grid Infrastructure for a standalone server

- Installations on a cluster

  If Oracle Clusterware or Oracle RAC is installed on the system, then Oracle Universal Installer displays the Grid Installation Options screen. You must select Single instance database installation, unless you want to install Oracle RAC. The other options in this screen are Oracle Real Application Clusters database installation and Oracle RAC One Node database installation.

  > **See Also:** *Oracle Real Application Clusters Installation Guide for Linux and UNIX*

This section covers the following topics:

- Selecting the Database Character Set
- Using an Existing Oracle Automatic Storage Management Disk Group

Selecting the Database Character Set

Oracle Database uses the database character set for:

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

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

Oracle recommends Unicode AL32UTF8 as the database character set. Unicode is the universal character set that supports most of the currently spoken languages of the world. It also supports many historical scripts (alphabets). Unicode is the native encoding of many technologies, including Java, XML, XHTML, ECMAScript, and LDAP. Unicode is ideally suited for databases supporting the Internet and the global economy.

Because AL32UTF8 is a multibyte character set, database operations on character data may be slightly slower when compared to single-byte database character sets, such as WE8MSWIN1252. Storage space requirements for text in most languages that use characters outside of the ASCII repertoire are higher in AL32UTF8 compared to legacy character sets supporting the language. The increase in storage space concerns only

character data and only data that is not in English. The universality and flexibility of Unicode usually outweighs these additional costs.

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

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

> **See Also:** "Choosing a Database Character Set for a CDB" in *Oracle Database Globalization Support Guide*

The default character set suggested or used by Oracle Universal Installer and Database Configuration Assistant in this release is based on the language configuration of the operating system.

For most languages, the default character set is one of the Microsoft Windows character sets, for example WE8MSWIN1252, even though the database is not installed on Windows. This results from the assumption that most clients connecting to the database run under the Microsoft Windows operating system. Because the database should be able to store all characters coming from the clients and Microsoft Windows character sets have a richer character repertoire than the corresponding ISO 8859 character sets, the Microsoft Windows character sets are usually the better choice. For example, the EE8MSWIN1250 character set supports the Euro currency symbol and various smart quote characters, while the corresponding EE8ISO8859P2 character set does not support them. Oracle Database converts the data between the database character set and the client character sets, which are declared by the NLS_LANG settings.

The list of database character sets that is presented to you for selection by Oracle Universal Installer contains only the recommended character sets. Even though Oracle Database supports many more character sets, they are either deprecated or they are binary subsets of another recommended character set. For example, WE8DEC is a deprecated character set and US7ASCII and WE8ISO8859P1 are both binary subsets of WE8MSWIN1252.

If, for compatibility reasons, you must create a database in one of the character sets that was not recommended, select the Advanced database configuration option. Oracle Database Configuration Assistant (Oracle DBCA) in the interactive mode gives you the opportunity to select any of the database character sets supported on Linux.

> **See Also:** "Choosing a Character Set" in *Oracle Database Globalization Support Guide*

Using an Existing Oracle Automatic Storage Management Disk Group

This section is optional and describes how to identify disk groups and determine the free disk space that they contain. You can store either database or recovery files in an existing Oracle ASM disk group that you created during the Oracle Grid Infrastructure for a standalone server installation.

> **Note:** The Oracle ASM instance that manages the existing disk group runs in the Oracle Grid Infrastructure home directory.

To determine if an existing Oracle ASM disk group exists or to determine if there is sufficient disk space in a disk group, you can use Oracle Enterprise Manager Cloud Control or you can use the following Oracle ASM command line tool (`asmcmd`) procedure:

1. Connect to the Oracle Automatic Storage Management instance and start the instance if necessary:

    ```
    # $ORACLE_HOME/bin/asmcmd
    ASMCMD> startup
    ```

2. Enter one of the following commands to view the existing disk groups, their redundancy level, and the amount of free disk space in each one:

    ```
    ASMCMD> lsdg;
    ```

    or

    ```
    $ORACLE_HOME/bin/asmcmd -p lsdg
    ```

3. From the output, identify a disk group with the appropriate redundancy level, and note the free space that it contains.

4. If necessary, install or identify the additional disk devices required to meet the storage requirements listed in the previous section.

    Note: If you are adding devices to an existing disk group, then Oracle recommends that you use devices that have the same size and performance characteristics as the existing devices in that disk group.

    See Also: "Oracle Automatic Storage Management Storage Configuration" on page 6-7

Accessing the Installation Software

The Oracle Database software is available on installation media, or you can download it from the Oracle Technology Network website, or the Oracle Software Delivery Cloud portal. To install the software from the hard disk, you must either download it and unpack it, or copy it from the installation media, if you have it.

You can access and install Oracle Database by using one of the following methods:

- To copy the software to a hard disk, see "Copying the Software to the Hard Disk" on page 7-7

- To download the software from Oracle Technology Network, see "Downloading Oracle Software" on page 7-4

Downloading Oracle Software

You can download the trial version of the installation files from the Oracle Technology Network (OTN) or the Oracle Software Delivery Cloud portal and extract them on your hard disk. Ensure that you review and understand the terms of the license. Most downloads include the development license. This section contains the following topics:

- Downloading the Installation Archive Files from OTN

- Downloading the Software from Oracle Software Delivery Cloud Portal
- Extracting the Installation Files

Downloading the Installation Archive Files from OTN

To download the installation archive files from Oracle Technology Network:

1. Use any browser to access the software download page from Oracle Technology Network:

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

2. Go to the download page for the product to install.

3. On the download page, 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 file system 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 file system that you selected in Step 4, create a parent directory for each product, for example OraDB12c, to hold the installation directories.

6. Download all of the installation archive files to the directory that you created in Step 5.

7. Verify that the files you downloaded are the same size as the corresponding files on Oracle Technology Network. Also verify the checksums are the same as noted on Oracle Technology Network using a command similar to the following:

   cksum *filename*.zip

8. Extract the files in each directory that you just created.

9. After you have extracted the required installation files, see "Installing the Oracle Database Software" on page 7-8.

Downloading the Software from Oracle Software Delivery Cloud Portal

You can download the software from Oracle Software Delivery Cloud as Media Packs. A Media Pack is an electronic version of the software that is also available to Oracle customers on CD-ROM or DVD. To download the Media Pack:

1. Use any browser to access the Oracle Software Delivery Cloud portal:

   https://edelivery.oracle.com/

2. Complete the Export Validation process by entering information (name, company, email address, and country) in the online form.

3. In the Media Pack Search page, specify the Product Pack and Platform to identify the Media Pack you want to download. If you do not know the name of the Product Pack, you can search for it using the License List.

4. Optionally, select the relevant product to download from the **Results** list.

5. In the search results page, click **Readme** to download and review the Readme file for download instructions and product information.

6. After you review the Readme, choose the appropriate Media Pack from the search results to download the individual zip files. Follow the Download Notes

instructions in this page. After you download and extract the contents of the required zip files, proceed with the installation of the software.

> **Note:** Print the page with the list of downloadable files. It contains a list of part numbers and their corresponding descriptions that you may refer during the installation process.

7. After you download the files, click **View Digest** to verify that the MD5 or SHA-1 checksum matches with what is listed in the media download page.

> **See Also:**
>
> - My Oracle Support note 549617.1 for information on how to verify the integrity of a software download at:
>
>   https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=549617.1
>
> - *Frequently Asked Questions* section on the Oracle Software Delivery Cloud portal for more information about Media Packs

Extracting the Installation Files

To extract the installation archive files, perform the following steps:

1. If necessary, change to the directory that contains the downloaded installation archive files.

2. Oracle RDBMS software is available as two archive files. Ensure that you extract both the archive files to the same directory.

If the downloaded file has the `.zip` extension, use the following command to extract the content:

```
unzip file_name.zip
```

If the downloaded file has the `cpio.gz` extension, use the following command:

```
$ gunzip filename.cpio.gz
```

This command creates files with names similar to the following:

```
filename.cpio
```

To extract the installation files, enter a command similar to the following:

```
$ cpio -idcmv < filename.cpio
```

> **Note:** See the download page for information about the correct options to use with the `cpio` command.
>
> Some browsers uncompress files while downloading them, but they leave the `.gz` file extension.

For each file, this command creates a subdirectory named `Diskn`, where *n* is the disk number identified in the file name.

When you have extracted all of the required installation files, see "Installing the Oracle Database Software" on page 7-8.

Copying the Software to the Hard Disk

Before installing Oracle Database, Oracle recommends that you copy the software to the hard disk to enable the installation process to run faster. Before copying the installation media content to the hard disk, you must mount the disk. The following sections describe how to mount the disk and copy its content to the hard disk.

Mounting Disks

On most systems, the disk mounts automatically when you insert it into the disk drive. If the disk does not mount automatically, then follow these steps to mount it:

1. If necessary, log in as the `root` user and enter a command similar to one of the following to eject the currently mounted disk, then remove it from the drive:

   - Oracle Linux and Red Hat Enterprise Linux:

     ```
     # eject /mnt/dvd
     ```

   - SUSE Linux Enterprise Server:

     ```
     # eject /media/dvd
     ```

   In these examples, `/mnt/dvd` and `/media/dvd` are the mount point directories for the installation media.

2. Insert the appropriate installation media into the disk drive.

3. To verify if the disk is mounted automatically, enter one of the following commands depending on the platform:

   - Oracle Linux and Red Hat Enterprise Linux:

     ```
     # ls /mnt/dvd
     ```

   - SUSE Linux Enterprise Server:

     ```
     # ls /media/dvd
     ```

4. Before running the following command, ensure that the `/mnt/dvd` directory exists on Red Hat Enterprise Linux. If not, create the `/mnt/dvd` as required, to mount the installation media.

   If this command fails to display the contents of the installation media, enter a command similar to the following to mount it, depending on the platform:

   - Oracle Linux and Red Hat Enterprise Linux:

     ```
     # mount -t iso9660 /dev/dvd /mnt/dvd
     ```

   - SUSE Linux Enterprise Server:

     ```
     # mount -t iso9660 /dev/dvd /media/dvd
     ```

   In these examples, `/mnt/dvd` and `/media/dvd` are the mount point directories for the installation media.

5. If Oracle Universal Installer displays the Disk Location dialog box, enter the disk mount point directory path, for example:

   ```
   /mnt/dvd
   ```

To continue, go to one of the following sections:

- To copy software to a hard disk, see "Copying the Oracle Database Software to a Hard Disk" on page 7-8.

- To install the software from the installation media, see "Installing the Oracle Database Software" on page 7-8.

Copying the Oracle Database Software to a Hard Disk

> **Note:** If the system does not have an installation media, you can copy the software from the disk to a file system on another system, then either mount that file system using NFS or use FTP to copy the files to the system where you want to install the software.

To copy the contents of the installation media to a hard disk:

1. Create a directory on the hard disk, outside of the Oracle base directory, to hold the Oracle software:

   ```
   $ mkdir OraDb12c
   ```

2. Change the directory to the directory you created in Step 1:

   ```
   $ cd OraDb12c
   ```

3. Mount the disk, if it is not mounted.

   Some platforms automatically mount the disk when you insert it into the drive. If the disk does not mount automatically, see the "Mounting Disks" section on page 7-7 for platform-specific information.

4. Copy the contents of the mounted disk to the corresponding new subdirectory as follows:

   ```
   $ cp -R /directory_path OraDb12c
   ```

   In this example, /directory_path is the disk mount point directory.

5. If necessary, mount the next disk and repeat Step 4.

Installing the Oracle Database Software

In most cases, you use the graphical user interface (GUI) provided by Oracle Universal Installer to install Oracle Database. The instructions in the following sections explain how to run the Oracle Universal Installer GUI to perform most database installations.

> **Note:**
>
> - If you plan to use Oracle Restart or Oracle ASM, then you must install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart. For information about installing Oracle Grid Infrastructure for a standalone server, see "Installing and Configuring Oracle Grid Infrastructure for a Standalone Server" on page 6-14.
>
> - You may have to shut down existing Oracle processes before you start the database installation. See "Stopping Existing Oracle Processes" on page 5-12.
>
> - To install Oracle Database by using the silent or response file installation method, without the GUI, see Appendix A. This method is useful to perform multiple installations of Oracle Database. This appendix also describes other advanced installation topics.

Running Oracle Universal Installer

For any type of installation process, start Oracle Universal Installer and install the software, as follows:

1. Log on as the Oracle software owner user (typically, `oracle`) to the computer on which you want to install Oracle components.

2. If you are installing the software from installation media, mount the disk if it is not mounted.

   If the disk does not mount automatically, see the "Mounting Disks" section on page 7-7 for platform-specific information.

   Some platforms automatically mount the disk when you insert the installation media into the drive.

3. To start Oracle Universal Installer, complete one of the following steps depending on the location of the installation files:

   > **Note:** Start Oracle Universal Installer from the terminal session where you logged in as the `oracle` user and set the user's environment.

   > **See Also:** "Configuring Oracle Software Owner Environment" on page 5-13

   - If the installation files are on installation media, enter commands similar to the following, where *directory_path* is the path of the `database` directory on the installation media:

     ```
     $ /directory_path/runInstaller
     ```

   - If the installation files are on the hard disk, change the directory to the `database` directory and enter the following command:

     ```
     $ ./runInstaller
     ```

If Oracle Universal Installer is not displayed, see "X Window Display Errors" on page I-2 and "Remote Terminal Installation Error" on page I-2 for information about troubleshooting.

4. Use the following guidelines to complete the installation:

   ■ Do not install Oracle Database 12*c* software into an existing Oracle home.

   ■ Follow the instructions displayed on the Oracle Universal Installer screens. If you need additional information, click **Help**.

      See Also: "Reviewing Accounts and Passwords" on page 9-4 for information about password guidelines

   ■ Do not modify the Java Runtime Environment (JRE) except by using a patch provided by Oracle Support Services. Oracle Universal Installer automatically installs the Oracle-supplied version of the JRE. This version is required to run Oracle Universal Installer and several Oracle assistants.

   ■ If errors are displayed while installing the software, see Appendix I for information about troubleshooting.

   ■ If you chose an installation type that runs Oracle Database Configuration Assistant (Oracle DBCA) in interactive mode, then you must provide detailed information about configuring the database and network.

      If you need help when using the Oracle Database Configuration Assistant in interactive mode, click **Help** on any screen.

      Note: If you chose a default installation, Oracle Database Configuration Assistant does not run interactively.

      See Also: "Creating and Managing a Database with DBCA" in *Oracle Database 2 Day DBA*

5. When the configuration assistant tasks are complete click **finish**, click **Exit**, then click **Yes** to exit from Oracle Universal Installer.

6. During the database installation, when Oracle Universal Installer prompts you to run a script with `root` privileges, enter a command similar to the following in a terminal where you are logged in as the `root` user, then click **OK**:

   ```
   # /script_path/script_name
   ```

   For more information see the description in the Install Product screen in the table that follows.

7. See Chapter 8 for information about tasks that you must complete after you install Oracle Database.

The following table lists the various screens displayed during an Enterprise Edition installation for Oracle Database 12*c*:

Screen	Action
Configure Security Updates	Enter your email address, preferably your My Oracle Support email address or user name in the **Email** field.
	Select the **I wish to receive security updates via My Oracle Support** check box to receive security updates.
	Enter your My Oracle Support password in the **My Oracle Support Password** field.
	Click **Next.**
	See Also: "Database Security Notification Options" on page 2-7
Select Installation Option	Select one of the following installation options, and click **Next:**
	■ Create and configure a database: This option creates a new database with sample schemas.
	■ Install database software only: This option only installs the database binaries. To configure the database, you must run Oracle Database Configuration Assistant after the software installation.
	■ Upgrade an existing database: This option installs the software binaries in a new Oracle home. After the installation, you can upgrade the existing database.
System Class	Select the type of system for installing the database, and click **Next.**
	■ Desktop Class: Select this option if you are installing on a laptop or desktop class system. This option includes a starter database and enables a minimal configuration. This option is designed for those who want to get the database running quickly.
	See Also: "Setting the ORACLE_HOSTNAME Environment Variable" on page E-1
	■ Server Class: Select this option if you are installing on a server class system, such as what you would use when deploying Oracle Database in a production data center. This option provides more advanced configuration options. Advanced configuration options available using this option include Oracle RAC, Oracle ASM, backup and recovery configuration, integration with Oracle Enterprise Manager Cloud Control, and more fine-grained memory tuning, among many others.
Grid Installation Options	Select the type of database installation you want to perform, and click **Next.**
	■ Single instance database installation: This option installs the database and the listener.
	■ Oracle Real Application Clusters database installation: This option installs Oracle Real Application Clusters.
	■ Oracle RAC One Node database installation: This option installs the Oracle RAC One Node database.

Screen	Action
Select Install Type	Select one of the following, and click **Next:** ■ Typical Install: This installation method is selected by default. It lets you quickly install Oracle Database using minimal input. It installs the software and optionally creates a general-purpose database using the information that you specify on this screen. ■ Advanced Install: This installation method enables you to perform more complex installations, such as creating individual passwords for different accounts, creating specific types of starter databases (for example, for transaction processing or data warehouse systems), using different language groups, specifying email notifications, and so on.
Select Product Languages	This option enables you to select the language in which you want to run the product. Select the product language from the Available Languages list, and move it to the Selected Languages list. Click **Next.**
Select Database Edition	Select Enterprise Edition, Standard Edition, Standard Edition One, or Standard Edition 2. Click **Next.** **See Also:** "Enabling and Disabling Database Options" on page 8-5 for information about enabling and disabling components that are installed with Oracle Database. "Product-Specific Postinstallation Tasks" on page 8-6 for information about configuring components that are installed with Oracle Database.
Specify Installation Location	The Oracle base path appears by default. You can change this path based on your requirement. Specify **Oracle Base**, **Software Location**, and click **Next.** The Oracle base directory is a top-level directory for Oracle software installations owned by an Oracle installation owner account. The default Oracle base path is *mountpoint/app/user*, where *user* is the user account running the installation. You can change the path based on your requirements. In the **Software Location** field, accept the default value or enter the Oracle home directory path in which you want to install the Oracle software. The directory path must not contain spaces. Click **Next.** **Note**: This screen is available only with Advanced Installation. Ensure that the Oracle home path for the database home and the Oracle base path use only ASCII characters. At the time of this release, the use of non-ASCII characters for an Oracle database home or Oracle base is not supported. **See Also:** "Naming Directories" on page F-2 and "Identifying Required Software Directories" on page 4-34

Screen	Action
Create Inventory	This screen is displayed if this is the first time you are installing Oracle software on your system.
	You are prompted by Oracle Universal Installer to specify the **Inventory Directory** path for the central inventory the first time you install any Oracle software on your computer.
	Select the **oraInventory Group Name** of the operating system group that will own the Oracle Inventory directory (the Oracle Inventory group).
	Click **Next.**
	Note: By default, the Oracle Inventory directory is not installed under the Oracle Base directory. This is because all Oracle software installations share a common Oracle Inventory, so there is only one Oracle Inventory for all users, whereas there is a separate Oracle Base directory for each user.
Select Configuration Type	Select one of the following, and click **Next:**
	■ General Purpose / Transaction Processing: This is a starter database designed for general usage or transaction-heavy applications.
	■ Data Warehousing: A starter database optimized to run Data Warehousing applications.
	See the online help provided by either Oracle Universal Installer or Oracle Database Configuration Assistant for a description of these preconfigured database types.

Screen	Action
Specify Database Identifiers	Provide the following information, and click **Next:**
	Database Naming
	Provide the Global Database Name using the following syntax:
	`db_unique_name.db_domain`
	■ `db_unique_name` is the name of the database. It can contain a maximum of 30 characters if the first 8 characters are unique and begin with an alphabetic character. The characters can include alphanumeric, underscore (_), dollar sign ($), and pound sign (#), no other special characters are permitted in a database name.
	■ `db_domain` is the computer environment used for the database. It can contain no more than 128 characters (alphanumeric, underscore (_), and pound sign (#)), inclusive of all periods.
	Note: Ensure that the combination of database name (first 8 unique characters of the unique name for the database), delimiter, and the database domain name does not exceed 128 characters.
	For example:
	`sales.example.com`
	■ `db_unique_name` is `sales`
	■ `db_domain` is `example.com`
	When you enter the Global Database Name, Oracle Universal Installer automatically populates the SID prefix with the database name. You can change this name in Advanced Installation. Oracle Universal Installer limits the SID to 12 alphanumeric characters and the SID cannot contain an underscore (_), dollar sign ($), or pound sign (#).
	See "Setting the ORACLE_HOSTNAME Environment Variable" on page E-1 and "Identifying Databases" on page 9-8
	Select the **Create as Container database** option to create the database as a multitenant container database (CDB) that can support one pluggable database (PDB). If you want Oracle Universal Installer to create a PDB when it creates the CDB, specify the PDB name in the **Pluggable Database Name** field.
	The PDB name must be unique and must follow the database naming conventions. See "Identifying Databases" on page 9-8
	To create additional PDBs and to manage PDBs, use Oracle Database Configuration Assistant.
	See Also: *Oracle Database 2 Day DBA*

Screen	Action
Specify Configuration Options	Provide the following configuration information, and click **Next:** **Memory:** Select the **Enable Automatic Memory Management** option to allow the database to automatically distribute memory between SGA and PGA. If you do not select this option, then the SGA and PGA must be sized manually. **See Also:** "Consider Memory Allocation and Automatic Memory Management" on page 2-4 **Character Sets:** This option enables you to store the character data in the database in one of the following methods: ■ Use the default: This option uses the operating system language settings. ■ Use Unicode: This option enables you to store multiple language groups ■ Choose from the following list of character sets: This option enables the **Select Database Character Set** drop down list. **See Also:** ■ "Selecting the Database Character Set" on page 7-2 ■ *Oracle Database Globalization Support Guide* for information about choosing a character set **Sample Schemas:** The Create database with sample schemas option is not selected by default. You can select this option, to create a starter database with sample schemas. If you create the database as a CDB with one PDB, then the sample schema is created as a PDB. **Note**: By default, Oracle database is configured to include enhanced security settings.
Specify Database Storage Options	Select one of the following options, and click **Next.** ■ File System: Specify the database file location. ■ Oracle Automatic Storage Management. **See Also**: "Using an Existing Oracle Automatic Storage Management Disk Group" on page 7-3

Screen	Action
Specify Management Options	This screen gives you the option to manage your database using Oracle Enterprise Manager Cloud Control. Select **Register with Enterprise Manager (EM) Cloud Control** and specify the following for Oracle Enterprise Manager Cloud Control configuration, and click **Next:** ■ OMS Host: The system name where the Management repository is running. ■ OMS Port: The Oracle Enterprise Manager port number to receive requests from the Management service. ■ EM Admin Username: The user name to log in to Oracle Enterprise Manager. ■ EM Admin Password: The password to log in to Oracle Enterprise Manager. ■ Specify password of ASMSNMP user: The password for the ASMSNMP user configured in Oracle ASM, required only if you choose Oracle ASM as your database storage option. **Note:** Oracle Enterprise Manager Database Express is always installed and configured by default irrespective of whether you register Oracle Enterprise Manager Cloud Control.
Specify Recovery Options	Select **Enable Recovery** to enable recovery using one of the following options: ■ Select **File System** to use a file system directory for the fast recovery area, and then specify the fast recovery area path in the **Recovery Area location** field. ■ Select **Oracle Automatic Storage Management** to use an Automatic Storage Management disk group for the fast recovery area. Click **Next.** **See Also**: "Oracle Automatic Storage Management Installation Considerations" on page 6-7 and "Database Backup and Recovery Options" on page 2-14.
Select ASM Disk Group	This screen is displayed only if you select Oracle Automatic Storage Management as your storage option in the Specify Storage Option screen. Disk groups are created during the Oracle Grid Infrastructure installation. Disk groups are configured with the SYSASM privilege using asmcmd or SQL create diskgroup commands. An ASM disk group consists of multiple disk partitions. The table in this screen displays existing disk groups created during the Oracle Grid Infrastructure installation. Select the disk group to use for database file storage.
Specify Schema Passwords	Enter and confirm passwords for the privileged database accounts: SYS, SYSTEM, and DBSNMP. If you chose to create the database as a CDB, then Oracle Universal Installer also asks for the PDBADMIN password. Click **Next.** **Note:** Optionally, you can use the same password for all accounts. However, Oracle recommends that you specify a different password for each account. You must remember the passwords that you specify.

Screen	Action
Privileged Operating System Groups	The operating system groups are selected by default. You can also manually select the `OSDBA` and `OSOPER` groups.
	Select the `OSBACKUPDBA`, `OSDGDBA`, and `OSKMDBA` groups.
	Click **Next.**
	See Also: "Creating Required Operating System Groups and Users" on page 5-1
Perform Prerequisite Checks	This option verifies that the minimum system requirements to perform the database installation are met.
	If you click **Check Again,** then you can run the prerequisite check again to see if the minimum requirements are met to carry on with the database installation.
	Click **Fix & Check Again,** if you want the installer to fix the problem and verify the system requirements again.
	Note: The **Fix & Check Again** option generates a script that you must run as the `root` user. This generated script sets some system parameters to Oracle-recommended values. Oracle recommends that you do not modify the contents of this script.
	See Also: "Using Installation Fixup Scripts" on page 4-10 for more information.
	To get a list of failed requirements, select **ShowFailed** from the list. To get a list of all the prerequirement checks run by Oracle Universal Installer, select **Show All**. To get a list of the prerequirement checks that are successful, select **Show Succeeded.**
	Note: Oracle recommends that you use caution when selecting the **Ignore All** option. If you select this option, then Oracle Universal Installer may not confirm that your system can install Oracle Database successfully.
	See Also: Chapter 4, "Oracle Database Preinstallation Tasks"
Summary	Review the information displayed on this screen, and click **Install.**
	Note: You can save all the installation steps into a response file by clicking **Save Response File.** Later, this file can be used for a silent installation.
Install Product	This screen displays the progress of a database installation. During this process, the Execute Configuration Scripts window appears. Do not click OK until you run the `root.sh` and, if required, the `orainstRoot.sh` configuration scripts mentioned in this screen as the `root` user. Click **Next.**
	This screen then displays the status information for the configuration assistants that configure the software and create a database.
	A message is displayed at the end of the Database Configuration Assistant process. Review the database information, specially the Oracle Enterprise Manager Database Express URL, and click **OK.**
	Note: If this is the first time you are installing Oracle software on your system, then Oracle Universal Installer prompts you to run the `orainstRoot.sh` script.
	See Also: *Oracle Database 2 Day DBA* for information about Oracle Database Configuration Assistant

Screen	Action
Finish	This screen is shown automatically when all the configuration tools are successful.
	Click **Close.**

> **Caution:** After installation is complete, do not manually remove or run `cron` jobs that remove `/tmp/.oracle` or `/var/tmp/.oracle` directories or their files while Oracle software is running. If you remove these files, then the Oracle software can intermittently stop responding. Oracle Restart installations fail with the following error:
>
> ```
> CRS-0184: Cannot communicate with the CRS daemon.
> ```

Installing Oracle Database Examples

If you plan to use the following products or features, then download and install the products from the Oracle Database Examples media:

- Oracle JDBC Development Drivers

- Oracle Database Examples

- Oracle Text Knowledge Base

- Various Oracle product demonstrations

For information about installing software and various Oracle product demonstrations from the Oracle Database Examples media, see *Oracle Database Examples Installation Guide*.

8

Oracle Database Postinstallation Tasks

This chapter describes tasks that you must perform after you install the database software. It includes information about the following topics:

- Creating a Database
- Required Postinstallation Tasks
- Recommended Postinstallation Tasks
- Product-Specific Postinstallation Tasks
- Postinstallation Tasks for SQL Developer

You must perform the tasks listed in "Required Postinstallation Tasks" on page 8-1. Oracle recommends that you perform the tasks listed in "Recommended Postinstallation Tasks" on page 8-2 after all installations.

If you install and intend to use any of the products listed in "Product-Specific Postinstallation Tasks" on page 8-6, then you must perform the tasks listed in the product-specific sections.

> **Note:** This chapter describes basic configuration only. See *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems*, *Oracle Database Administrator's Guide*, and product-specific administration and tuning guides for more detailed configuration and tuning information.

> **See Also:** "Post-installation Database Configuration" section in *Oracle Configuration Manager Installation and Administration Guide*

Creating a Database

If you did not create a database during the database installation, then use Oracle Database Configuration Assistant (Oracle DBCA) to create a database after the database installation.

> **See Also:** *Oracle Database 2 Day DBA* for information about creating a database using Oracle DBCA

Required Postinstallation Tasks

Perform the following task after completing Oracle Database installation.

Downloading and Installing Patches

Check the My Oracle Support website for required patch updates for your installation.

To download required patches:

1. Use a web browser to view the My Oracle Support website:

   https://support.oracle.com/

2. Log in to My Oracle Support.

   > **Note:** If you are not a My Oracle Support registered user, click **Register here** and follow the registration instructions.

3. On the main My Oracle Support page, click the **Patches & Updates** tab.

4. In the Patch Search group, select **Product or Family (Advanced).**

5. In the Product field, select **Oracle Database.**

6. In the Release field select one or more release numbers. For example, Oracle 12.1.0.1.0.

7. Click **Search.**

8. Any available patch updates are displayed in the Patch Search page.

9. Select the patch number and click **ReadMe.** The README page is displayed and contains information about the patch set and how to apply the patches to your installation.

10. Return to the Patch Search page, click **Download**, and save the file on your system.

11. Use the unzip utility provided with Oracle Database 12*c* to uncompress the Oracle patch updates that you downloaded from My Oracle Support. The unzip utility is located in the $ORACLE_HOME/bin directory.

Recommended Postinstallation Tasks

Oracle recommends that you perform the tasks described in the following section after completing an installation:

- Creating a Backup of the root.sh Script
- Creating and Configuring Additional Operating System Accounts
- Setting Language Preferences for Client Connections
- Guidelines for Setting Semaphore Parameters
- Creating a Fast Recovery Area Disk Group
- Enabling and Disabling Database Options
- Downloading and Installing the ORAchk Health Check Tool

Creating a Backup of the root.sh Script

Oracle recommends that you back up the root.sh script after you complete an installation. If you install other products in the same Oracle home directory, then Oracle Universal Installer updates the contents of the existing root.sh script during

the installation. If you require information contained in the original `root.sh` script, then you can recover it from the backed up `root.sh` file.

Creating and Configuring Additional Operating System Accounts

If required, create additional operating system accounts. Users must be members of the OSDBA or OSOPER groups to connect to the database with administrator privileges.

Configuring the Accounts of Oracle Users

Update the startup files of the `oracle` user and the operating system accounts of Oracle users, specifying the appropriate environment variables in the environment file.

For the Bourne, Bash, or Korn shell, add the environment variables to the profile file for the Bash shell on Oracle Linux and Red Hat Enterprise Linux.

For the C shell, add the environment variables to the `.login` file.

> **Note:** You can use the `oraenv` or `coraenv` script to ensure that Oracle user accounts are updated.

Setting Language Preferences for Client Connections

Set the language preferences for clients connecting to the database by setting the `NLS_LANG` environment variable.

The `NLS_LANG` environment variable declares the language and territory character sets that client applications use to connect to the database. Oracle Database uses that character set for data entered or displayed by Oracle client programs, such as SQL*PLus.

> **See Also:** Appendix H, "Configuring Oracle Database Globalization Support" for more information about the `NLS_LANG` environment variable

Guidelines for Setting Semaphore Parameters

Use the following guidelines only if the default semaphore parameter values are too low to accommodate all Oracle processes:

> **Note:** Oracle recommends that you see the operating system documentation for more information about setting semaphore parameters.

1. Calculate the minimum total semaphore requirements using the following formula:

   ```
   sum (process parameters of all database instances on the system) + overhead for
   oracle background processes + system and other application requirements
   ```

2. Set `semmns` (total semaphores systemwide) to this total.

3. Set `semmsl` (semaphores per set) to 250.

4. Set `semmni` (total semaphores sets) to `semmns/semmsl` rounded up to the nearest multiple of 1024.

> **See Also:** My Oracle Support note 226209.1, "Linux: How to Check Current Shared Memory, Semaphore Values," at the following URL:
>
> https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=226209.1

Creating a Fast Recovery Area Disk Group

During installation, by default you can create one disk group. If you plan to add an Oracle Database for a standalone server, then you should create the fast recovery area for database files.

About the Fast Recovery Area and the Fast Recovery Area Disk Group

The fast recovery area is a unified storage location for all Oracle Database files related to recovery. Database administrators can define the DB_RECOVERY_FILE_DEST parameter to the path for the fast recovery area to enable on disk backups and rapid recovery of data. Enabling rapid backups for recent data can reduce requests to system administrators to retrieve backup tapes for recovery operations.

When you enable fast recovery in the init.ora file, Oracle Database writes all RMAN backups, archive logs, control file automatic backups, and database copies to the fast recovery area. RMAN automatically manages files in the fast recovery area by deleting obsolete backups and archiving files no longer required for recovery.

Oracle recommends that you create a fast recovery area disk group. Oracle Clusterware files and Oracle Database files can be placed on the same disk group, and you can also place fast recovery files in the same disk group. However, Oracle recommends that you create a separate fast recovery disk group to reduce storage device contention.

The fast recovery area is enabled by setting the DB_RECOVERY_FILE_DEST parameter. The size of the fast recovery area is set with DB_RECOVERY_FILE_DEST_SIZE. As a general rule, the larger the fast recovery area, the more useful it becomes. For ease of use, Oracle recommends that you create a fast recovery area disk group on storage devices that can contain at least three days of recovery information. Ideally, the fast recovery area is large enough to hold a copy of all of your data files and control files, the online redo logs, and the archived redo log files needed to recover your database using the data file backups kept under your retention policy.

Multiple databases can use the same fast recovery area. For example, assume you have created a fast recovery area disk group on disks with 150 GB of storage, shared by 3 different databases. You can set the size of the fast recovery for each database depending on the importance of each database. For example, if database1 is your least important database, database2 is of greater importance and database3 is of greatest importance, then you can set different DB_RECOVERY_FILE_DEST_SIZE settings for each database to meet your retention target for each database: 30 GB for database1, 50 GB for database2, and 70 GB for database3.

Creating the Fast Recovery Area Disk Group

To create a fast recovery file disk group:

1. Go to the Grid home bin directory, and start ASM Configuration Assistant (ASMCA), for example:

    ```
    $ cd /u01/grid/bin
    ```

```
$ ./asmca
```

2. ASMCA opens at the Disk Groups tab. Click **Create** to create a new disk group.

3. The Create Disk Groups window opens.

   In the **Disk Group Name** field, enter a descriptive name for the fast recovery area group. For example: FRA.

   In the Redundancy section, select the level of redundancy you want to use.

   In the **Select Member Disks** field, select eligible disks to be added to the fast recovery area, and click **OK**.

4. The Diskgroup Creation window opens to inform you when disk group creation is complete. Click **OK**.

5. Click **Exit**.

> **See Also:**
>
> - "Setting the Fast Recovery Area Location and Initial Size" section in *Oracle Database Backup and Recovery User's Guide*
> - *Oracle Automatic Storage Management Administrator's Guide*

Enabling and Disabling Database Options

When you install Oracle Database, some options are enabled and others are disabled. To enable or disable a particular database feature for an Oracle home, shut down the database and use the chopt tool. See Example 8–1.

The chopt tool is a command-line utility that is located in the `ORACLE_HOME/`bin directory. The syntax for chopt is:

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

The possible values for `db_option` are described in the following table:

Value	Description
dm	Oracle Data Mining RDBMS Files
olap	Oracle OLAP
partitioning	Oracle Partitioning
rat	Oracle Real Application Testing

Example 8–1   Complete Example of Running the Chopt Tool

To enable the Oracle Data Mining option in your Oracle binary files, use the following command:

```
cd $ORACLE_HOME/bin
srvctl stop database -d myDb
chopt enable dm
srvctl start database -d myDb
```

Downloading and Installing the ORAchk Health Check Tool

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

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

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

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

Oracle recommends that you download and run the latest version of ORAchk from My Oracle Support. For information about downloading, configuring and running ORAchk utility, refer to My Oracle Support note 1268927.1:

https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1268927.1

Product-Specific Postinstallation Tasks

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

- Configuring Oracle Net Services
- Configuring Oracle Label Security
- Configuring Oracle Database Vault
- Configuring Oracle Messaging Gateway
- Configuring Oracle Precompilers
- Configuring Secure Sockets Layer
- Installing Oracle Text Supplied Knowledge Bases
- Configuring or Reinstalling Oracle XML DB
- Configuring New or Upgraded Databases
- Configuring Direct NFS Client

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

> **Note:** Perform postinstallation tasks only for products that you intend to use.

Configuring Oracle Net Services

If you have an earlier release of Oracle software installed on this system, then you can copy information from the Oracle Net `tnsnames.ora` and `listener.ora` configuration files from the earlier release to the corresponding files for the new release.

The following sections describe about how to configure the Oracle Net Services:

- Modifying the listener.ora File
- Modifying the tnsnames.ora File

> **Note:** The default location for the `tnsnames.ora` and `listener.ora` files is the `$ORACLE_HOME/network/admin/` directory. However you can relocate these files and set the `TNS_ADMIN` environment variable to the directory containing them.

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 listener instead of the listener from the previous release.

If you have referenced the previous Oracle home directory names in the static listener information, then these directory names must be modified before the `listener.ora` file can be used in the 12.1 environment.

To use the listener from the current release, copy the static service information from the `listener.ora` file from the previous release to the version of that file used by the new release.

For any database instances earlier than release 8.0.3, add static service information to the `listener.ora` file. Oracle Database releases later than release 8.0.3 do not require static service information.

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

Modifying the tnsnames.ora File

Unless you are using a central `tnsnames.ora` file, copy Oracle Net Services names and connect descriptors from the earlier 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.

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

Configuring Oracle Label Security

You must configure Oracle Label Security in a database to use it. See "Oracle Label Security Using Oracle Internet Directory" in *Oracle Label Security Administrator's Guide* for more information.

Configuring Oracle Database Vault

Oracle Database includes Database Vault by default, but you must register it before you can use it. Ensure that you create the Database Vault Owner and, optionally, the Database Vault Account Manager administrative accounts before you can use them.

Oracle Database Vault installs a baseline database auditing policy. This policy covers the access control configuration information stored in Oracle Database Vault database tables, information stored in Oracle Catalog (rollback segments, tablespaces, and so on), the use of system privileges, and Oracle Label Security configuration. When you install Oracle Database Vault, the security specific database initialization parameters are initialized with default values.

For information on Oracle Database Vault policy audit events, see *Oracle Database Vault Administrator's Guide*.

To register Oracle Database Vault, see *Oracle Database Vault Administrator's Guide*.

If you plan to use Oracle Data Guard with Oracle Database Vault, then see *Oracle Database Vault Administrator's Guide*.

Configuring Oracle Messaging Gateway

Oracle Messaging Gateway, an Oracle Database Advanced Queuing feature, requires additional configuration after you install Oracle Database if you plan to use Oracle Database Advanced Queuing.

> **See Also:** *Oracle Database Advanced Queuing User's Guide* to configure Oracle Messaging Gateway and for additional instructions about configuring the `listener.ora`, `tnsnames.ora`, and `mgw.ora` files

Configuring Oracle Precompilers

This section describes postinstallation tasks for Oracle precompilers:

- Configuring Pro*C/C++
- Configuring Pro*FORTRAN

> **Note:** All precompiler configuration files are located in the `$ORACLE_HOME/precomp/admin` directory.

Configuring Pro*C/C++

Verify that the `PATH` environment variable setting includes the directory that contains the C compiler executable.

The following table shows the default directories and the appropriate command to verify the path setting of the compiler.

Table 8–1   C/C++ Compiler Directory

Path	Command
/usr/bin	$ which gcc
/opt/intel_cce_80/bin/icc	$ which icc

Configuring Pro*FORTRAN

Verify that the PATH environment variable setting includes the directory that contains the FORTRAN compiler executable.

The following table shows the default directories and the appropriate command to verify the path setting of the compiler.

Table 8–2   FORTRAN Compiler Directory

Path	Command
/usr/bin	$ which xlf

Configuring Secure Sockets Layer

Oracle recommends that you configure and use a Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.

> **See Also:** *Oracle Database Security Guide* for information about Secure Socket Layer

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, then you can install two supplied knowledge bases (English and French).

> **See Also:**
>
> - *Oracle Database Examples Installation Guide*
> - *Oracle Text Reference* for information about creating and extending knowledge bases, such as extending the supplied knowledge bases to accommodate your requirements, or creating your own knowledge bases in languages other than English and French

Configuring or Reinstalling Oracle XML DB

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

> **See Also:**   The "Using FTP on the Standard Port Instead of the Oracle XML DB Default Port" and "Using HTTP(S) on a Standard Port Instead of an Oracle XML DB Default Port" sections in *Oracle XML DB Developer's Guide* for configuring the Oracle XML DB ports

Also, see *Oracle XML DB Developer's Guide* for more information about the following tasks:

- Reinstalling Oracle XML DB
- Configuring or customizing the Oracle XML DB tablespace
- Configuring FTP, HTTP/WebDAV port numbers

Configuring New or Upgraded Databases

You must run the utlrp.sql script after creating or upgrading a database. This script recompiles all PL/SQL modules that are in an invalid state, including packages, procedures, and types. You must run the utlrp.sql script immediately following the installation and not at a later date.

> **See Also:** *Oracle Database Upgrade Guide*

1. Switch the user to `oracle`.

2. Use the `oraenv` or `coraenv` script to set the environment for the database where you want to run the `utlrp.sql` script:

   - Bourne, Bash, or Korn shell:

     ```
     $ . /usr/local/bin/oraenv
     ```

   - C shell:

     ```
     % source /usr/local/bin/coraenv
     ```

   When prompted, provide the `SID` for the database.

3. Start SQL*Plus, as follows:

   ```
   $ sqlplus / AS SYSDBA
   ```

4. Start the database in restricted mode and run the `utlrp.sql` script:

   ```
   SQL> STARTUP RESTRICT
   SQL> @$ORACLE_HOME/rdbms/admin/utlrp.sql
   ```

Configuring Direct NFS Client

Direct NFS Client is an alternative to using kernel-managed NFS. Refer to the following sections to configure Direct NFS Client:

- About Direct NFS Client Configuration

- About the oranfstab File and Direct NFS Client

- Mounting NFS Storage Devices with Direct NFS Client

- Checking NFS Buffer Size Parameters

- Setting TCP Network Protocol Buffer for Direct NFS Client

- Specifying Network Paths with the oranfstab File

- Enabling Direct NFS Client

- Disabling Direct NFS Client

- Enabling HCC on Direct NFS Client

About Direct NFS Client Configuration

With Oracle Database, instead of using the operating system kernel NFS client, you can configure Oracle Database to access NFS servers directly using an Oracle internal Direct NFS Client. The Direct NFS Client supports NFSv3, NFSv4 and NFSv4.1 protocols (excluding the Parallel NFS extension) to access the NFS server.

Direct NFS Client supports up to four network paths to the NFS server. The Direct NFS Client performs load balancing across all specified paths. If a specified path fails, then Direct NFS Client reissues I/O commands over any remaining paths.

Some NFS file servers require NFS clients to connect using reserved ports. If your filer is running with reserved port checking, then you must disable it for Direct NFS Client to operate. To disable reserved port checking, see your NFS file server documentation.

For NFS servers that restrict port range, you can use the `insecure` option to enable clients other than `root` to connect to the NFS server. Alternatively, you can disable Direct NFS Client as described in "Disabling Direct NFS Client" on page 8-14.

> **See Also:** Use NFS servers supported for Oracle Database. See the
> My Oracle Support website for support information:
>
> https://support.oracle.com

About the oranfstab File and Direct NFS Client

Direct NFS Client uses either the configuration file `$ORACLE_HOME/dbs/oranfstab` or
the operating system mount tab file `/etc/mtab` to find out what mount points are
available. If `oranfstab` is not present, then by default Direct NFS Client servers mount
entries found in `/etc/mtab`. No other configuration is required. You can use `oranfstab`
to specify additional Oracle Database specific options to Direct NFS Client. For
example, you can use `oranfstab` to specify additional paths for a mount point.

You can add a new `oranfstab` file specifically for Oracle Database, either in the path
`/etc` or `$ORACLE_HOME/dbs`. When `oranfstab` is placed in `$ORACLE_HOME/dbs`, its entries
are specific to a single database. However, when `oranfstab` is placed in `/etc`, then it is
global to all Oracle databases, and can contain mount points for all Oracle databases.

> **Caution:** Direct NFS Client cannot serve an NFS server with write
> size values (`wtmax`) less than 32768.

Mounting NFS Storage Devices with Direct NFS Client

Direct NFS Client determines mount point settings to NFS storage devices based on
the configurations in `/etc/mtab`. Direct NFS Client searches for the mount entries in
the following order:

1. `$ORACLE_HOME/dbs/oranfstab`

2. `/etc/oranfstab`

3. `/etc/mtab`

Direct NFS Client uses the first matched entry as the mount point.

Oracle Database requires that mount points be mounted by the kernel NFS system
even when served through Direct NFS Client.

If Oracle Database cannot open an NFS server using Direct NFS Client, then Oracle
Database uses the platform operating system kernel NFS client. In this case, the kernel
NFS mount options must be set up as defined in "Checking NFS Buffer Size
Parameters" on page 8-11. Additionally, an informational message is logged in the
Oracle alert and trace files indicating that Direct NFS Client could not be established.

The Oracle database files available on the NFS server that are served by the Direct NFS
Client are also accessible through the operating system kernel NFS client. The usual
considerations for maintaining integrity of Oracle database files apply in this situation.

> **See Also:** *Oracle Automatic Storage Management Administrator's Guide*
> for guidelines to follow regarding managing Oracle database data files
> created with Direct NFS Client or kernel NFS

Checking NFS Buffer Size Parameters

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

Direct NFS Client issues writes at `wtmax` granularity to the NFS server.

For example, to use `rsize` and `wsize` buffer settings with the value 32768, update the `/etc/fstab` file on each node with an entry similar to the following:

```
nfs_server:/vol/DATA/oradata /home/oracle/netapp nfs\
rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,vers=3,timeo=600
```

> **See Also:** My Oracle Support Note 359515.1 for updated NAS mount option information:
>
> https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=359515.1

Setting TCP Network Protocol Buffer for Direct NFS Client

By default, the network buffer size is set to 1 MB for TCP, and 2 MB for UDP. The TCP buffer size can set a limit on file transfers, which can negatively affect performance for Direct NFS Client users.

To check the current TCP buffer size, enter the following command:

```
# sysctl -a |grep -e net.ipv4.tcp_[rw]mem
```

The output of this command is similar to the following:

```
net.ipv4.tcp_rmem = 4096       87380    1048576
net.ipv4.tcp_wmem = 4096       16384    1048576
```

Oracle recommends that you set the value based on the link speed of your servers. For example, perform the following steps:

1. As `root`, use a text editor to open `/etc/sysctl.conf`, and add or change to the following:

   ```
   # net.ipv4.tcp_rmem = 4096       87380    4194304
   # net.ipv4.tcp_wmem = 4096       16384    4194304
   ```

2. Restart the network:

   ```
   # /etc/rc.d/init.d/network restart
   ```

Specifying Network Paths with the oranfstab File

Direct NFS Client can use up to four network paths defined in the `oranfstab` file for an NFS server. The Direct NFS Client performs load balancing across all specified paths. If a specified path fails, then Direct NFS Client reissues I/O commands over any remaining paths.

Use the following SQL*Plus views for managing Direct NFS Client in a single instance environment:

- **v$dnfs_servers**: Shows a table of servers accessed using Direct NFS Client.

- **v$dnfs_files**: Shows a table of files currently open using Direct NFS Client.

- **v$dnfs_channels**: Shows a table of open network paths (or channels) to servers for which Direct NFS Client is providing files.

- **v$dnfs_stats**: Shows a table of performance statistics for Direct NFS Client.

Enabling Direct NFS Client

Complete the following procedure to enable Direct NFS Client:

1. Create an `oranfstab` file with the following attributes for each NFS server to be accessed using Direct NFS Client:

   - **Server**: The NFS server name.

   - **Path**: Up to four network paths to the NFS server, specified either by IP address, or by name, as displayed using the `ifconfig` command on the filer.

   - **Local**: Up to four local paths on the database host, specified by IP address or by name, as displayed using the `ifconfig` command run on the database host.

   - **Export**: The exported path from the NFS server.

   - **Mount**: The corresponding local mount point for the exported volume.

   - **Dontroute**: Specifies that outgoing messages should not be routed by the operating system, but sent using the IP address to which they are bound. Note that this POSIX option sometimes does not work on Linux systems with multiple paths in the same subnet.

   - **mnt_timeout**: Specifies (in seconds) the time the Direct NFS Client should wait for a successful mount before timing out. This parameter is optional and the default timeout is 10 minutes (600).

   - **nfs_version**: Specifies the NFS protocol version that the Direct NFS Client uses. Possible values are `NFSv3`, `NFSv4` and `NFSv4.1`. The default version is `NFSv3`. If you want to specify `NFSv4.x`, then you must set the `nfs_version` parameter accordingly in the `oranfstab` file.

   - **management**: Enables Direct NFS Client to use the management interface for SNMP queries. You can use this parameter if SNMP is running on separate management interfaces on the NFS server. The default value is the `server` parameter value.

   - **community**: Specifies the community string for use in SNMP queries. Default value is `public`.

     See Also: "Limiting Asynchronous I/O in NFS Server Environments" in *Oracle Database Performance Tuning Guide*

Example 8–2   oranfstab file with two NFS server entries

```
server: MyDataServer1
local: 192.0.2.0
path: 192.0.2.1
local: 192.0.100.0
path: 192.0.100.1
nfs_version: nfsv3
dontroute
export: /vol/oradata1 mount: /mnt/oradata1

server: MyDataServer2
local: LocalPath1
path: NfsPath1
local: LocalPath2
path: NfsPath2
local: LocalPath3
path: NfsPath3
local: LocalPath4
path: NfsPath4
nfs_version: nfsv4
dontroute
export: /vol/oradata2 mount: /mnt/oradata2
```

```
export: /vol/oradata3 mount: /mnt/oradata3
export: /vol/oradata4 mount: /mnt/oradata4
export: /vol/oradata5 mount: /mnt/oradata5
management: MgmtPath1
community: private
```

2. By default, Direct NFS Client is installed in a disabled state with single instance Oracle Database installations. To enable Direct NFS Client, complete the following steps:

   a. Change the directory to `$ORACLE_HOME/rdbms/lib`.

   b. Enter the following command:

      ```
      make -f ins_rdbms.mk dnfs_on
      ```

Disabling Direct NFS Client

Complete the following steps to disable Direct NFS Client:

1. Log in as the Oracle software installation owner, and disable Direct NFS Client using the following commands:

   ```
   cd $ORACLE_HOME/rdbms/lib
   make -f ins_rdbms.mk dnfs_off
   ```

2. Remove the `oranfstab` file.

   > **Note:** If you remove an NFS path that an Oracle Database is using, then you must restart the database for the change to take effect.

Enabling HCC on Direct NFS Client

To enable Hybrid Columnar Compression (HCC) on Direct NFS Client, perform the following steps:

1. Ensure that SNMP is enabled on the ZFS storage server. For example:

   ```
   $ snmpget -v1 -c public server_name .1.3.6.1.4.1.42.2.225.1.4.2.0
   SNMPv2-SMI::enterprises.42.2.225.1.4.2.0 = STRING: "Sun Storage 7410"
   ```

2. If SNMP is enabled on an interface other than the NFS server, then configure `oranfstab` using the `management` parameter.

3. If SNMP is configured using a community string other than `public`, then configure `oranfstab` file using the `community` parameter.

4. Ensure that `libnetsnmp.so` is installed by checking if `snmpget` is available.

Postinstallation Tasks for SQL Developer

See the following sections in *Oracle SQL Developer Installation Guide* for recommended postinstallation tasks for SQL Developer:

- Migrating User Settings from a Previous Release

- Migrating Information from Previous Releases

- Location of User-Related Information

9

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:

- Checking the Installed Oracle Database Contents and Directory Location
- Logging In to Oracle Enterprise Manager Database Express
- Managing Oracle Automatic Storage Management
- Accessing Oracle Database with SQL*Plus
- Accessing Oracle Database with SQL Developer
- Reviewing Accounts and Passwords
- Unlocking and Resetting User Passwords
- Identifying Databases
- Locating the Server Parameter File
- Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files

Checking the Installed Oracle Database Contents and Directory Location

You can use Oracle Universal Installer to check the contents and directory location of an Oracle Database installation using the following steps:

1. To start Oracle Universal Installer, run the following command:

   ```
   $ $ORACLE_HOME/oui/bin/runInstaller
   ```

2. Click **Installed Products** to display the Inventory dialog box on the Welcome screen.

3. Select the Oracle Database product from the list to check the installed contents.

4. Click **Details** to find additional information about an installed product.

5. Click **Close** to close the Inventory dialog box.

6. Click **Cancel** to close Oracle Universal Installer, and then click **Yes** to confirm.

Logging In to Oracle Enterprise Manager Database Express

To start Oracle Enterprise Manager Database Express, use the EM Express URL provided by Oracle Database Configuration Assistant (Oracle DBCA) during the database installation and creation. For information about logging in to Oracle

Enterprise Manager Database Express see "Starting EM Express" in *Oracle Database 2 Day DBA* and "Accessing the Database Home Page" in *Oracle Database 2 Day DBA*

If Oracle DBCA did not provide the EM Express URL during the database installation and creation, or if you need to change the EM Express port later on, then see "Configuring the HTTP Port for EM Express" in *Oracle Database 2 Day DBA*.

Managing Oracle Automatic Storage Management

This section provides information about managing an Oracle Automatic Storage Management (Oracle ASM) installation. It covers the following topics:

- Starting and Stopping Oracle Automatic Storage Management
- Oracle Automatic Storage Management Utilities

Starting and Stopping Oracle Automatic Storage Management

To start Oracle ASM, and stop Oracle ASM, see "Starting Up an Oracle ASM Instance" and "Shutting Down an Oracle ASM Instance" in *Oracle Automatic Storage Management Administrator's Guide*.

Oracle Automatic Storage Management Utilities

To manage Oracle ASM, you can use the following tools:

- **asmcmd**: This command-line tool enables you to manage Oracle ASM disk group files and directories.

- **ASMCA**: Oracle Automatic Storage Management Configuration Assistant (Oracle ASMCA) is an interactive utility that enables you to create an Oracle ASM instance or upgrade existing Oracle ASM instances.

  It also enables you to create and configure disk groups, Oracle ASM volumes and Oracle Automatic Storage Management Cluster File System (Oracle ACFS).

- **Oracle Enterprise Manager Cloud Control**: If you have Oracle Enterprise Manager installed, you can use Cloud Control to manage Oracle ASM functions, such as migrating an existing database to Oracle ASM, checking the status of the Oracle ASM instance, checking the performance of the Oracle ASM disk groups, and creating or dropping Oracle ASM disk groups.

- **Oracle Enterprise Manager Database Express 12***c*: This utility enables you to perform basic administrative tasks such as user, performance, memory, and space management.

- **SQL*Plus**: You can run commands that are specific to Oracle ASM from either of these tools. To connect to an Oracle ASM instance, use the same methods that you use to connect to an Oracle database instance.

  See Also:

  - *Oracle Database Utilities* for more information about `asmcmd`
  - *Oracle Automatic Storage Management Administrator's Guide* for more information about managing your storage with Oracle ASM

Accessing Oracle Database with SQL*Plus

To run the SQL and PL/SQL statements to access Oracle Database, you can use SQL*Plus. This tool enables you to perform the same database management operations, and also to query, insert, update, or delete data directly in the database.

> **Note:**
>
> - Before you start SQL*Plus, ensure that all the environment variables, specially `ORACLE_HOME` and `ORACLE_SID`, are set. See, "Configuring Oracle Software Owner Environment" on page 5-13 for more information about setting environment variables.
>
> - In addition, it is advisable to set the `PATH` environment variable to include the `$ORACLE_HOME/bin` directory.

Use the following statement to start SQL*Plus and log in as the `SYS` user, connecting as `SYSDBA`:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

For example, to log on as `SYSTEM` enter:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYSTEM
Enter password: password
```

If you are logging on as `SYS`, you must connect as `SYSDBA`:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about accessing Oracle Database using SQL*Plus
>
> - *SQL*Plus User's Guide and Reference* for more information on connecting to the default database
>
> - *SQL*Plus User's Guide and Reference* for more information on the `CONNECT` command syntax

Accessing Oracle Database with SQL Developer

To run the SQL and PL/SQL statements to access Oracle Database, you can use SQL Developer. All SQL and PL/SQL statements are supported because they are passed directly from the SQL Worksheet to the Oracle Database.

Set Up the JDK Path for SQL Developer

Set the following environmental variables to ensure that the correct JDK is picked up:

- `$ORACLE_HOME`

- `$JAVA_HOME=$ORACLE_HOME/jdk`

- `$PATH=$JAVA_HOME/bin/:$PATH`

To start SQL Developer on which the Java SDK release JDK 1.6.0_11 is installed, use the following commands:

- Change to `$ORACLE_HOME/sqldeveloper`.

- Run `$ ./sqldeveloper.sh`.

- Right-click **Connections**. In the dialog box, enter a connection name, username, password, and for the host string, the name of the database to which you want to connect, and click **Connect**.

After you are connected, you can view, create, modify, and delete the database objects using the Connection Navigator or issue any SQL or PL/SQL statement using a SQL Worksheet. From the **Tools** menu, select **SQL Worksheet.**

SQL*Plus statements have to be interpreted by the SQL Worksheet before being passed to the database. The SQL Worksheet currently supports many SQL*Plus statements. SQL*Plus statements which are not supported by the SQL Worksheet are ignored and are not sent to Oracle Database.

> **See Also:** "SQL*Plus Statements Supported and Not Supported in SQL Worksheet" in *Oracle SQL Developer User's Guide*

Reviewing Accounts and Passwords

All databases created by the Database Configuration Assistant (DBCA) include the SYS, SYSTEM, and DBSNMP database accounts. In addition, Oracle Database provides several other administrative accounts. Before using these accounts, you must unlock them and reset their passwords. Table 9–1 describes these accounts and lists their user names.

> **See Also:**

> **Note:** Use Oracle Enterprise Manager Database Express 12*c* to view the complete list of database accounts.

Table 9–1   Database Accounts

User Name	Description	See Also
ANONYMOUS	Enables HTTP access to Oracle XML DB.	*Oracle XML DB Developer's Guide*
APEX_040200	The account that owns the Oracle Application Express schema and metadata.	*Oracle Application Express Application Builder User's Guide*
APEX_PUBLIC_USER	The minimally privileged account used for Oracle Application Express configuration with Oracle Application Express Listener or Oracle HTTP Server and `mod_plsql`.	*Oracle Application Express Application Builder User's Guide*
APPQOSSYS	Used for storing and managing all data and metadata required by Oracle Quality of Service Management.	None

Table 9–1 (Cont.) Database Accounts

User Name	Description	See Also
AUDSYS	The account where the unified audit data trail resides.	*Oracle Database Security Guide*
BI	The account that owns the Business Intelligence schema included in the Oracle Sample Schemas. It is available only if you loaded the sample schemas.	*Oracle Database Sample Schemas*
CTXSYS	The Oracle Text account.	*Oracle Text Reference*
DBSNMP	The account used by the Management Agent component of Oracle Enterprise Manager to monitor and manage the database.	*Oracle Enterprise Manager Cloud Control Administrator's Guide*
DIP	The account used by the Directory Integration Platform (DIP) to synchronize the changes in Oracle Internet Directory with the applications in the database.	None
DVSYS	There are two roles associated with this account. The Database Vault owner role manages the Database Vault roles and configurations. The Database Vault Account Manager is used to manage database user accounts. **Note**: Part of Oracle Database Vault user interface text is stored in database tables in the DVSYS schema. By default, only the English language is loaded into these tables. You can use the `DVSYS.DBMS_MACADM.ADD_NLS_DATA` procedure to add other languages to Oracle Database Vault. See the "Adding Languages to Oracle Database Vault" section in the *Oracle Database Vault Administrator's Guide*	*Oracle Database Vault Administrator's Guide*
DVF	The account owned by Database Vault that contains public functions to retrieve the Database Vault Factor values.	*Oracle Database Vault Administrator's Guide*
EXFSYS	The account owns the Expression Filter schema.	None
FLOWS_FILES	The account owns the Oracle Application Express uploaded files.	*Oracle Application Express Application Builder User's Guide*
GSMADMIN_INTERNAL	The internal account that owns the Global Data Services schema. It should not be unlocked or used for a database login.	*Oracle Database Global Data Services Concepts and Administration Guide*
GSMCATUSER	The account used by Global Service Manager to connect to the Global Data Services catalog.	*Oracle Database Global Data Services Concepts and Administration Guide*
GSMUSER	The account used by Global Service Manager to connect to the database.	*Oracle Database Global Data Services Concepts and Administration Guide*
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.	*Oracle Database Sample Schemas*

Table 9–1   (Cont.) Database Accounts

User Name	Description	See Also
IX	The account that owns the Information Transport schema included in the Oracle Sample Schemas. It is available only if you loaded the sample schemas.	*Oracle Database Sample Schemas*
LBACSYS	The Oracle Label Security administrator account.	*Oracle Label Security Administrator's Guide*
MDDATA	The schema used by Oracle Spatial and Graph for storing geocoder and router data.	*Oracle Spatial and Graph Developer's Guide*
MDSYS	The Oracle Spatial and Graph, and Oracle Multimedia Locator administrator account.	*Oracle Spatial and Graph Developer's Guide*
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.	*Oracle Database Sample Schemas*
ORDPLUGINS	The Oracle Multimedia user. Plug-ins supplied by Oracle and third-party plug-ins are installed in this schema.	*Oracle Multimedia Reference*
ORDSYS	The Oracle Multimedia administrator account.	*Oracle Multimedia Reference*
ORDDATA	This account contains the Oracle Multimedia DICOM data model.	*Oracle Multimedia DICOM Developer's Guide*
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.	*Oracle Database Concepts*
ORACLE_OCM	This account contains the instrumentation for configuration collection used by the Oracle Configuration Manager.	*Oracle Configuration Manager Installation and Administration Guide*
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.	*Oracle Database Sample Schemas*
SCOTT	An account used by Oracle sample programs and examples.	*Oracle Database Administrator's Guide*
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.	*Oracle Database Administrator's Guide*
SI_INFORMTN_SCHEMA	The account that stores the information views for the SQL/MM Still Image Standard.	*Oracle Multimedia Reference*
SPATIAL_CSW_ADMIN_USR	The Catalog Services for the Web (CSW) account. It is used by the Oracle Spatial and Graph CSW cache manager to load all record type metadata, and record instances from the database into the main memory for the record types that are cached.	*Oracle Spatial and Graph Developer's Guide*

Table 9–1   (Cont.) Database Accounts

User Name	Description	See Also
SPATIAL_WFS_ADMIN_USR	The Web Feature Service (WFS) account. It is used by the Oracle Spatial and Graph WFS cache manager to load all feature-type metadata, and feature instances from the database into main memory for the feature types that are cached.	*Oracle Spatial and Graph Developer's Guide*
SYS	The account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
SYSTEM	Another account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
SYSBACKUP	The account used to perform backup and recovery tasks.	*Oracle Database Installation Guide* (this guide)
SYSKM	The account used to perform encryption key management.	*Oracle Database Installation Guide* (this guide)
SYSDG	The account used to administer and monitor Oracle Data Guard.	*Oracle Database Installation Guide* (this guide)
WMSYS	The account used to store the metadata information for Oracle Workspace Manager.	*Oracle Database Workspace Manager Developer's Guide*
XDB	The account used for storing Oracle XML DB data and metadata.	*Oracle XML DB Developer's Guide*

Unlocking and Resetting User Passwords

Passwords for all Oracle system administration accounts except SYS, SYSTEM, 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, you must unlock it, using one of the following methods:

- Using Oracle Enterprise Manager Database Express to Unlock Accounts and Reset Passwords

- Using SQL*Plus to Unlock Accounts and Reset Passwords

Note: If you are creating a database using Database Configuration Assistant, you can unlock accounts after the database is created by clicking **Password Management** before you exit from Database Configuration Assistant.

See Also: *Oracle Database Security Guide* for more information about how to create a password that is secure

Using Oracle Enterprise Manager Database Express to Unlock Accounts and Reset Passwords

To unlock and reset user account passwords using Oracle Enterprise Manager Database Express 12*c* see *Oracle Database 2 Day DBA*.

> **See Also:** Click **Help** in the Oracle Enterprise Manager Database Express 12*c* window for more information

Using SQL*Plus to Unlock Accounts and Reset Passwords

To unlock and reset user account passwords using SQL*Plus:

1. Start SQL*Plus and log in as the `SYS` user, connecting as `SYSDBA`:

   ```
   $ $ORACLE_HOME/bin/sqlplus
   SQL> CONNECT SYS as SYSDBA
   Enter password: SYS_password
   ```

2. Enter a command similar to the following, where *account* is the user account to unlock and *password* is the new password:

   ```
   SQL> ALTER USER account IDENTIFIED BY password ACCOUNT UNLOCK;
   ```

   > **Note:** If you unlock an account but do not reset the password, then the password remains expired. The first time someone connects as that user, they must change the user's password.

   See Also:

   - *Oracle Database Security Guide* to learn how to add new users and change passwords

   - *Oracle Database SQL Language Reference* for the `ALTER USER` statement syntax used for unlocking user accounts

   - *Oracle Database Administrator's Guide* for information about the users `SYS` and `SYSTEM`

Identifying Databases

The Oracle Database 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 equals 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.

The database name input field is used to set the `DB_NAME`, `DB_UNIQUE_NAME`, and `DB_DOMAIN` Oracle initialization parameter values.

For example:

```
sales_world.example.com
```

In this example:

- `sales_world` is the name of the database. The database name (`DB_UNIQUE_NAME`) portion is a string of no more than 30 characters that can contain alphanumeric characters, underscore (_), dollar sign ($), and pound sign (#) but must begin with an alphabetic character. No other special characters are permitted in a database name.

- `sales_wo` is the `DB_NAME`. The `DB_NAME` initialization parameter specifies a database identifier of up to eight characters.

- `example.com` is the database domain in which the database is located. In this example, the database domain equals the network domain. Together, the database name and the database domain make the global database name unique. The domain portion is a string of no more than 128 characters that can contain alphanumeric characters, underscore (_), and pound sign (#). The `DB_DOMAIN` initialization parameter specifies the database domain name.

However, the `DB_NAME` parameter need not necessarily be the first eight characters of `DB_UNIQUE_NAME`.

The `DB_UNIQUE_NAME` parameter and the `DB_DOMAIN` name parameter combine to create the global database name value 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` equals the database name portion of the global database name.

> **See Also:** "`DB_UNIQUE_NAME`" and "`DB_NAME`" in *Oracle Database Reference*

Locating the Server Parameter File

By default, the preconfigured database uses a server parameter file named `spfile`*sid*`.ora`, which is stored in the `$ORACLE_HOME/dbs` directory. However, if you choose Oracle ASM for the database, Database Configuration Assistant typically uses the same storage mechanism for the server parameter file.

If the server parameter file is not located in the `$ORACLE_HOME/dbs` directory, the database uses the `SPFILE` parameter in an initialization parameter file to locate it. The default initialization parameter file is `$ORACLE_HOME/dbs/init`*sid*`.ora`.

To use Oracle Enterprise Manager Database Express 12*c* to to view the location of the server parameter file and list the initialization parameters, see the "Viewing and Modifying Initialization Parameters" section in *Oracle Database 2 Day DBA*.

> **See Also:** Click **Help** in the Oracle Enterprise Manager Database Express 12*c* window for more information

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:

- Identifying Tablespaces and Data Files

- Locating Redo Log Files

- 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 12*c* databases.

Table 9–2 describes the tablespaces provided by the default preconfigured database.

Table 9–2   Tablespaces and Data Files

Tablespace	Data File	Description
EXAMPLE	EXAMPLE01.DBF	Stores the sample schemas, if you included them.
SYSAUX	SYSAUX01.DBF	Acts as an auxiliary tablespace to the SYSTEM tablespace. Some products and options that previously used the SYSTEM tablespace now use the SYSAUX tablespace to reduce the load on the SYSTEM tablespace.
SYSTEM	SYSTEM01.DBF	Stores the data dictionary, including definitions of tables, views, and stored procedures needed by Oracle Database. Information in this area is maintained automatically.
TEMP	TEMP01.DBF	Stores temporary tables and indexes created during the processing of your SQL statement. If you run a SQL statement that involves a lot of sorting, such as the constructs GROUP BY, ORDER BY, or DISTINCT, then you may have to expand this tablespace.
UNDOTBS	UNDOTBS01.DBF	Stores undo information. The **undo tablespace** contains one or more undo segments that maintain transaction history that is used to roll back, or undo, changes to the database. All starter databases are configured to run in automatic undo management mode.
USERS	USERS01.DBF	Stores database objects created by database users.

> **See Also:** *Oracle Database Concepts* and the *Oracle Database Administrator's Guide* for more information

To use Oracle Enterprise Manager Database Express 12*c* to view the list of data files used by the database and their associated tablespaces see the "Viewing Tablespace and Data File Information" section in *Oracle Database 2 Day DBA*.

Locating Redo Log Files

The preconfigured database uses three redo log files. Redo log files record all 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 User's Guide* for more information about redo log files

To view or modify the redo log files for the preconfigured database using Oracle Enterprise Manager Database Express 12*c* see the "Viewing Online Redo Log File Information" and "Viewing Archived Redo Log File Information" sections in *Oracle Database 2 Day DBA*.

Locating Control Files

The preconfigured database uses two control files. Oracle recommends that you keep at least two control files for each database and set the CONTROL_FILES initialization parameter to specify the location of each file.

A control file is an administrative file. Oracle Database 12*c* requires a control file to start and run the database. The control file defines the physical structure of the database. For example, it defines the database name and the names and locations of the database data files and redo log files.

To use Oracle Enterprise Manager Database Express 12*c* to view information about the control files for the preconfigured database see the "Viewing Control File Information" section in *Oracle Database 2 Day DBA*.

> **See Also:** *Oracle Database Administrator's Guide* for more information about setting the CONTROL_FILES initialization parameter value

For more information about using Oracle Enterprise Manager Database Express 12*c* to perform various tasks related to tablespaces and data files, redo log files, and control files, click **Help** in the Oracle Enterprise Manager Database Express window.

10

Removing Oracle Database Software

This chapter describes how to completely remove Oracle software and configuration files related to the specified Oracle home. It includes information about removing Oracle software using the deinstallation tool.

The deinstallation tool removes standalone Oracle Database installations, Oracle Clusterware and Oracle Automatic Storage Management (Oracle ASM) from your server, and also Oracle Real Application Clusters (Oracle RAC) and Oracle Database client installations.

Oracle recommends that you use the deinstallation tool to remove the entire Oracle home associated with the Oracle Database, Oracle Clusterware, Oracle ASM, Oracle RAC, or Oracle Database client installation. Oracle does not support the removal of individual products or components.

The following sections describe the deinstallation tools, and provide information about additional options to use the tool:

- About the Deinstallation Tool
- Example of Running the Deinstall Command
- Deinstallation Response File Example for Oracle Database
- Deinstallation Response File Example for Oracle Grid Infrastructure

> **Caution:** If you have a standalone database on a node in a cluster and you have multiple databases with the same global database name (GDN), then you cannot use the deinstall tool to remove one database only.

> **See Also:**
>
> - *Oracle Grid Infrastructure Installation Guide* and *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for information about removing an Oracle RAC installation
> - "Dropping Disk Groups" in *Oracle Automatic Storage Management Administrator's Guide* for information about removing an Oracle ASM disk group

About the Deinstallation Tool

Starting with Oracle Database 12*c*, the deinstallation tool is integrated with the database installation media. You can run the deinstallation tool using the `runInstaller` command with the `-deinstall` and `-home` options from the base

directory of the Oracle Database, Oracle Database Client or Oracle Grid Infrastructure installation media.

The deinstallation tool is also available as a separate command (`deinstall`) in Oracle home directories after installation. It is located in the `$ORACLE_HOME/deinstall` directory.

The deinstallation tool uses the information you provide, plus information gathered from the software home to create a response file. You can alternatively supply a response file generated previously by the `deinstall` command using the `-checkonly` option, or by editing the response file template.

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system for a specific Oracle home. If you run the deinstallation tool to remove an Oracle Grid Infrastructure installation, then the deinstaller prompts you to run the `roothas.sh` script, as the `root` user, to deconfigure Oracle Restart.

> **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), the `roothas.sh` script replaces the `roothas.pl` script in the Oracle Grid Infrastructure home.

If the software in the Oracle home is not running (for example, after an unsuccessful installation), then the deinstallation tool cannot determine the configuration, and you must provide all the configuration details either interactively or in a response file.

> **Caution:** When you run the deinstallation tool, if the central inventory (oraInventory) contains no other registered homes besides the home that you are deconfiguring and removing, then the deinstall command removes the following files and directory contents in the Oracle base directory of the Oracle Database installation owner:
>
> - `admin`
>
> - `cfgtoollogs`
>
> - `checkpoints`
>
> - `diag`
>
> - `oradata`
>
> - `fast_recovery_area`
>
> Oracle strongly recommends that you configure your installations using an Optimal Flexible Architecture (OFA) configuration, and that you reserve Oracle base and Oracle home paths for exclusive use of Oracle software. If you have any user data in these locations in the Oracle base that is owned by the user account that owns the Oracle software, then the deinstallation tool deletes this data.
>
> In addition, for Oracle Grid Infrastructure installations:
>
> - Oracle Automatic Storage Management Cluster File System (Oracle ACFS) must be dismounted and Oracle Automatic Storage Management Dynamic Volume Manager (Oracle ADVM) must be disabled.
>
> - If Grid Naming Service (GNS) is in use, then the entry for the subdomain must be deleted from DNS by your DNS administrator.
>
> The deinstallation tool deletes Oracle Database configuration files, user data, and fast recovery area (FRA) files even if they are located outside of the Oracle base directory path.

Oracle recommends that you run the deinstallation tool as the Oracle software installation owner. The default method for running the deinstallation tool is from the deinstall directory in the Oracle home as the installation owner:

```
$ $ORACLE_HOME/deinstall/deinstall
```

The deinstall command uses the following syntax, where variable content is indicated in italics:

```
./deinstall [-silent] [-checkonly] [-paramfile complete path of input response
file] [-params name1=value
name2=value . . .] [-o complete path of directory for saving files]
[-tmpdir complete path of temporary directory to use]
[-logdir complete path of log directory to use] [-help]
```

To run the deinstallation tool from the database installation media, use the runInstaller command with the -deinstall option, followed by the -home option to specify the path of the Oracle home you want to remove using the following syntax, where variable content is indicated in italics:

```
./runInstaller -deinstall -home complete path of Oracle home [-silent]
```

```
[-checkonly] [-paramfile complete path of input response file] [-params
name1=value
name2=value . . .] [-o complete path of directory for saving files]
[-tmpdir complete path of temporary directory to use]
[-logdir complete path of log directory to use] [-help]
```

Provide information about your servers as prompted or accept the defaults.

The deinstallation tool stops Oracle software, and removes Oracle software and configuration files on the operating system.

In addition, you can run the deinstallation tool with a response file, or select the following options to run the tool:

- -home

  Use this flag to indicate the home path of the Oracle home to check or deinstall.

  To deinstall Oracle software using the deinstall command, located in the Oracle home you plan to deinstall, provide a response file located outside the Oracle home, and do not use the -home flag.

  If you run the deinstallation tool from the $ORACLE_HOME/deinstall path, then the -home flag is not required because the tool identifies the location of the home where it is run. If you use runInstaller -deinstall from the installation media, then -home is mandatory.

- -silent

  Use this flag to run the deinstallation tool in noninteractive mode. This option requires one of the following:

  - A working system that it can access to determine the installation and configuration information. The -silent flag does not work with failed installations.

  - A response file that contains the configuration values for the Oracle home that is being deinstalled or deconfigured.

  You can generate a response file to use or modify by running the tool with the -checkonly flag. The tool then discovers information from the Oracle home to deinstall and deconfigure. It generates the response file that you can then use with the -silent option.

  You can also modify the template file deinstall.rsp.tmpl, located in the $ORACLE_HOME/deinstall/response directory.

- -checkonly

  Use this flag to check the status of the Oracle software home configuration. Running the deinstallation tool with the -checkonly flag does not remove the Oracle configuration. The -checkonly flag generates a response file that you can use with the deinstallation tool and -silent option.

- -paramfile complete path of input response file

  Use this flag to run the deinstallation tool with a response file in a location other than the default. When you use this flag, provide the complete path where the response file is located.

  The default location of the response file depends on the location of the deinstallation tool:

  - From the installation media or stage location: /response

      – After installation from the installed Oracle home: `$ORACLE_HOME/deinstall/response`

- `-params [name1=value name2=value name3=value ...]`

  Use this flag with a response file to override one or more values to change in a response file you have created.

- `-o complete path of directory for saving response files`

  Use this flag to provide a path other than the default location where the response file (`deinstall.rsp.tmpl`) is saved.

  The default location of the response file depends on the location of the deinstallation tool:

  – From the installation media or stage location: `/response`

  – After installation from the installed Oracle home: `$ORACLE_HOME/deinstall/response`

- `-tmpdir complete path of temporary directory`

  Use this flag to specify a non-default location where Oracle Deinstallation Tool writes the temporary files for the deinstallation.

- `-logdir complete path of log directory`

  Use this flag to specify a non-default location where Oracle Deinstallation Tool writes the log files for the deinstallation.

- `-help`

  Use the help option (`-help`) to get additional information about the command option flags.

> **See Also:** *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for information about the `-local` option

Deinstalling Previous Release Grid Home

For upgrades from previous releases, if you want to deinstall the previous release Grid home, then as the `root` user, you must manually change the permissions of the previous release Grid home, and then run the deinstall command.

For example:

```
# chown -R grid:oinstall /u01/app/grid/11.2.0
# chmod -R 775 /u01/app/grid/11.2.0
```

In this example, `/u01/app/grid/11.2.0` is the previous release Grid home.

Example of Running the Deinstallation Tool

If you run the deinstallation tool using the `runInstaller` command with the `-deinstall` option from the installation media, then help is displayed unless you enter a `-home` flag and provide a path to the home directory of the Oracle software to remove from your system.

Use the optional flag `-paramfile` to provide a path to a response file.

In the following example, the `runInstaller` command is in the path */directory_path*, where *directory_path* is the path to the `database` directory on the installation media,

and `/u01/app/oracle/product/12.1.0/dbhome_1/` is the path to the Oracle home remove:

```
$ cd /directory_path/
$ ./runInstaller -deinstall -home /u01/app/oracle/product/12.1.0/dbhome_1/
```

The following example uses a response file in the software owner location `/home/usr/oracle`:

```
$ cd /directory_path/
$ ./runInstaller -deinstall -paramfile /home/usr/oracle/my_db_paramfile.tmpl
```

Example of Running the Deinstall Command

If you run the deinstallation tool using the `deinstall` command from the `$ORACLE_HOME/deinstall` directory, then the deinstallation starts without prompting you for the Oracle home path.

Use the optional flag `-paramfile` to provide a path to a response file.

In the following example, the `deinstall` command is in the path `/u01/app/oracle/product/12.1.0/dbhome_1/deinstall`, and it uses a response file in the software owner location `/home/usr/oracle`:

```
$ cd /u01/app/oracle/product/12.1.0/dbhome_1/deinstall
$ ./deinstall -paramfile /home/usr/oracle/my_db_paramfile.tmpl
```

For the Oracle Grid Infrastructure home, use the deinstallation script in the Oracle Grid Infrastructure for a standalone server home, which in this example is `/u01/app/oracle/product/12.1.0/grid`:

```
$ cd /u01/app/oracle/product/12.1.0/grid/deinstall
$ ./deinstall -paramfile /home/usr/oracle/my_grid_paramfile.tmpl
```

Deinstallation Response File Example for Oracle Database

You can run the deinstallation tool with the `-paramfile` option to use the values you specify in the response file. The following is an example of a response file, in which the Oracle Database binary owner is `oracle`, the Oracle Database home (Oracle home) is in the path `/u01/app/oracle/product/12.1.0/dbhome_1/`, the Oracle base (where other Oracle software is installed) is `/u01/app/oracle/`, the central Oracle Inventory home (oraInventory) is `/u01/app/oraInventory`, the virtual IP address (VIP) is `192.0.2.1`, the local node (the node where you run the deinstallation session from) is `myserver`, and the OSDBA group is `dba`:

```
# Copyright (c) 2005, 2009 Oracle Corporation.  All rights reserved.
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
ORACLE_BASE.orcl=/u01/app/oracle
FAST_RECOVERY_LOC.orcl=/u01/app/oracle/fast_recovery_area/ORCL
STORAGE_TYPE.orcl=FS
DB_TYPE.orcl=SI_DB
NETCA_LOCAL_LISTENERS=LISTENER
LOGDIR=/u01/app/oraInventory/logs/
NODE_LIST.orcl=myserver
ObaseCleanupPtrLoc=/tmp/deinstall2012-06-12_09-14-11AM/orabase_cleanup.lst
ARCHIVE_LOG_DESTINATION_LOC.orcl=
ORACLE_BASE=/u01/app/oracle
DUMP_DESTINATION_LOC.orcl=/u01/app/oracle/admin/orcl
```

```
LOCAL_SID.orcl=orcl
INVENTORY_LOCATION=/u01/app/oraInventory
RAW_MAPPING_FILE.orcl=
SID_LIST.orcl=orcl
DB_UNIQUE_NAME_LIST=orcl
DATAFILE_LOC.orcl=/u01/app/oracle/oradata/orcl,/u01/app/oracle/fast_recovery_
area/orcl
HOME_TYPE=SIDB
CRS_HOME=false
CREATION_MODE.orcl=y
CONFIGFILE_LOC.orcl=
ORACLE_BINARY_OK=true
DIAG_DEST.orcl=/u01/app/oracle/diag/rdbms/orcl
LOCAL_NODE=myserver
local=false
SPFILE_LOC.orcl=/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfileorcl.ora
inst_group=dba
inventory_loc=/u01/app/oraInventory
MinimumSupportedVersion=11.2.0.1.0
silent=false
DBCA_LOG.orcl=/u01/app/oracle/cfgtoollogs/dbca/orcl
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
CCR_CONFIG_STATUS=CCR_DEL_HOME
EMCA_LOG.orcl=/u01/app/oracle/cfgtoollogs/emca/orcl
ORACLE_HOME_VERSION_VALID=true
```

Deinstallation Response File Example for Oracle Grid Infrastructure

You can run the deinstallation tool with the `-paramfile` option to use the values you specify in the response file.

The following is an example of a response file, in which the Oracle Grid Infrastructure binary owner is `oracle`, the Oracle Grid Infrastructure home is in the path `/u01/app/oracle/product/12.1.0/grid`, the Oracle base (where other Oracle software is installed) is `/u01/app/oracle/`, the central Oracle Inventory home (oraInventory) is `/u01/app/oraInventory`, the local node (the node where you run the deinstallation session from) is `myserver`, and the OSDBA group is `dba`:

```
# Copyright (c) 2005, 2009 Oracle Corporation.  All rights reserved.
ORACLE_HOME=/u01/app/oracle/product/12.1.0/grid
LOCAL_NODE=myserver
HOME_TYPE=SIHA
ASM_REDUNDANCY=EXTERNAL
ORACLE_BASE=/u01/app/oracle/
ObaseCleanupPtrLoc=/tmp/deinstall_bootstrap/orabase_cleanup.lst
SCAN_PORT=0
silent=false
ASM_UPGRADE=false
ORA_CRS_HOME=/u01/app/oracle/product/12.1.0/grid
MinimumSupportedVersion=11.2.0.1.0
GPNPCONFIGDIR=$ORACLE_HOME
LOGDIR=/u01/app/oraInventory/logs/
ORACLE_HOME_VERSION_VALID=true
ASM_DISCOVERY_STRING=/scratch/raw/raw*
GPNPGCONFIGDIR=$ORACLE_HOME
ORACLE_OWNER=oracle
ISROLLING=true
ASM_DISKSTRING=/scratch/raw/raw*
CRS_STORAGE_OPTION=0
ORACLE_BINARY_OK=true
```

```
MGMT_DB=false
NETCA_LISTENERS_REGISTERED_WITH_HAS=LISTENER
inst_group=svrtech
ASM_AU_SIZE=1
HUB_SIZE=0
ASM_ORACLE_BASE=/u01/app/oracle
ORA_DBA_GROUP=dba
JREDIR=/u01/app/oracle/product/12.1.0/grid/jdk/jre/
USER_IGNORED_PREREQ=false
inventory_loc=/u01/app/oraInventory
ASM_DISK_GROUPS="+DATA"
ORA_ASM_GROUP=dba
LANGUAGE_ID=AMERICAN_AMERICA.AL32UTF8
CSS_LEASEDURATION=400
ASM_HOME=/u01/app/oracle/product/12.1.0/grid
ASM_DIAGNOSTIC_DEST=/u01/app/oracle/product/OB
TZ=PST8PDT
REUSEDG=false
SILENT=false
local=false
INVENTORY_LOCATION=/u01/app/oraInventory
GNS_CONF=false
BIG_CLUSTER=false
ASM_
DISKS=/scratch/raw/raw1,/scratch/raw/raw2,/scratch/raw/raw3,/scratch/raw/raw4,/scr
atch/raw/raw5,/scratch/raw/raw6,/scratch/raw/raw7,/scratch/raw/raw8
ORACLE_HOME=/u01/app/oracle/product/12.1.0/grid
CRS_HOME=true
ASM_IN_HOME=true
CRFHOME="/u01/app/oracle/product/12.1.0/grid"
ASM_DROP_DISKGROUPS=true
ASM_LOCAL_SID=+ASM
JLIBDIR=/u01/app/oracle/product/12.1.0/grid/jlib
VNDR_CLUSTER=false
ASM_DISK_GROUP=DATA
```

A

Installing and Configuring Oracle Database Using Response Files

This appendix describes how to install and configure Oracle products using response files. It includes information about the following topics:

- How Response Files Work
- Preparing a Response File
- Running Oracle Universal Installer Using a Response File
- Running Net Configuration Assistant Using a Response File
- Running Database Configuration Assistant Using a Response File
- Postinstallation Configuration Using a Response File

How Response Files Work

You can automate the installation and configuration of Oracle software, either fully or partially, by specifying a response file when you start Oracle Universal Installer. Oracle Universal Installer uses the values contained in the response file to provide answers to some or all of Oracle Universal Installer prompts. It includes information about the following topics:

- Reasons for Using Silent Mode or Response File Mode
- Creating a Database Using Oracle Automatic Storage Management as the Storage Option for Database Files
- General Procedure for Using Response Files

Typically, Oracle Universal Installer runs in interactive mode, which means that it prompts you to provide information in graphical user interface (GUI) screens. When you use response files to provide this information, you run Oracle Universal Installer at a command prompt using either of the following modes:

- Silent mode

  If you include responses for all of the prompts in the response file and specify the `-silent` option when starting Oracle Universal Installer, then Oracle Universal Installer runs in silent mode. During a silent mode installation, Oracle Universal Installer does not display any screens. Instead, it displays progress information in the terminal that you used to start it.

- Response file mode

If you include responses for some or all of the prompts in the response file and omit the -silent option, then Oracle Universal Installer runs in response file mode. During a response file mode installation, Oracle Universal Installer displays all the screens, screens for which you specify information in the response file and also screens for which you did not specify the required information in the response file. The advantage is that you can validate the values in the screens for which you have provided the information in the response file and continue with the installation.

You define the settings for a silent or response file installation by entering values for the variables listed in the response file. For instance, to specify the Oracle home location, you would supply the appropriate value for the ORACLE_HOME variable, as follows:

```
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
```

Another way of specifying the variable settings of the response file is to pass them as command line arguments when you run Oracle Universal Installer. For example:

```
 -silent directory_path
```

Where, *directory_path* is the path to the database directory on the installation media or on the hard drive.

> **See Also:**
>
> - *Oracle Universal Installer User's Guide* for more information about response file formats
>
> - My Oracle Support website for more information about response files:
>
>   https://support.oracle.com/

Reasons for Using Silent Mode or Response File Mode

The following table describes several reasons why you might want to run Oracle Universal Installer in silent mode or response file mode.

Mode	Uses
Silent	Use silent mode to:
	- Complete an unattended installation, which you might schedule using operating system utilities such as cron
	- Complete several similar installations on multiple systems without user interaction
	- Install the software on a system that does not have X Window System software installed on it
	Oracle Universal Installer displays progress information in the terminal that you used to start it, but it does not display any of Oracle Universal Installer screens.
Response File	Use response file mode to complete similar Oracle software installations on multiple systems, providing default answers to some, but not all of Oracle Universal Installer prompts.
	In response file mode, all the installer screens are displayed, but defaults for the fields in these screens are provided by the response file. You must provide information for the fields in screens where you have not provided values in the response file.

Creating a Database Using Oracle Automatic Storage Management as the Storage Option for Database Files

Before you create a database that uses Oracle Automatic Storage Management (Oracle ASM), you must run the root.sh script. For this reason, you cannot create a database using Oracle ASM as the storage option for database files during a silent-mode installation. Instead, you can complete a software-only installation using silent mode, and then run Oracle Net Configuration Assistant and Database Configuration Assistant in silent mode after you complete the software-only installation and run the root.sh script.

> **Note:** This limitation applies only to databases that use Oracle ASM as the storage option for database files. You can create a database that uses the file system option during a silent mode installation.

General Procedure for Using Response Files

The following are the general steps to install and configure Oracle products using Oracle Universal Installer in silent or response file mode:

> **Note:** You must complete all required preinstallation tasks on a system before running Oracle Universal Installer in silent or response file mode.

1. Prepare a response file.

2. Run Oracle Universal Installer in silent or response file mode.

3. Run the root scripts as prompted by Oracle Universal Installer.

4. If you completed a software-only installation, then run Net Configuration Assistant and Database Configuration Assistant in silent or response file mode, if required.

These steps are described in the following sections.

Preparing a Response File

This section describes the following methods to prepare a response file for use during silent mode or response file mode installations:

> **Note:** Ensure that your response file name has the .rsp suffix.

- Editing a Response File Template
- Saving a Response File

Editing a Response File Template

Oracle provides response file templates for each product and installation type, and for each configuration tool. These files are located in the ORACLE_HOME/assistants/ directory, and the database/response directory on the installation media.

> **Note:** If you copied the software to a hard disk, the response files
> are located in the *stage_area*/database/response directory.

Table A–1 lists the response files provided with Oracle Database.

Table A–1   Response Files

Response File	Description
db_install.rsp	Silent installation of Oracle Database 12*c*
grid_install.rsp	Silent installation of Oracle Grid Infrastructure
dbca.rsp	Silent installation of Database Configuration Assistant
netca.rsp	Silent installation of Oracle Net Configuration Assistant

To copy and modify a response file:

1. Copy the response file from the response file directory to a directory on your system:

    ```
    $ cp /directory_path/response/response_file.rsp local_directory
    ```

    In this example, *directory_path* is the path to the database directory on the installation media. If you copy the software to a hard drive, then edit the file in the response directory.

2. Open the response file in a text editor:

    ```
    $ vi /local_dir/response_file.rsp
    ```

    > **See Also:** *Oracle Universal Installer User's Guide* for information about creating response files

3. Follow the instructions in the file to edit it.

    > **Note:** The installer or configuration assistants fail if you do not correctly configure the response file. See the "Silent-Mode Response File Error Handling" section on page I-7 for more information about troubleshooting a failed response file mode installation.

4. Change the permissions on the file to 700:

    ```
    $ chmod 700 /local_dir/response_file.rsp
    ```

    > **Note:** A fully specified response file for an Oracle Database installation contains the passwords for database administrative accounts and for a user who is a member of the OSDBA group (required for automated backups). Ensure that only the Oracle software owner user can view or modify response files or consider deleting them after the installation succeeds.

Saving a Response File

You can use Oracle Universal Installer in interactive mode to save a response file, which you can edit and then use to complete silent mode or response file mode installations.

Starting with Oracle Database 11*g* Release 2 (11.2), you can save all the installation steps into a response file during installation. You can click the **Save Response File** button on the Summary page to do this. Later, this file can be used for a silent installation.

When you save the response file, you can either complete the installation, or you can exit from Oracle Universal Installer on the Summary page, before it starts to copy the software to the system.

If you save a response file during a silent installation, then Oracle Universal Installer saves the variable values that were specified in the original source response file into the new response file.

> **Note:** Oracle Universal Installer does not save passwords in the response file.

To save a response file:

1. Complete the preinstallation tasks listed in Chapter 4.

   When you run Oracle Universal Installer to save a response file, it checks the system to verify that it meets the requirements to install the software. For this reason, Oracle recommends that you complete all of the required preinstallation tasks and save the response file while completing an installation.

2. Ensure that the Oracle software owner user has permissions to create or write to the Oracle home path that you specify when you run Oracle Universal Installer.

3. On each Oracle Universal Installer screen, provide the required information.

   > **See Also:** "Running Oracle Universal Installer" on page 7-9 for information about the installation process

4. When Oracle Universal Installer displays the Summary screen, perform the following:

   a. Click **Save Response File** and provide a file name and location for the response file. Then, click **Save** to save the values to the file.

   b. Click **Finish** to continue with the installation.

      Click **Cancel** if you do not want to continue with the installation. The installation stops, but the saved response file is retained.

5. Before you use the saved response file on another system, edit the file and make any required changes.

   Use the instructions in the file as a guide when editing it.

Running Oracle Universal Installer Using a Response File

Now, you are ready to run Oracle Universal Installer at the command line, specifying the response file you created, to perform the installation. The Oracle Universal Installer executable, `runInstaller`, provides several options. For help information

about the full set of these options, run the runInstaller command with the -help
option, for example:

```
$ directory_path/runInstaller -help
```

The help information appears in a window after some time.

To run Oracle Universal Installer using a response file:

1. Complete the preinstallation tasks listed in Chapter 4.

2. Log in as the Oracle software owner user (typically, oracle).

3. If you are completing a response file mode installation, set the DISPLAY
   environment variable.

> **Note:** You do not have to set the DISPLAY environment variable if
> you are completing a silent mode installation.

4. To start Oracle Universal Installer in silent or response file mode, enter a command
   similar to the following:

```
$ /directory_path/runInstaller [-silent] [-noconfig] \
 -responseFile responsefilename
```

> **Note:** Do not specify a relative path to the response file. If you
> specify a relative path, then Oracle Universal Installer fails.

In this example:

- *directory_path* is the path of the database directory on the DVD or on the
  hard drive.

- -silent runs Oracle Universal Installer in silent mode.

  "Silent-Mode Response File Error Handling" on page I-7

- -noconfig suppresses running the configuration assistants during installation,
  and a software-only installation is performed instead.

- *responsefilename* is the full path and file name of the installation response
  file that you configured.

5. Oracle Universal Installer prompts you to run the root.sh script. Log in as the
   root user and run the root.sh script:

```
$ sudo sh
password:
# /oracle_home_path/root.sh
```

6. If this is the first time you are installing Oracle software on your system, then
   Oracle Universal Installer prompts you to run the orainstRoot.sh script.

   Log in as the root user and run the orainstRoot.sh script:

```
$ sudo sh
password:
# /u01/app/oraInventory/orainstRoot.sh
```

> **Note:**
>
> ■ You do not have to manually create the `oraInst.loc` file. Running the `orainstRoot.sh` script is sufficient as it specifies the location of the Oracle Inventory directory.
>
> ■ During an Oracle Grid Infrastructure installation, Oracle Universal Installer provides options to automatically run the root scripts. You also have the option to manually run the root scripts. See "Determining Root Script Execution Plan" on page 5-16 for more information.

Running Net Configuration Assistant Using a Response File

You can run Net Configuration Assistant in silent mode to configure and start an Oracle Net listener on the system, configure naming methods, and configure Oracle Net service names. To run Net 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 in the `ORACLE_HOME/assistants/netca` directory, and the `response` directory in the `database/response` directory on the DVD.

> **Note:** If you copied the software to a hard disk, then the response file template is located in the *stage_area*/`database/response` directory.

To run Net Configuration Assistant using a response file:

1. Copy the `netca.rsp` response file template from the response file directory to a directory on your system:

   ```
   $ cp /directory_path/response/netca.rsp local_directory
   ```

   In this example, *directory_path* is the path of the `database` directory on the DVD. If you copy the software to a hard drive, you can edit the file in the `response` directory if you prefer.

2. Open the response file in a text editor:

   ```
   $ vi /local_dir/netca.rsp
   ```

3. Follow the instructions in the file to edit it.

   > **Note:** Net Configuration Assistant fails if you do not correctly configure the response file.

4. Log in as the Oracle software owner user, and set the `ORACLE_HOME` environment variable to specify the correct Oracle home directory.

5. Enter a command similar to the following to run Net Configuration Assistant in silent mode:

   ```
   $ $ORACLE_HOME/bin/netca -silent -responsefile /local_dir/netca.rsp
   ```

   In this command:

- The `-silent` option runs Net 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 Using a Response File

You can run Database Configuration Assistant in response file mode to configure and start an Oracle database on the system. To run Database Configuration Assistant in response file mode, you must copy and edit a response file template. Oracle provides a response file template named `dbca.rsp` in the `ORACLE_HOME/assistants/dbca` directory, and also in the `database/response` directory on the DVD.

> **Note:** If you copied the software to a hard disk, then the response file template is located in the *stage_area*`/database/response` directory.

To run Database Configuration Assistant in response file mode, you must use the `-responseFile` flag in combination with either the `-silent` or `-progressOnly` flag. You must also use a graphical display and set the `DISPLAY` environment variable.

> **See Also:**
> - "Creating a Database with Noninteractive/Silent DBCA" in *Oracle Database Administrator's Guide*
> - "Oracle ASM Configuration Assistant Command-Line Interface" section in *Oracle Automatic Storage Management Administrator's Guide* for information about running Oracle ASMCA in noninteractive mode

This section contains the following topics:

- Silent Mode of Database Configuration Assistant
- Progress Only Mode of Database Configuration Assistant
- Running Database Configuration Assistant in Response File Mode

Silent Mode of Database Configuration Assistant

Use the `-silent` flag in combination with the `-responseFile` flag to set the mode to silent. In the silent mode, Database Configuration Assistant uses values that you specify, in the response file or as command-line options, to create a database. No window or user interface is displayed in the silent mode.

Progress Only Mode of Database Configuration Assistant

Use the `-progressOnly` flag in combination with the `-responseFile` flag, to set the mode to progress only. As it configures and starts the database, Database Configuration Assistant displays a window that contains status messages and a progress bar.

In this mode, Database Configuration Assistant uses values that you specify, in the response file or as command line options, to create a database.

Running Database Configuration Assistant in Response File Mode

To run Database Configuration Assistant in response file mode, that is, silent mode or progress only mode:

> **Note:** Instead of editing the response file template, you can create a database by specifying all required information as command-line options when you run Database Configuration Assistant. For information about the list of options supported, enter the following command:
>
> `$ $ORACLE_HOME/bin/dbca -help`

1. Copy the `dbca.rsp` response file template from the response file directory to a directory on your system:

   `$ cp /directory_path/response/dbca.rsp local_directory`

   In this example, `directory_path` is the path of the `database` directory on the DVD. If you copy the software to a hard drive, you can edit the file in the `response` directory if you prefer.

2. Open the response file in a text editor:

   `$ vi /local_dir/dbca.rsp`

3. Edit the file, following the instructions in the file.

   > **Note:** Database Configuration Assistant fails if you do not correctly configure the response file.

4. Log in as the Oracle software owner user, and set the `ORACLE_HOME` environment variable to specify the correct Oracle home directory.

5. If you intend running Database Configuration Assistant in response file mode, set the `DISPLAY` environment variable.

6. Enter a command similar to the following to run Database Configuration Assistant in response file or silent mode with a response file:

   ```
   $ORACLE_HOME/bin/dbca {-progressOnly | -silent} -responseFile \
   /local_dir/dbca.rsp
   ```

   In this example:

   - The `-silent` option runs Database Configuration Assistant in silent mode.

   - The `-progressOnly` option runs Database Configuration Assistant in response file mode.

   - `local_dir` is the full path of the directory where you copied the `dbca.rsp` response file template.

Postinstallation Configuration Using a Response File

Use the following sections to create and run a response file configuration after installing the Oracle software.

About the Postinstallation Configuration File

When you run a silent or response file installation, you provide information about your servers in a response file that you would otherwise provide manually during a graphical user interface installation. However, the response file does not contain passwords for user accounts that configuration assistants require after the software installation is complete. The configuration assistants are started with a script called `configToolAllCommands`. You can run this script in response file mode by using a password response file. The script uses the passwords to run the configuration tools in succession to complete the configuration.

If you keep the password file to use for clone installations, then Oracle recommends that you store it in a secure location. In addition, if you must stop an installation to fix an error, you can run the configuration assistants using `configToolAllCommands` and a password response file.

The `configToolAllCommands` password response file consists of the following syntax options:

- *internal_component_name* is the name of the component that the configuration assistant configures

- *variable_name* is the name of the configuration file variable

- *value* is the desired value to use for the configuration

The command syntax is as follows:

internal_component_name|variable_name=value

For example:

```
oracle.assistants.asm|S_ASMPASSWORD=welcome
```

Oracle recommends that you maintain security with a password response file:

- Set the permissions on the response file to 600.

- The response file owner must be the installation owner user, with the group set to the central inventory (oraInventory) group.

Running Postinstallation Configuration Using a Response File

To run configuration assistants with the `configToolAllCommands` script:

1. Create a response file using the syntax *filename*.properties. For example:

   ```
   $ touch cfgrsp.properties
   ```

2. Open the file with a text editor, and cut and paste the password template, modifying it as needed.

Example A–1   Password response file for Oracle Grid Infrastructure for a standalone server

Oracle Grid Infrastructure requires passwords for Automatic Storage Management Configuration Assistant (Oracle ASMCA) and for Intelligent Platform Management Interface Configuration Assistant (IPMICA) if you have a BMC card and you want to enable this feature. Provide the following response file:

```
oracle.assistants.asm|S_ASMPASSWORD=password
oracle.assistants.asm|S_ASMMONITORPASSWORD=password
```

Example A–2   Password response file for Oracle Database

Oracle Database configuration requires the SYS, SYSTEM, and DBSNMP passwords for use with Database Configuration Assistant (DBCA). The S_ASMSNMPPASSWORD password is necessary only if the database is using Oracle ASM for storage. Similarly, the S_PDBADMINPASSWORD password is necessary only if you create a multitenant container database (CDB) with one or more pluggable databases (PDBs). Also, if you selected to configure Oracle Enterprise Manager, then you must provide the password for the Oracle software installation owner for the S_EMADMINPASSWORD password, similar to the following example:

```
oracle.assistants.server|S_SYSPASSWORD=password
oracle.assistants.server|S_SYSTEMPASSWORD=password
oracle.assistants.server|S_DBSNMPPASSWORD=password
oracle.assistants.server|S_PDBADMINPASSWORD=password
oracle.assistants.server|S_EMADMINPASSWORD=password
oracle.assistants.server|S_ASMSNMPPASSWORD=password
```

If you do not want to enable Oracle Enterprise Manager or Oracle ASM, then leave those password fields blank

3. Change permissions to secure the file. For example:

```
$ ls -al cfgrsp.properties
-rw------- 1 oracle oinstall 0 Apr 30 17:30 cfgrsp.properties
```

4. Change the directory to $ORACLE_HOME/cfgtoollogs

   Run the configuration script using the following syntax:

   ```
   configToolAllCommands RESPONSE_FILE=/path/name.properties
   ```

   for example:

   ```
   $ ./configToolAllCommands RESPONSE_FILE=/home/oracle/cfgrsp.properties
   ```

B

Cloning an Oracle Home

Cloning an Oracle home involves creating a copy of the Oracle home and then configuring it for a new environment. If you are performing multiple Oracle Database installations, then you may want to use this method to create each Oracle home, because copying files from an existing Oracle Database installation takes less time than creating a new version of them. This method is also useful if the Oracle home that you are cloning has had patches applied to it. When you clone the Oracle home, the new Oracle home has the patch updates.

This appendix includes information about the following topics:

- Cloning an Oracle Home

- Configuring Oracle Configuration Manager in a Cloned Oracle Home

Cloning an Oracle Home

Perform the following to clone an Oracle home:

1. Verify that the installation of Oracle Database to clone is successful.

   You can do this by reviewing the installActions*date_time*.log file for the installation session, which is typically located in the */orainventory_ location*/logs directory.

   If you install patches, then check their status using the following:

   ```
   $ cd $ORACLE_HOME/OPatch
   ```
   Include $ORACLE_HOME/OPatch in $PATH

   ```
   $ opatch lsinventory
   ```

2. Stop all processes related to the Oracle home. See "Stopping Existing Oracle Processes" on page 5-12 for more information about stopping the processes for an Oracle home.

3. Create a ZIP or TAR file with the Oracle home (but not the Oracle base) directory.

   For example, if the source Oracle installation is in the /u01/app/oracle/product/12.1.0/dbhome_1, then you zip the dbhome_1 directory by using the following command:

   ```
   # zip -r dbhome_1.zip /u01/app/oracle/product/12.1.0/dbhome_1
   ```

   You can also use the TAR command, for example:

   ```
   # tar -cvf dbhome_1.tar /u01/app/oracle/product/12.1.0/dbhome_1
   ```

Do not include the `admin`, `fast_recovery_area`, and `oradata` directories that are under the Oracle base directory. These directories are created in the target installation later, when you create a new database there.

4. Copy the ZIP or TAR file to the root directory of the target computer. If you use File Transfer Protocol (FTP), then transfer the ZIP or TAR file in binary mode only.

5. Extract the ZIP or TAR file content using the following command:

```
# unzip -d / dbhome_1.zip

# tar -xvf dbhome_1.tar
```

6. Repeat steps 4 and 5 for each computer where you want to clone the Oracle home, unless the Oracle home is on a shared storage device.

7. On the target computer, change the directory to the unzipped Oracle home directory, and remove all the .ora (*.ora) files present in the unzipped `$ORACLE_HOME/network/admin` directory.

8. From the `$ORACLE_HOME/clone/bin` directory, run the `clone.pl` file for the unzipped Oracle home. Use the following syntax:

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/clone/bin/clone.pl ORACLE_BASE="target_
oracle_base" ORACLE_HOME="target_oracle_home"
OSDBA_GROUP=OSDBA_privileged_group OSOPER_GROUP=OSOPER_privileged_group
-defaultHomeName
```

For example:

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/clone/bin/clone.pl ORACLE_
BASE="/u01/app/oracle/" ORACLE_HOME="/u01/app/oracle/product/12.1.0/dbhome_1"
OSDBA_GROUP=dba OSOPER_GROUP=oper -defaultHomeName
```

Oracle Universal Installer starts, and then records the cloning actions in the `cloneActionstimestamp.log` file. This log file is typically located in `/orainventory_location/logs` directory.

9. To configure the connection information for the new database, run Net Configuration Assistant:

```
$ cd $ORACLE_HOME/bin
$ ./netca
```

10. To create a new database for the newly cloned Oracle home, run Database Configuration Assistant:

```
$ cd $ORACLE_HOME/bin
$ ./dbca
```

> **See Also:**
>
> - *Oracle Universal Installer User's Guide* for detailed information about using Oracle Universal Installer to clone an Oracle Database home
>
> - *Oracle Database Administrator's Guide* for information about cloning Oracle databases and cloning an Oracle Database home

Configuring Oracle Configuration Manager in a Cloned Oracle Home

Configuring Oracle Configuration Manager for a cloned Oracle home depends on its configuration in the original Oracle home.

If you have already installed but not configured Oracle Configuration Manager in the original Oracle home, then run the following commands from the cloned Oracle home:

```
$ cd $ORACLE_HOME/ccr/bin
$ setupCCR
```

If you have already configured Oracle Configuration Manager in the original Oracle home, then run the following commands from the cloned Oracle home:

```
$ cd $ORACLE_HOME/ccr/bin
$ ./deriveCCR
```

If `deriveCCR` cannot find the original configuration, then the Oracle Configuration Manager command-line interface prompts for your My Oracle Support (MOS) credentials. Provide your MOS credentials to proceed.

C

Using NAS Devices

This appendix provides guidelines for using a network attached storage (NAS) device for Oracle software and database files. It includes information about the following:

- General Configuration Guidelines for NAS Devices
- Choosing Mount Points
- Creating Files on a NAS Device for Use with Oracle Automatic Storage Management

> **See Also:**
>
> - "Configuring Direct NFS Client" on page 8-10
> - *Oracle Grid Infrastructure Installation Guide* for information about using NAS devices on Oracle Real Application Clusters

General Configuration Guidelines for NAS Devices

See the documentation provided with the NAS device for specific information about how to configure it. In addition, use the following guidelines to ensure the performance of the Oracle software:

- The performance of Oracle software and databases stored on NAS devices depends on the performance of the network connection between the Oracle server and the NAS device.

  For this reason, Oracle recommends that you connect the server to the NAS device using a private dedicated network connection, which should be Gigabit Ethernet or better.

- For single-instance database installations, Oracle recommends that you create a separate Oracle home directory for each installation. Run the software in this Oracle home directory only from the system that you used to install it.

Choosing Mount Points

This section provides guidelines on how to choose the mount points for the file systems to use for the Oracle software and database files. The guidelines contained in the following sections follow the Optimal Flexible Architecture recommendations:

- Choosing Mount Points for Oracle Software Files
- Choosing Mount Points for Oracle Database and Recovery Files

Choosing Mount Points for Oracle Software Files

Oracle software files are stored in three different directories:

- Oracle base directory
- Oracle inventory directory
- Oracle home directory

The Oracle base directory is a top-level directory for Oracle software installations and is identified by the ORACLE_BASE environment variable. For example, for a first installation, the Oracle base, Oracle Inventory, and Oracle home directories might have paths similar to the following:

Directory	Path
Oracle base ($ORACLE_BASE)	/u01/app/oracle
Oracle Inventory	ORACLE_BASE/../oraInventory (or) /u01/app/oraInventory
Oracle home	$ORACLE_BASE/product/12.1.0/dbhome_1

For subsequent installations, you can use either the same Oracle base directory or a different one, but every subsequent installation uses the original Oracle Inventory directory. For example, if you use the /u02/app/oracle directory as the Oracle base directory for a new installation, then the Oracle Inventory directory continues to be /u01/app/oraInventory.

To enable you to effectively maintain the Oracle software on a particular system, Oracle recommends that you keep the Oracle Inventory directory only on a local file system, if possible. If you must place the Oracle Inventory directory on a NAS device, create a specific directory for each system, to prevent multiple systems from writing to the same inventory directory.

> **See Also:** "Identifying Required Software Directories" on page 4-34

Directory-Specific Guidelines

You can use any of the following directories as mount points for NFS used to store Oracle software:

> **Note:** In the following examples, the paths shown are the defaults if the ORACLE_BASE environment variable is set before you start Oracle Universal Installer.

- Oracle base directory or its parents (/u01/app/oracle for example)

  If you use the Oracle base directory or one of its parents as a mount point, then the default location for all Oracle software and database files is on that file system. During the installation, consider changing the default location of the following directories:

  - The Oracle Inventory directory (*oracle_base*/../oraInventory)

    Specify a local file system or a host-specific directory on the NFS, for example:

    u01/*app*/oraInventory

  - The Oracle database file directory (*oracle_base*/oradata)

You might want to use a different file system for database files, for example, to enable you to specify different mount options or to distribute I/O.

- The Oracle database recovery file directory (`oracle_base`/fast_recovery_area)

  Oracle recommends that you use different file systems for database and recovery files.

If you use this mount point, then all Oracle installations that use this Oracle base directory use the NFS.

- The product directory (`oracle_base`/product)

  By default, only software files are located on the NFS. You can also use this mount point to install software from different releases, for example:

  ```
  /u01/app/oracle/product/9.2.0
  /u01/app/oracle/product/10.2.0/dbhome_1
  /u01/app/oracle/product/12.1.0/dbhome_1
  ```

- The release directory (`oracle_base`/product/12.1.0)

  By default, only software files are located on the NFS. You can also use this mount point to install different products from the same release, for example:

  ```
  /u01/app/oracle/product/12.1.0/dbhome_1
  /u01/app/oracle/product/12.1.0/client_1
  ```

- The Oracle home directory (`oracle_base`/product/12.1.0/dbhome_1)

  By default, only software files are located on the NFS file system. This is the most restrictive mount point. You can use it only to install a single release of one product:

  ```
  /u01/app/oracle/product/12.1.0/dbhome_1
  ```

> **See Also:**

Choosing Mount Points for Oracle Database and Recovery Files

To store Oracle database or recovery files on a NAS device, you can use different paths depending on whether you want to store files from only one database or from multiple databases:

- Use the NFS for files from multiple databases

  To store the database files or recovery files from multiple databases on the same NFS, use paths or mount points similar to the following:

File Type	Path or Mount Point
Database files	`/u02/oradata`
Recovery files	`/u03/fast_recovery_area`

When Oracle Universal Installer prompts you for the data file and the recovery file directories, specify these paths. Database Configuration Assistant and Oracle Enterprise Manager create subdirectories in these directories using the value you specify for the database name (DB_NAME) as the directory name, for example:

```
/u02/oradata/db_name1
/u03/fast_recovery_area/db_name1
```

- Use the NFS for files from only one database

  To store the database files or recovery files for only one database in the NFS, you can create mount points similar to the following, where `orcl` is the name of the database:

  ```
  /u02/oradata/orcl
  /u03/fast_recovery_area/orcl
  ```

  Specify the directory `/u02/oradata` when Oracle Universal Installer prompts you for the data file directory and specify the directory `/u03/fast_recovery_area` when Oracle Universal Installer prompts you for the recovery file location. The `orcl` directory is used automatically either by Database Configuration Assistant or by Oracle Enterprise Manager.

Creating Files on a NAS Device for Use with Oracle Automatic Storage Management

If you have a certified NAS device, then you can create zero-padded files in an NFS mounted directory and use those files as disk devices in an Oracle Automatic Storage Management (Oracle ASM) disk group. To create these files, follow these steps:

> **Note:** To use files as disk devices in an Oracle ASM disk group, the files must be on an NFS mounted file system. You cannot use files on local file systems.

1. If necessary, create an exported directory for the disk group files on the NAS device.

   See the NAS device documentation for more information about completing this step.

2. Switch the user to `root`:

   ```
   $ sudo sh
   password:
   ```

3. Create a mount point directory on the local system:

   ```
   # mkdir -p /mnt/oracleasm
   ```

4. To ensure that NFS is mounted when the system restarts, add an entry for the file system in the `/etc/fstab` mount file.

   > **See Also:** My Oracle Support Note 359515.1 for updated NAS mount option information:
   >
   > https://support.oracle.com/CSP/main/article?cmd=show&type=NO
   > T&id=359515.1

5. Enter a command similar to the following to mount the NFS on the local system:

   ```
   # mount /mnt/oracleasm
   ```

6. Choose a name for the disk group to create, for example, `sales1`.

7. Create a directory for the files on the NFS file system, using the disk group name as the directory name:

```
# mkdir /mnt/oracleasm/sales1
```

8. Use commands similar to the following to create the required number of zero-padded files in this directory:

```
# dd if=/dev/zero of=/mnt/oracleasm/sales1/disk1 bs=1024k count=1000
oflag=direct
```

This example creates 1 GB files on the NFS. You must create one, two, or three files respectively to create an external, normal, or high redundancy disk group.

> **Note:** Creating multiple zero-padded files on the same NAS device does not guard against NAS failure. Instead, create one file for each NAS device and mirror them using the Oracle ASM technology.

9. Enter commands similar to the following to change the owner, group, and permissions on the directory and files that you created:

```
# chown -R grid:asmadmin /mnt/oracleasm
# chmod -R 660 /mnt/oracleasm
```

In this example, the installation owner is grid and the OSASM group is asmadmin.

10. During database installation and creation, edit the Oracle ASM disk discovery string to specify a regular expression that matches the file names you created. For example:

```
/mnt/oracleasm/sales1/*
```

> **Note:** During installation, disks labelled as ASMLIB disks or ASMFD disks are listed as candidate disks when using the default discovery string. However, if the disk has a header status of MEMBER, then it is not a candidate disk.

D

How to Complete Preinstallation Tasks Manually

This appendix provides instructions for how to complete configuration tasks manually that Cluster Verification Utility (CVU) and Oracle Universal Installer normally complete during installation. Use this appendix as a guide if you cannot use the fixup script.

This appendix contains the following information:

- Configuring Kernel Parameters for Linux
- Setting UDP and TCP Kernel Parameters Manually
- Configuring Storage Paths and Disk Devices

Configuring Kernel Parameters for Linux

This section contains the following topics:

- Minimum Parameter Settings for Installation
- Displaying and Changing Kernel Parameter Values
- Additional Parameter and Kernel Settings for SUSE Linux

> **Note:** Unless otherwise specified, the kernel parameter and shell limit values shown in the following table are minimum values only. For production database systems, Oracle recommends that you tune these values to optimize the performance of the system. See the operating system documentation for more information about tuning kernel parameters.

Minimum Parameter Settings for Installation

During the Oracle Database installation, you can generate and run the fixup script to check and set the kernel parameter values required for successful installation of the database. This script updates required kernel parameters, if necessary, to minimum values.

If you cannot use the fixup script, then review the following table to set the values manually:

Parameter	Value	File
`semmsl`	250	`/proc/sys/kernel/sem`
`semmns`	32000	
`semopm`	100	
`semmni`	128	
`shmall`	40 percent of the size of physical memory in pages **Note:** If the server supports multiple databases, or uses a large SGA, then set this parameter to a value that is equal to the total amount of shared memory, in 4K pages, that the system can use at one time.	`/proc/sys/kernel/shmall`
`shmmax`	Half the size of physical memory in bytes See My Oracle Support Note 567506.1 for additional information about configuring `shmmax`.	`/proc/sys/kernel/shmmax`
`shmmni`	4096	`/proc/sys/kernel/shmmni`
`panic_on_oops`	1	`/proc/sys/kernel/panic_on_oops`
`file-max`	6815744	`/proc/sys/fs/file-max`
`aio-max-nr`	1048576 **Note**: This value limits concurrent outstanding requests and should be set to avoid I/O subsystem failures.	`/proc/sys/fs/aio-max-nr`
`ip_local_port_range`	**Minimum**: 9000 **Maximum**: 65500 See Setting UDP and TCP Kernel Parameters Manually	`/proc/sys/net/ipv4/ip_local_port_range`
`rmem_default`	262144	`/proc/sys/net/core/rmem_default`
`rmem_max`	4194304	`/proc/sys/net/core/rmem_max`
`wmem_default`	262144	`/proc/sys/net/core/wmem_default`
`wmem_max`	1048576	`/proc/sys/net/core/wmem_max`

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

See Also:

- "Guidelines for Setting Semaphore Parameters" on page 8-3
- "Using Installation Fixup Scripts" on page 4-10

Displaying and Changing Kernel Parameter Values

Enter the commands shown in the following table to display the current values of the kernel parameters. Note these values and identify any values that you must change:

Parameter	Command
semmsl, semmns, semopm, and semmni	# /sbin/sysctl -a | grep sem This command displays the value of the semaphore parameters in the order listed.
shmall, shmmax, and shmmni	# /sbin/sysctl -a | grep shm This command displays the details of the shared memory segment sizes.
file-max	# /sbin/sysctl -a | grep file-max This command displays the maximum number of file handles.
ip_local_port_range	# /sbin/sysctl -a | grep ip_local_port_range This command displays a range of port numbers.
rmem_default	# /sbin/sysctl -a | grep rmem_default
rmem_max	# /sbin/sysctl -a | grep rmem_max
wmem_default	# /sbin/sysctl -a | grep wmem_default
wmem_max	# /sbin/sysctl -a | grep wmem_max
aio-max-nr	# /sbin/sysctl -a | grep aio-max-nr

If the value of any kernel parameter is different from the minimum value, then perform the following:

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

   Note: Include lines only for the kernel parameter values to change. For the semaphore parameters (kernel.sem), you must specify all four values. If any of the current values are larger than the minimum value, then specify the larger value.

   ```
   fs.aio-max-nr = 1048576
   fs.file-max = 6815744
   kernel.shmall = 2097152
   kernel.shmmax = 4294967295
   kernel.shmmni = 4096
   kernel.sem = 250 32000 100 128
   net.ipv4.ip_local_port_range = 9000 65500
   net.core.rmem_default = 262144
   net.core.rmem_max = 4194304
   net.core.wmem_default = 262144
   net.core.wmem_max = 1048576
   ```

By specifying the values in the `/etc/sysctl.conf` file, they persist when you restart the system. On SUSE Linux Enterprise Server systems, enter the following command to ensure that the system reads the `/etc/sysctl.conf` file when it restarts:

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

2. Enter the following command to change the current values of the kernel parameters:

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

   Review the output from this command to verify that the values are correct. If the values are incorrect, edit the `/etc/sysctl.conf` file, then enter this command again.

3. Enter the command `/sbin/sysctl -a` to confirm that the values are set correctly.

4. After updating the values of the kernel parameters in the `/etc/sysctl.conf` file, either restart the computer, or run the command `sysctl -p` to make the changes in the `/etc/sysctl.conf` file available in the active kernel memory.

Additional Parameter and Kernel Settings for SUSE Linux

On SUSE Linux Enterprise Server systems only, complete the following steps as needed:

1. Enter the following command to cause the system to read the `/etc/sysctl.conf` file when it restarts:

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

2. Enter the GID of the `oinstall` group as the value for the parameter `/proc/sys/vm/hugetlb_shm_group`. Doing this grants members of `oinstall` a group permission to create shared memory segments.

   For example, where the oinstall group GID is 501:

```
# echo 501 > /proc/sys/vm/hugetlb_shm_group
```

   After running this command, use `vi` to add the following text to `/etc/sysctl.conf`, and enable the `boot.sysctl` script to run on system restart:

```
vm.hugetlb_shm_group=501
```

Note: Only one group can be defined as the `vm.hugetlb_shm_group`.

Setting UDP and TCP Kernel Parameters Manually

If you do not use a Fixup script or CVU to set ephemeral ports, then set TCP/IP ephemeral port range parameters to provide enough ephemeral ports for the anticipated server workload. Ensure that the lower range is set to at least 9000 or higher, to avoid Well Known ports, and to avoid ports in the Registered Ports range commonly used by Oracle and other server ports. Set the port range high enough to avoid reserved ports for any applications you may intend to use. If the lower value of the range you have is greater than 9000, and the range is large enough for your anticipated workload, then you can ignore Oracle Universal Installer warnings regarding the ephemeral port range.

For example, with IPv4, use the following command to check your current range for ephemeral ports:

```
$ cat /proc/sys/net/ipv4/ip_local_port_range
32768 61000
```

In the preceding example, the lowest port (32768) and the highest port (61000) are set to the default range.

If necessary, update the UDP and TCP ephemeral port range to a range high enough for anticipated system workloads, and to ensure that the ephemeral port range starts at 9000 and above. For example:

```
# echo 9000 65500 > /proc/sys/net/ipv4/ip_local_port_range
```

Oracle recommends that you make these settings permanent. For example, as `root`, use a text editor to open `/etc/sysctl.conf`, and add or change to the following: `net.ipv4.ip_local_port_range = 9000 65500`, and then restart the network (`# /etc/rc.d/init.d/network restart`). Refer to your Linux distribution system administration documentation for detailed information about how to automate this ephemeral port range alteration on system restarts.

> **See Also:** "Setting TCP Network Protocol Buffer for Direct NFS Client" on page 8-12 if you use Direct NFS Client

Configuring Storage Paths and Disk Devices

Oracle recommends that you use Oracle ASM Filter Driver (Oracle ASMFD) to maintain device persistence. However, you can choose to use Oracle Automatic Storage Management library driver (Oracle ASMLIB) or set `udev` rules for device persistence.

This section contains the following topics:

- Configuring Storage Device Path Persistence Using Oracle ASMLIB
- Configuring Disk Devices Manually for Oracle Automatic Storage Management

Configuring Storage Device Path Persistence Using Oracle ASMLIB

Review the following section to configure Oracle ASMLIB:

- About Oracle ASM with Oracle ASMLIB
- Configuring Oracle ASMLIB to Maintain Block Devices
- Deinstalling Oracle ASMLIB

> **Note:** Oracle ASMLIB is not supported on IBM:Linux on System z.

About Oracle ASM with Oracle ASMLIB

The Oracle Automatic Storage Management library driver simplifies the configuration and management of block disk devices by eliminating the need to rebind block disk devices used with Oracle Automatic Storage Management (Oracle ASM) each time the system is restarted.

With Oracle ASMLIB, you define the range of disks you want to have made available as Oracle ASM disks. Oracle ASMLIB maintains permissions and disk labels that are

persistent on the storage device, so that the label is available even after an operating system upgrade.

> **Note:** If you configure disks using Oracle ASMLIB, then you must change the disk discovery string to `ORCL:*`. If the diskstring is set to `ORCL:*`, or is left empty (`""`), then the installer discovers these disks.

> **See Also:** "Using Oracle ASM Library Driver" in *Oracle Automatic Storage Management Administrator's Guide*

Configuring Oracle ASMLIB to Maintain Block Devices

To use Oracle ASMLIB to configure Oracle ASM devices, complete the following tasks:

- Installing and Configuring Oracle ASMLIB Software
- Configuring Disk Devices to Use Oracle ASMLIB
- Administering Oracle ASMLIB and Disks

> **Note:** To create a database during the installation using the Oracle ASM library driver, you must choose an installation method that runs ASMCA in interactive mode. You must also change the disk discovery string to `ORCL:*`.

> **See Also:** "Using an Existing Oracle Automatic Storage Management Disk Group" on page 7-3

Installing and Configuring Oracle ASMLIB Software Oracle ASMLIB is included with the Oracle Linux packages, and with SUSE Linux Enterprise Server 11. If you are a member of the Unbreakable Linux Network, then you can install the Oracle ASMLIB RPMs by subscribing to the Oracle Linux channel, and using `yum` to retrieve the most current package for your system and kernel. For additional information, see the following URL:

http://www.oracle.com/technetwork/topics/linux/asmlib/index-101839.html

To install and configure the Oracle Automatic Storage Management library driver software manually, perform the following steps:

1. Enter the following command to determine the kernel version and architecture of the system:

   ```
   # uname -rm
   ```

2. If necessary, download the required Oracle Automatic Storage Management library driver packages from the Oracle Technology Network website:

   http://www.oracle.com/technetwork/server-storage/linux/downloads/index-088143.html

> **Note:** You must install `oracleasm-support` package version 2.0.1 or later to use Oracle ASMLIB on Red Hat Enterprise Linux 5 Advanced Server. Oracle ASMLIB is already included with SUSE Linux Enterprise Server distributions.

> **See Also:** My Oracle Support note 1089399.1 for information about Oracle ASMLIB support with Red Hat distributions:
>
> https://support.oracle.com/CSP/main/article?cmd=show&type=NO
> T&id=1089399.1

3. Switch to the `root` user:

   ```
   $ su -
   ```

4. Install the following packages in sequence, where *version* is the version of the Oracle Automatic Storage Management library driver, *arch* is the system architecture, and *kernel* is the version of the kernel that you are using:

   ```
   oracleasm-support-version.arch.rpm
   oracleasm-kernel-version.arch.rpm
   oracleasmlib-version.arch.rpm
   ```

   Enter a command similar to the following to install the packages:

   ```
   # rpm -ivh oracleasm-support-version.arch.rpm \
           oracleasm-kernel-version.arch.rpm \
           oracleasmlib-version.arch.rpm
   ```

   For example, if you are using the Red Hat Enterprise Linux 5 AS kernel on an AMD64 system, then enter a command similar to the following:

   ```
   # rpm -ivh oracleasm-support-2.1.3-1.el5.x86_64.rpm \
           oracleasm-2.6.18-194.26.1.el5xen-2.0.5-1.el5.x86_64.rpm \
           oracleasmlib-2.0.4-1.el5.x86_64.rpm
   ```

5. Enter the following command to run the `oracleasm` initialization script with the `configure` option:

   ```
   # /usr/sbin/oracleasm configure -i
   ```

   > **Note:** The oracleasm command in `/usr/sbin` is the command you should use. The `/etc/init.d` path is not deprecated, but the `oracleasm` binary in that path is now used typically for internal commands.

6. Enter the following information in response to the prompts that the script displays:

Prompt	Suggested Response
Default user to own the driver interface:	**Standard groups and users configuration**: Specify the Oracle software owner user (for example, `oracle`)
	Job role separation groups and users configuration: Specify the Grid Infrastructure software owner (for example, `grid`)

Prompt	Suggested Response
Default group to own the driver interface:	**Standard groups and users configuration**: Specify the OSDBA group for the database (for example, `dba`).
	Job role separation groups and users configuration: Specify the OSASM group for storage administration (for example, `asmadmin`).
Start Oracle ASM Library driver on boot (y/n):	Enter `y` to start the Oracle Automatic Storage Management library driver when the system starts.
Scan for Oracle ASM disks on boot (y/n)	Enter `y` to scan for Oracle ASM disks when the system starts.

The script completes the following tasks:

- Creates the `/etc/sysconfig/oracleasm` configuration file

- Creates the `/dev/oracleasm` mount point

- Mounts the ASMLIB driver file system

> **Note:** The Oracle ASMLIB file system is not a regular file system. It is used only by the Oracle ASM library to communicate with the Oracle ASMLIB.

7. Enter the following command to load the `oracleasm` kernel module::

```
# /usr/sbin/oracleasm init
```

Configuring Disk Devices to Use Oracle ASMLIB To configure the disk devices to use in an Oracle Automatic Storage Management disk group, perform the following steps:

1. If you intend to use IDE, SCSI, or RAID devices in the Oracle Automatic Storage Management disk group, then perform the following steps:

a. Install or configure the disk devices that you intend to use for the disk group and restart the system.

b. Enter the following command to identify the device name for the disks to use:

```
# /sbin/fdisk -l
```

Depending on the type of disk, the device name can vary:

Disk Type	Device Name Format	Description
IDE disk	`/dev/hd`*xn*	In this example, *x* is a letter that identifies the IDE disk, and *n* is the partition number. For example, `/dev/hda` is the first disk on the first IDE bus.
SCSI disk	`/dev/sd`*xn*	In this example, *x* is a letter that identifies the SCSI disk, and *n* is the partition number. For example, `/dev/sda` is the first disk on the first SCSI bus.

Disk Type	Device Name Format	Description
RAID disk	`/dev/rd/cxdypz` `/dev/ida/cxdypz`	Depending on the RAID controller, RAID devices can have different device names. In the examples shown, *x* is a number that identifies the controller, *y* is a number that identifies the disk, and *z* is a number that identifies the partition. For example, `/dev/ida/c0d1` is the second logical drive on the first controller.

To include devices in a disk group, you can specify either whole-drive device names or partition device names.

> **Note:** Oracle recommends that you create a single whole-disk partition on each disk to use.

c. Use either `fdisk` or `parted` to create a single whole-disk partition on the disk devices.

2. Enter a command similar to the following to mark a disk as an Oracle Automatic Storage Management disk:

```
# /usr/sbin/oracleasm createdisk DISK1 /dev/sdb1
```

In this example, `DISK1` is a name assigned to the disk.

> **Note:**
>
> - The disk names you specify can contain uppercase letters, numbers, and the underscore character. They must start with an uppercase letter.
>
> - To create a database during the installation using the Oracle Automatic Storage Management library driver, you must change the disk discovery string to `ORCL:*`.
>
> - If you are using a multi-pathing disk driver with Oracle ASM, then make sure that you specify the correct logical device name for the disk.

Administering Oracle ASMLIB and Disks To administer the Oracle Automatic Storage Management library driver and disks, use the `/usr/sbin/oracleasm` initialization script with different options, as described in the following table:

Table D–1   Disk Management Tasks Using ORACLEASM

Task	Command Example	Description
Configure or reconfigure ASMLIB	`oracleasm configure -i`	Use the configure option to reconfigure the Oracle Automatic Storage Management library driver, if necessary.
		To see command options, enter `oracleasm configure` without the `-i` flag.
Change system restart load options for ASMLIB	`oracleasm enable`	Options are `disable` and `enable`.
		Use the `disable` and `enable` options to change the actions of the Oracle Automatic Storage Management library driver when the system starts. The `enable` option causes the Oracle Automatic Storage Management library driver to load when the system starts.
Load or unload ASMLIB without restarting the system	`oracleasm restart`	Options are `start`, `stop` and `restart`.
		Use the `start`, `stop`, and `restart` options to load or unload the Oracle Automatic Storage Management library driver without restarting the system.
Mark a disk for use with ASMLIB	`oracleasm createdisk VOL1 /dev/sda1`	Use the `createdisk` option to mark a disk device for use with the Oracle Automatic Storage Management library driver and give it a name, where *labelname* is the name you want to use to mark the device, and *devicepath* is the path to the device:
		`oracleasm createdisk` *labelname devicepath*
Unmark a named disk device	`oracleasm deletedisk VOL1`	Use the `deletedisk` option to unmark a named disk device, where *diskname* is the name of the disk:
		`oracleasm deletedisk` *diskname*
		Caution: Do not use this command to unmark disks that are being used by an Oracle Automatic Storage Management disk group. You must delete the disk from the Oracle Automatic Storage Management disk group before you unmark it.
Determine if ASMLIB is using a disk device	`oracleasm querydisk`	Use the `querydisk` option to determine if a disk device or disk name is being used by the Oracle Automatic Storage Management library driver, where *diskname_devicename* is the name of the disk or device that you want to query:
		`oracleasm querydisk` *diskname_devicename*

Table D–1 (Cont.) Disk Management Tasks Using ORACLEASM

Task	Command Example	Description
List Oracle ASMLIB disks	`oracleasm listdisks`	Use the `listdisks` option to list the disk names of marked Oracle ASM library driver disks.
Identify disks marked as ASMLIB disks	`oracleasm scandisks`	Use the `scandisks` option to enable cluster nodes to identify which shared disks have been marked as ASMLIB disks on another node.

Rename ASMLIB disks	`oracleasm renamedisk VOL1 VOL2`	Use the renamedisk option to change the label of an Oracle ASM library driver disk or device by using the following syntax, where *manager* specifies the manager device, *label_device* specifies the disk you intend to rename, as specified either by OracleASM label name or by the device path, and *new_label* specifies the new label you want to use for the disk:
		
		`oracleasm renamedisk [-l manager] [-v]`
		`label_device new_label`
		
		Use the `-v` flag to provide a verbose output for debugging.
		
		Caution: You must ensure that all Oracle Database and Oracle ASM instances have ceased using the disk before you relabel the disk. If you do not do this, then you may lose data.

Deinstalling Oracle ASMLIB

If Oracle ASMLIB is installed but you do not use it for device path persistence, then deinstall Oracle ASMLIB:

1. Stop Oracle ASM and any running database instance:

   ```
   $ srvctl stop asm
   $ srvctl stop instance -d db_unique_name
   ```

2. Log in as `root`.

3. Stop the Oracle Restart stack:

   ```
   # cd Grid_home/bin
   # crsctl stop has
   ```

4. Stop Oracle ASMLIB:

   ```
   # /etc/init.d/oracleasm disable
   ```

5. Remove the `oracleasm` library and tools RPMs:

   ```
   # rpm -e oracleasm-support
   # rpm -e oracleasmlib
   ```

6. Check if any `oracleasm` RPMs are remaining:

   ```
   # rpm -qa| grep oracleasm
   ```

7. If any `oracleasm` configuration files are remaining, remove them:

   ```
   # rpm -qa| grep oracleasm | xargs rpm -e
   ```

   Oracle ASMLIB and associated RPMs are now removed.

8. Start the Oracle Restart stack. Optionally, you can install and configure Oracle ASM Filter Driver (Oracle ASMFD) before starting the Oracle Restart stack.

> **See Also:** *Oracle Automatic Storage Management Administrator's Guide*

Configuring Disk Devices Manually for Oracle Automatic Storage Management

This section contains the following information about preparing disk devices manually for use by Oracle ASM:

- About Device File Names and Ownership for Linux
- Configuring a Permissions File for Disk Devices for Oracle ASM

> **Note:** The operation of `udev` depends on the Linux version, vendor, and storage configuration.

About Device File Names and Ownership for Linux

By default, the device file naming scheme `udev` dynamically creates device file names when the server is started, and assigns ownership of them to `root`. If `udev` applies default settings, then it changes Oracle device file names and owners for the disks, making the disks inaccessible when the server is restarted. For example, a voting disk on a device named `/dev/sdd` owned by the user `grid` may be on a device named `/dev/sdf` owned by `root` after restarting the server.

If you use Oracle ASMFD, then you do not have to ensure permissions and device path persistency in `udev`.

If you do not use Oracle ASMFD, then you must create a custom rules file. Linux vendors customize their udev configurations and use different orders for reading rules files. For example, on some Linux distributions when `udev` is started, it sequentially carries out rules (configuration directives) defined in rules files. These files are in the path `/etc/udev/rules.d/`. Rules files are read in lexical order. For example, rules in the file `10-wacom.rules` are parsed and carried out before rules in the rules file `90-ib.rules`.

When specifying the device information in the UDEV rules file, ensure that the OWNER, GROUP and MODE are specified before all other characteristics in the order shown. For example, to include the characteristic ACTION on the UDEV line, specify ACTION after OWNER, GROUP, and MODE.

Where rules files describe the same devices, on the supported Linux kernel versions, the last file read is the one that is applied.

Configuring a Permissions File for Disk Devices for Oracle ASM

To configure a permissions file for disk devices for use by Oracle ASM, complete the following tasks:

1. Configure SCSI devices as trusted devices, while listed, by editing the `/etc/scsi_id.config` file and adding "options=-g" to the file. For example:

```
# cat > /etc/scsi_id.config
vendor="ATA",options=-p 0x80
options=-g
```

2. Using a text editor, create a `udev` rules file for the Oracle ASM devices, setting permissions to 0660 for the installation owner and the group whose members are administrators of the grid infrastructure software.

   For example, on Oracle Linux, to create a role-based configuration `rules.d` file, where the installation owner is `grid`, and the OSASM group is `asmadmin`, enter commands similar to the following:

   ```
   # vi /etc/udev/rules.d/99-oracle-asmdevices.rules

   KERNEL=="sd?1", BUS=="scsi", PROGRAM=="/sbin/scsi_id",
   RESULT=="14f70656e66696c00000000", OWNER="grid", GROUP="asmadmin", MODE="0660"
   KERNEL=="sd?2", BUS=="scsi", PROGRAM=="/sbin/scsi_id",
   RESULT=="14f70656e66696c00000000", OWNER="grid", GROUP="asmadmin", MODE="0660"
   KERNEL=="sd?3", BUS=="scsi", PROGRAM=="/sbin/scsi_id",
   RESULT=="14f70656e66696c00000000", OWNER="grid", GROUP="asmadmin", MODE="0660"
   ```

3. Load updated block device partition tables on the server using: `/sbin/partprobe` *devicename*. You must do this as the `root` user.

4. Enter the command to restart the `udev` service.

   On Oracle Linux, and Red Hat Enterprise Linux, the commands are:

   ```
   # /sbin/udevcontrol reload_rules
   # /sbin/start_udev
   ```

   On SUSE Linux Enterprise Server, the command is:

   ```
   # /etc/init.d boot.udev restart
   ```

   Verify that the device permissions and ownerships are set correctly.

E

Configuring Networks for Oracle Database

Typically, the computer on which you want to install Oracle Database is connected to the network. The computer has local storage to store the Oracle Database installation. It also contains a display monitor and DVD drive. This appendix describes how to install Oracle Database on computers that do not meet the typical scenario. It describes the following cases:

- Installing on Multihomed Computers
- Installing on Computers with Multiple Aliases
- Installing on Non-Networked Computers

> **See Also:** The Certify pages on My Oracle Support for the latest information about supported network protocols and hardware for Oracle Database:
>
> https://support.oracle.com/

Installing on Multihomed Computers

You can install Oracle Database on a multihomed computer. A multihomed computer is associated with multiple IP addresses. This is typically achieved by having multiple network cards on the computer. Each IP address is associated with a host name. In addition, you can set up aliases for the host name. By default, Oracle Universal Installer uses the `ORACLE_HOSTNAME` environment variable setting to find the host name. If `ORACLE_HOSTNAME` is not set and you are installing on a computer that has multiple network cards, then Oracle Universal Installer determines the host name from the `/etc/hosts` file.

Clients must be able to access the computer either by using this host name or by using aliases for this host name. To verify, ping the host name from the client computers using the short name (host name only) and the full name (host name and domain name).

> **Note:** Both tests must be successful.

Setting the ORACLE_HOSTNAME Environment Variable

Use the following procedure to set the `ORACLE_HOSTNAME` environment variable. For example, if the fully qualified host name is `somehost.example.com`, then enter one of the following commands:

In Bourne, Bash, or Korn shell:

```
$ ORACLE_HOSTNAME=somehost.example.com
```

```
$ export ORACLE_HOSTNAME
```

In C shell:

```
% setenv ORACLE_HOSTNAME somehost.example.com
```

Installing on Computers with Multiple Aliases

A computer with multiple aliases is registered with the naming service under a single IP but with multiple aliases. The naming service resolves any of those aliases to the same computer. Before installing Oracle Database on such a computer, set the `ORACLE_HOSTNAME` environment variable to the computer whose host name you want to use.

Installing on Non-Networked Computers

You can install Oracle Database on a non-networked computer. If the computer, such as a laptop, is configured for DHCP and you plan to connect the computer to the network after the Oracle Database installation, then use the `ping` command on the computer on which you want to install the database to check if the computer can connect to itself. Perform this step by first using only the host name and then using the fully qualified name, which should be in the `/etc/hosts` file.

> **Note:** When you run the `ping` command on the computer itself, the `ping` command should return the IP address of the computer.

If the `ping` command fails, then contact your network administrator.

Connecting the Computer to the Network after Installation

If you connect the computer to a network after installation, then the Oracle Database instance on the computer can work with other instances on the network. The computer can use a static IP or DHCP, depending on the network to which you are connected.

F

Optimal Flexible Architecture

This appendix describes the Optimal Flexible Architecture (OFA) standard. This standard is a set of configuration guidelines created to ensure well organized Oracle installations that are easier to maintain. It includes information about the following topics:

- Overview of the Optimal Flexible Architecture Standard
- Understanding Optimal Flexible Architecture

Overview of the Optimal Flexible Architecture Standard

The Optimal Flexible Architecture standard helps you to organize database software and configure databases to allow multiple databases, of different versions, owned by different users to coexist. Optimal Flexible Architecture assists in identification of `ORACLE_BASE` with its Automatic Diagnostic Repository (ADR) diagnostic data to properly collect incidents.

All Oracle components on the installation media are compliant with Optimal Flexible Architecture. Oracle Universal Installer places Oracle Database components in directory locations, assigning the default permissions that follow Optimal Flexible Architecture guidelines.

Oracle recommends that you use Optimal Flexible Architecture, especially if the database is huge, or if you plan to have multiple databases.

Advantages of Multiple Oracle Homes and OFA

When you install Oracle database, you are installing a large application that your computer can support. Using multiple Oracle homes and Optimal Flexible Architecture provides many advantages when administering large databases. The following advantages are important:

- Structured organization of directories and files, and consistent naming for database files simplify database administration.
- Distribution of I/O across multiple disks prevents performance bottlenecks caused by multiple read or write commands issued simultaneously to a single drive.
- Distribution of applications across multiple disks safeguards against database failures.
- Login home directories are not at risk when database administrators add, move, or delete Oracle home directories.
- Multiple databases, of different versions, owned by different users can coexist concurrently.

- Software upgrades can be tested in an Oracle home in a separate directory from the Oracle home where your production database is located.

Understanding Optimal Flexible Architecture

This section describes the naming strategy recommended by the Optimal Flexible Architecture standard. It contains the following sections:

- File Systems
- Naming Directories
- Naming Database Files
- Separating Segments with Different Requirements
- Identifying the Optimal Flexible Architecture Structure for Oracle Files
- Optimal Flexible Architecture File Mapping

File Systems

The following sections describe the conventions for mount points:

- Number of File Systems
- Naming Conventions

Number of File Systems

To fully implement the Optimal Flexible Architecture recommendations for a database stored on file systems that are not striped or mirrored, you require at least three file systems located on separate physical devices.

Naming Conventions

Name all file system mount points using the syntax /*pm*, where *p* is a string constant and *m* is a unique fixed-length key (typically a two-digit number) used to distinguish each mount point. For example: /u01 and /u02, or /disk01 and /disk02.

Naming Directories

The following sections describe the naming conventions for directories that are compliant with the Optimal Flexible Architecture standard:

- Oracle Base Directory Naming Convention
- Naming Mount Points for Very Large Databases (VLDBs)
- Referring to Path Names
- Oracle Home Directory Naming Convention
- Naming Subdirectories

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

Oracle Base Directory Naming Convention

The Oracle Base directory is the top level directory that you can use to install the various Oracle software products. You can use the same Oracle base directory for multiple installations. If different operating system users install Oracle software on the same system, then each user must create a separate Oracle base directory.

Name Oracle base directories using the syntax */pm/s/u*. Table F–1 describes the variables used in this syntax.

Table F–1   Syntax for Naming Oracle Base Directories

Variable	Description
pm	A mount point name
s	A standard directory name
u	The name of the owner of the directory (the user running Oracle Universal Installer)

For example, `/u01/app/oracle` is an Oracle base directory created by the `oracle` user and `/u01/app/applmgr` is an Oracle base directory created by the `applmgr` user.

Placing Oracle base directories at the same level in the UNIX file system is advantageous because it enables you to refer to the collection of Oracle base directories on different mount points using a single pattern matching string, `/*/app/*`.

Naming Mount Points for Very Large Databases (VLDBs)

If each disk drive contains database files from one application and there are enough drives for each database to prevent I/O bottlenecks, use the syntax */h/q/d* for naming mount points. Table F–2 describes the variables used in this syntax.

Table F–2   Syntax for Naming Mount Points for Very Large Databases

Variable	Description
h	Oracle base directory
q	A string denoting that Oracle data is stored in this directory, for example, `oradata`
d	The value of the initialization parameter `DB_NAME` (typically the same as the instance `SID` for single-instance databases)

For example, to allocate two drives exclusively for the `test` database, name the mount points `/u01/app/oracle/oradata/test` and `/u02/app/oracle/oradata/test`.

Referring to Path Names

Refer to explicit path names only in files designed specifically to store them, such as the password file, `/etc/passwd`, and the Oracle `oratab` file. Refer to group memberships only in the `/etc/group` file.

Oracle Home Directory Naming Convention

To help fulfill the Optimal Flexible Architecture requirement of simultaneously running multiple versions of Oracle software, install the software in a directory matching the pattern */pm/s/u*`/product/`*v/type_*`[`*n*`]`.

Table F–3 describes the variables used in this syntax.

Table F–3 Syntax for Naming Oracle Home Directories

Variable	Description
pm	A mount point name
s	A standard directory name
u	The name of the owner of the directory
v	The version of the software
type	The type of installation, for example Database (`dbhome_1`), Client (`client`), or Oracle Grid Infrastructure (`grid`)
n	An optional counter, which enables you to install the same product more than once in the same Oracle base directory

For example:

`/u01/app/oracle/product/12.1.0/dbhome_1` indicates the Oracle home directory for the first installation of Oracle Database on this system.

The `ORACLE_HOME` environment variable is set to the Oracle home directory.

Naming Subdirectories

To facilitate the organization of administrative data, Oracle recommends that you store database-specific administration files in subdirectories matching the pattern `/h/admin/d/a/`, where *h* is the Oracle base directory, *d* is the database name (DB_NAME), and *a* is a subdirectory for specific types of database administration files. Table F–4 describes the database administration file subdirectories.

Table F–4 Subdirectories for Database Administration Files

Subdirectory	Description
`arch`	Archived redo log files
`adump`	Audit files (Set the `AUDIT_FILE_DEST` initialization parameter to specify the `adump` directory) **See Also:** *Oracle Database Security Guide*
`create`	Contains database creation log files and scripts used to create the database
`dpdump`	Default directory for data pump operations. Also contains the data pump file `dp.log`
`exp`	Database export files
`logbook`	Files recording the status and history of the database
`pfile`	Instance parameter files
`scripts`	Ad hoc SQL scripts

For example, `/u01/app/oracle/admin/orcl/scripts/` is the `scripts` subdirectory associated with the database named `orcl`.

The ADR diagnostic data goes into the `/h/diag/rdbms/d/i/` directory by default.

 where

h is Oracle Base

d is the database name

i is the instance name.

The ADR home has the trace, alert, and incident sub-directories. Table F–5 describes the ADR directories.

Table F–5   Locations for Diagnostic Traces

Diagnostic Data	10*g* Location	11*g* and 12*c* Location
Foreground Process traces	`user_dump_dest`	*ADR_HOME*`/trace/`
Background Process traces	`background_dump_dest`	*ADR_HOME*`/trace/`
Alert Log Data	`background_dump_dest`	*ADR_HOME*`/alert/`
Core Dump	`core_dump_dest`	*ADR_HOME*`/cdumps/`
Incident Dumps	`user_dump_dest` or `background_dump_dest` depending on the process	*ADR_HOME*`/incident/`

> **See Also:** "Structure, Contents, and Location of the Automatic Diagnostic Repository" in *Oracle Database Administrator's Guide*

Naming Database Files

The following table lists the recommended file naming conventions for database files:

> **Note:** Oracle Managed Files (OMF) and files stored in Oracle Automatic Storage Management disk groups use different naming conventions. For more information about these naming conventions, refer to the *Oracle Database Administrator's Guide*.

File Type	File Naming Convention
Control files	`/`*h*`/`*q*`/`*d*`/control.ctl`
Redo log files	`/`*h*`/`*q*`/`*d*`/redon.log`
Data files	`/`*h*`/`*q*`/`*d*`/`*tn*`.dbf`

The following table describes this syntax:

Variable	Description
h	Oracle base directory
q	A string (typically `oradata`) distinguishing Oracle data from all other files
d	The value of the `DB_NAME` initialization parameter (typically, the same as the instance `SID` for single-instance databases)
t	An Oracle tablespace name
n	A two-digit string

> **Note:** Do not store files other than control files, redo log files, or data files associated with database *d* in the path `/`*h*`/`*q*`/`*d*.

Using this convention, it is easy to determine the database to which the `/u01/app/oracle/oradata/sab/system01.dbf` file belongs.

Separating Segments with Different Requirements

Separate groups of segments with different lifespans, I/O request demands, and backup frequencies across different tablespaces.

Table F–6 describes the special tablespaces that the Database Configuration Assistant creates for each Oracle database. If you manually create a database, you must create the required tablespaces. These tablespaces are in addition to those required for application segments.

> **See Also:** "Creating and Configuring an Oracle Database" in *Oracle Database Administrator's Guide* for information about creating databases manually

Table F–6   Special Tablespaces

Tablespace	Required	Description
EXAMPLE	No	The EXAMPLE tablespace used to store the Sample Schemas
SYSAUX	Yes	Auxiliary tablespace to the SYSTEM tablespace
SYSTEM	Yes	Data dictionary segments
TEMP	Yes	Temporary segments
UNDOTBS1	Yes	Used by Oracle to store undo information
USERS	No	Miscellaneous user segments

Creating these special tablespaces is effective because data dictionary segments are never dropped, and no other segments that can be dropped are allowed in the SYSTEM tablespace.

> **See Also:** "Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files" on page 9-9 for information about redo log, and control files

Identifying the Optimal Flexible Architecture Structure for Oracle Files

Table F–7 describes the syntax used for identifying classes of files.

Table F–7   Directory Structure Syntax for Identifying Classes of Files

Directory Structure Syntax	Description
/u[0-9][0-9]	User data directories
/*/home/*	User home directories
/*/app/*	User application software directories
/*/app/applmgr	Oracle applications software subtrees
/*/app/oracle/product	Oracle software subtrees
/*/app/oracle/product/12.1.0	Oracle software subtree for release 12*c* products
/*/app/oracle/product/12.1.0/db*	Oracle home directories for Oracle Database 12*c*
/*/app/oracle/product/12.1.0/grid*	Oracle home directory for Oracle Grid Infrastructure 12*c* for a standalone server, for user oracle

Table F–7   (Cont.) Directory Structure Syntax for Identifying Classes of Files

Directory Structure Syntax	Description
/*/app/oracle/admin/orcl	orcl database administrative subtrees
/*/app/oracle/admin/orcl/arch/*	orcl database archived log files
/*/app/oracle/oradata	Oracle data directories
/*/app/oracle/oradata/orcl/*	orcl database files
/*/app/oracle/oradata/orcl/*.log	orcl database redo log files

Optimal Flexible Architecture File Mapping

Table F–8 shows a hierarchical file mapping of a sample Optimal Flexible Architecture-compliant installation with two Oracle home directories and two databases. The database files are distributed across three mount points, /u02, /u03, and /u04.

> **Note:**   Oracle recommends that you use Oracle ASM to provide greater redundancy and throughput.

Table F–8   Hierarchical File Mapping for an Optimal Flexible Architecture Installation

Directory	Description
/	Root directory
/u01/	User data mount point 1
/u01/app/	Subtree for application software
/u01/app/oracle/	Oracle Base directory
/u01/app/oracle/admin/	Subtree for database administration files
/u01/app/oracle/admin/TAR	Subtree for support log files
/u01/app/oracle/admin/*db_name1*/	admin subtree for *db_name1* database
/u01/app/oracle/admin/*db_name2*/	admin subtree for *db_name2* database
/u01/app/oracle/fast_recovery_area/	Subtree for recovery files
/u01/app/oracle/fast_recovery_area/*db_name1*	Recovery files for *db_name1* database
/u01/app/oracle/fast_recovery_area/*db_name2*	Recovery files for *db_name2* database
/u02/app/oracle/oradata	
/u03/app/oracle/oradata	Oracle data directory
/u04/app/oracle/oradata	
/u01/app/oracle/product/	Distribution files
/u01/app/oracle/product/12.1.0/dbhome_1	Oracle home directory for Oracle Database, for user oracle
/u01/app/oracle/product/12.1.0/grid	Oracle home directory for Oracle Grid Infrastructure for a standalone server, for user oracle
/u01/app/kjf/	Oracle base directory for user kjf
/u01/app/edm/	Oracle base directory for user edm

Table F–9 shows a hierarchical file mapping for log files of a sample Optimal Flexible Architecture-compliant installation in the `orcl` database.

Table F–9   Hierarchical File Mapping for Log Files in an Optimal Flexible Architecture Installation

Directory	Description
/u01/app/oracle/admin/TAR	Subtree for support log files
/u01/app/oracle/admin/orcl/arch/*	Archived log files
/u01/app/oracle/admin/orcl/create/	Contains the database creation log files
/u01/app/oracle/oradata/orcl/*.log	Redo log files
/u01/app/oracle/admin/orcl/dpdump/	Contains the data pump file `dp.log`
/u01/app/oracle/diag	Contains all database, listener, sqlnet and other diagnostic logs
/u01/app/oracle/audit	Contains all audit logs
/u01/app/oracle/cfgtoollogs	Contains logs for configuration assistants such as Oracle Database Configuration Assistant, Database Upgrade Assistant, and Oracle Net Configuration Assistant

G

Managing Oracle Database Port Numbers

During installation, Oracle Universal Installer 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

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 ensure that the same port number is not used by two services on your host. Enter the following command to identify the ports currently used on your computer:

```
$/bin/netstat -a
```

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 /etc/services file?

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 `$ORACLE_HOME/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 single instance database installation. By default, the first port in the range is assigned to the component, if it is available.

Table G–1   Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle Net Services Listener Enables Oracle client connections to the database over the Oracle Net Services protocol. You can configure it during installation. To reconfigure this port, use Net Configuration Assistant.	1521	Port number changes to the next available port. Modifiable manually to any available port.	TCP
Oracle Connection Manager Listening port for Oracle client connections to Oracle Connection Manager. It is not configured during installation, but can be configured manually by editing the `cman.ora` parameter file. You can find the file under `/network/admin` directory.	1630	1630	TCP
Oracle XML DB The Oracle XML DB HTTP port is used if web-based applications must access an Oracle database from an HTTP listener. You must configure this port manually. **See Also**: "Using HTTP(S) on a Standard Port Instead of an Oracle XML DB Default Port" in *Oracle XML DB Developer's Guide*	0	Configured Manually	HTTP
Oracle XML DB The Oracle XML DB FTP is used when applications must access an Oracle database from an FTP listener. You must configure this port manually. **See Also**: "Using FTP on the Standard Port Instead of the Oracle XML DB Default Port" in *Oracle XML DB Developer's Guide*	0	Configured Manually	FTP
Cluster Synchronization Service (CSS) CSS daemon internode connection for the GM layer. The port number is assigned automatically. You cannot view or modify it.	42424	Dynamic	UDP
Oracle Cluster Registry The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	UDP

See Also:

- *Oracle Enterprise Manager Cloud Control Advanced Installation and Configuration Guide* for information on Oracle Management Agent ports

- *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for a list of clusterware ports used in Oracle components

H

Configuring Oracle Database Globalization Support

This appendix describes the following Globalization Support topics:

- Installing and Using Oracle Components in Different Languages
- Running Oracle Universal Installer in Different Languages

> **See Also:** *Oracle Database Globalization Support Guide* for an overview of globalization support for Oracle Database

Installing and Using Oracle Components in Different Languages

This section describes the following procedures:

- Configuring Oracle Components to Run in Different Languages
- Installing Translation Resources

Configuring Oracle Components to Run in Different Languages

You can specify the language and the territory, or locale, in which you want to use Oracle components. The locale setting of a component determines the language of the user interface of the component and the globalization behavior, such as date and number formatting. Depending on the Oracle component, the locale of the component is either inherited from the operating system session that started the component, or is defined by the `NLS_LANG` environment variable.

The operating system locale usually influences Oracle components that are based on Java technology. The `NLS_LANG` environment variable usually influences Oracle components that use Oracle Client libraries such as OCI.

> **Note:** The user interface of an Oracle component is displayed in a selected language only if the appropriate translation is available and has been installed. Else, the user interface is displayed in English.

This section describes the following procedures:

- Determining the Operating System Locale by Using the LANG Environment Variable
- Configuring Locale and Character Sets Using NLS_LANG

Determining the Operating System Locale by Using the LANG Environment Variable

The locale setting of your operating system session determines the language of the user interface and the globalization behavior for components such as Oracle Universal Installer, Oracle Net Configuration Assistant, and Oracle Database Configuration Assistant. It also determines the globalization behavior of Oracle Database sessions created by a user application through Oracle JDBC driver, unless overridden by the application.

The operating system locale is determined by the value of the LANG environment variable. Depending on your desktop environment, you can select a default session locale on a login screen, in a configuration panel, or in a configuration file.

> **Note:** Refer to the operating system documentation on how to select a locale for the operating system session in your desktop environment.

You can modify the LANG variable in the environment of your shell to start an Oracle component in a selected language. For example, to start Oracle Database Configuration Assistant in German, enter one of the following commands:

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

  ```
  $ LANG=de_DE.iso88591 dbca
  ```

- C shell (csh):

  ```
  % (setenv LANG de_DE.iso88591; dbca)
  ```

> **Note:** The LC_ALL environment variable overrides the value of the LANG environment variable. For the commands listed in this section to work, either ensure that the LC_ALL environment variable is not set in the environment, or substitute LC_ALL for LANG.

To modify the operating system locale for all Oracle components started by the given shell, modify the LANG variable using one of the following commands:

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

  ```
  $ LANG=de_DE.iso88591; export LANG
  $ ...
  ```

- C shell (csh):

  ```
  % setenv LANG de_DE.iso88591
  $ ...
  ```

The value of the LANG environment variable must be a valid operating system locale. To see the list of valid locales, enter the following command:

```
$ locale -a
```

> **Note:** Refer to the operating system documentation for a mapping between values of the LANG environment variable and the languages and territories that they represent.

Configuring Locale and Character Sets Using NLS_LANG

The NLS_LANG environment variable determines the language of the user interface and the globalization behavior for components such as SQL*Plus, exp, and imp. It sets the language and territory used by the client application and the database user session. It also declares the character set for entering and displaying data by the client application.

The NLS_LANG environment variable uses the following format:

```
NLS_LANG=language_territory.characterset
```

In this format:

- *language* specifies the language used for displaying Oracle messages, sorting, day names, and month names

- *territory* specifies the conventions for default date, monetary, and numeric formats

- *characterset* specifies the encoding used by the client application

  In most cases, this is the Oracle character set that corresponds to the character set of the user terminal or the operating system.

The NLS_LANG environment variable is set as a local environment variable for the shell on all UNIX-based platforms. For example, if the operating system locale setting is en_US.UTF-8, then the corresponding value of NLS_LANG environment variable is AMERICAN_AMERICA.AL32UTF8.

> **See Also:** "Setting Up a Globalization Support Environment" in *Oracle Database Globalization Support Guide* for information about the NLS_LANG parameter and Globalization Support initialization parameters

The following examples illustrate some valid values for the NLS_LANG environment variable.

> **Note:** Refer to the operating system 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

Installing Translation Resources

To view the user interface of Oracle components in different languages, you must install the appropriate language translations along with the component.

> **Note:** Part of Oracle Database Vault user interface text is stored in database tables in the DVSYS schema. By default, only the English language is loaded into these tables. You can use Oracle Database Vault Configuration Assistant to add more languages to Oracle Database Vault. For the necessary steps, refer to "Adding Languages to Oracle Database Vault" in *Oracle Database Vault Administrator's Guide*.

To install the translation resources:

1. Start Oracle Universal Installer.

2. In the Configure Security Updates screen enter the relevant information and click **Next.**

3. In the Select Installation Option screen, select the installation option and click **Next.**

4. In the System Class screen, select the type of system class for installing the database, and click **Next.**

5. In the Grid Installation Options screen, select the type of database installation you want to perform, and click **Next.**

6. In the Select Product Languages screen, select the language in which you want to use Oracle components from the Available Languages field.

   > **Note:** The Available Languages field lists all languages supported by Oracle globalization libraries. The set of languages for which a translation is actually available is usually smaller and depends on a particular component. The scope of translation for a given component may differ between languages. For example, some translations may include all user interface text, while others may include only error messages and no help files.

7. Use the **>** arrow to move the selected language to the Selected Languages field, and then click **Next.**

   > **Note:** Oracle Universal Installer ignores languages in the Selected Languages field for which no translation is available.

Running Oracle Universal Installer in Different Languages

Your operating system locale determines the language in which Oracle Universal Installer runs. Oracle Universal Installer may run in one of the following languages:

- Brazilian Portuguese (pt_BR)

- French (fr)

- German (de)

- Italian (it)

- Japanese (ja)

- Korean (ko)

- Simplified Chinese (zh_CN)

- Spanish (es)

- Traditional Chinese (zh_TW)

To run Oracle Universal Installer in an available language, change the locale in which your operating system session is running before you start Oracle Universal Installer with the `./runInstaller` command. If the selected language is not one of those that were listed earlier, then Oracle Universal Installer runs in English.

You must ensure that the selected value for the `LANG` environment variable starts with the appropriate language abbreviation. In the aforementioned list of languages, in which Oracle Universal Installer can run, the required abbreviation appears in parentheses beside the language name. For example, fr_FR and fr_CA are valid values to run the Oracle Universal Installer in French.

I

Troubleshooting

This appendix contains information about troubleshooting. It includes information about the following topics:

- Verify Requirements
- X Window Display Errors
- Remote Terminal Installation Error
- What to Do If an Installation Error Occurs?
- Reviewing the Log of an Installation Session
- Troubleshooting and Deconfiguring Oracle Restart
- Troubleshooting Host Name Changes and CSS
- Troubleshooting Configuration Assistants
- Troubleshooting Inventory Issues
- Troubleshooting Screen Display Issues
- Troubleshooting Memory Size Error
- Troubleshooting File Descriptors Error
- Silent-Mode Response File Error Handling
- Cleaning Up After a Failed Installation
- Continuing Installations or Upgrades After Server Restarts

> **See Also:** Chapter 6, "Troubleshooting Oracle Configuration Manager" in *Oracle Configuration Manager Installation and Administration Guide* for information about some errors that may occur while using Oracle Configuration Manager and tips to troubleshoot these errors

Verify 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 4.

Read the Release Notes

Read the release notes for the product before installing it. The latest version of the release notes is available at:

http://docs.oracle.com

X Window Display Errors

If you run Oracle Universal Installer on a remote system and you want to display Oracle Universal Installer's user interface on your local system, you might see error messages similar to the following:

```
"Failed to connect to server"
"Connection refused by server"
"Can't open display"
```

If you see any of these error messages, follow these steps:

> **Note:** This procedure applies only to users of UNIX workstations. If you are using a PC or other system with X server software installed, contact your X server vendor, system administrator, or refer to the X server documentation for information about how to permit remote systems to display X applications on the local system.

1. In a local terminal window, log in as the user that started the X Window session.

2. Enter the following command:

   ```
   $ xhost fully_qualified_remote_host_name
   ```

   For example:

   ```
   $ xhost somehost.example.com
   ```

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 shell:

     ```
     % setenv DISPLAY workstation_name:0.0
     ```

4. To determine if an X Window application displays correctly on the local system, enter the following command:

   ```
   $ xclock
   ```

   The X clock should appear on your monitor.

5. If the X clock appears, close the X clock and start Oracle Universal Installer again.

   > **See Also:** PC-X Server or operating system vendor documents for further assistance

Remote Terminal Installation Error

If you run the installation from a remote terminal, or if you use an su command to change users you might receive an error similar to the following:

```
Could not execute auto check for display colors using command
/usr/X11R6/bin/xdpyinfo
```

This can occur if the `DISPLAY` variable is not set, or the user running the installation is not authorized to open an X window. For instance, if you use an `su` command to change from a user that is authorized to open an X window to a user account that is not authorized to open an X window on the display, such as a lower-privileged user opening windows on the `root` user's console display.

To troubleshoot this issue, run the command `echo $DISPLAY` to ensure that the display variable is set to the correct visual or to the correct host. If the display variable is set correctly then either ensure that you are logged in as the user authorized to open an X window, or run the command `xhost +` to allow any user to open an X window.

What to Do If an Installation Error Occurs?

If you encounter an error during installation:

- Do not exit Oracle Universal Installer.

- If you click **Next** after you enter incorrect information on one of the installation screens, click **Back** to return to the screen and correct the information.

- If you encounter errors while Oracle Universal Installer is copying or linking files, then review the installation logs for more information.

  For copy file errors review:

  ```
  /u01/app/oraInventory/logs/timestamp for date of install.log
  /u01/app/oraInventory/logs/timestamp for date of install.err
  /u01/app/oraInventory/logs/timestamp for date of install.out
  ```

  For errors during linking review:

  ```
  $ORACLE_HOME/install/make.log
  ```

  If you encounter errors when you run the Oracle Universal Installer, then rerun the Oracle Universal Installer with the `-debug` option:

  ```
  $./runInstaller -debug
  ```

  Check the log file for details. Refer to "Reviewing the Log of an Installation Session" section on page I-3.

- If you encounter an error while a configuration assistant is running, refer to "Troubleshooting Configuration Assistants" section on page I-5.

- If you cannot resolve the problem, remove the failed installation by following the steps listed in the "Cleaning Up After a Failed Installation" section on page I-7.

Reviewing the Log of an Installation Session

During an installation, Oracle Universal Installer records all of 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, follow these steps:

1. If necessary, enter the following command to determine the location of the `oraInventory` directory:

   On Linux x86-64 and IBM: Linux on System z:

```
$ cat /etc/oraInst.loc
```

The `inventory_loc` parameter in this file specifies the location of the `oraInventory` directory.

2. Enter the following command to change directory to Oracle Universal Installer log file directory, where *orainventory_location* is the location of the `oraInventory` directory:

```
$ cd /orainventory_location/logs
```

3. Enter the following command to determine the name of the log file:

```
$ ls -ltr
```

Run these commands to list 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 the time when the installation started:

```
installActionsdate_time.log
oraInstalldate_time.err
oraInstalldate_time.out
```

4. To view the most recent entries in the log file, where information about a problem is most likely to appear, enter a command similar to the following:

```
$ tail -50 installActionsdate_time.log | more
```

This command displays the last 50 lines in the log file.

5. If the error displayed by Oracle Universal Installer or listed in the log file indicates a relinking problem, refer to the following file for more information:

```
$ORACLE_HOME/install/make.log
```

Troubleshooting and Deconfiguring Oracle Restart

Running the `roothas.sh` command flags `-deconfig -force` enables you to deconfigure Oracle Restart without removing installed binaries. This feature is useful if you encounter an error during an Oracle Grid Infrastructure for a standalone server installation, when running the `root.sh` command, such as a missing operating system package. By running `roothas.sh -deconfig -force` you can deconfigure Oracle Restart, correct the cause of the error, and then run `root.sh` again.

> **Note:** Stop any databases, services, and listeners that may be installed and running before deconfiguring Oracle Restart.

To deconfigure Oracle Restart:

1. Log in as the `root` user.

2. Go to the `Grid_home/crs/install` directory. For example:

```
# cd /u01/app/12.1.0/grid/crs/install
```

3. Run `roothas.sh` with the `-deconfig -force` flags. For example:

```
# roothas.sh -deconfig -force
```

> **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), the `roothas.sh` script replaces the `roothas.pl` script in the Oracle Grid Infrastructure home.

Troubleshooting Host Name Changes and CSS

If you change the host name for Oracle Automatic Storage Management (Oracle ASM), then the Oracle CSS daemon does not start. To solve this issue, perform the following steps:

1. Log in as the `root` user

2. Run `roothas.sh` to to deconfigure CSS:

```
# cd /u01/app/oracle/product/12.1.0/grid/crs/install
# perl roothas.sh -deconfig -force
```

   This removes any configuration on the system that referenced the old host name.

3. Run `root.sh` to reconfigure CSS using the new host name:

```
# cd /u01/app/oracle/product/12.1.0/grid
# ./root.sh
```

4. Go to the grid home's `bin` directory. Use the `srvctl add database` command with the `-c SINGLE` flag to add the database in an Oracle Restart configuration. Also use the `srvctl add` command to add the listener, the Oracle ASM instance, all Oracle ASM disk groups, and any database services to the Oracle Restart configuration.

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

   > **Note:** Starting with Oracle Database 12*c* Release 1 (12.1.0.2), the `roothas.sh` script replaces the `roothas.pl` script in the Oracle Grid Infrastructure home.

Troubleshooting Configuration Assistants

To troubleshoot an installation error that occurs when a configuration assistant is running:

- Review the installation log files listed in the "Reviewing the Log of an Installation Session" section on page I-3.

- Review the specific configuration assistant log file located in the `$ORACLE_HOME/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 "Irrecoverable Errors" on page I-6 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:

oraInventory_location/logs/installActions*date_time*.log

The execution status codes are listed in the following table:

Status	Result Code
Configuration assistant succeeded	0
Configuration assistant failed	1
Configuration assistant canceled	-1

Irrecoverable Errors

If you receive an irrecoverable 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 the "Cleaning Up After a Failed Installation" section on page I-7.

2. Correct the cause of the irrecoverable error.

3. Reinstall the Oracle software.

Troubleshooting Inventory Issues

If you face any of the following situations for Oracle home, then run the `opatch lsinventory -detail` command to list the contents of the inventory and see section "Recovering from inventory corruption" in the *Oracle Universal Installer User's Guide* for information about fixing the issue.

- Oracle home is cloned without completing the inventory steps.

- There is bad inventory.

- Inventory is not available but it is created when the Oracle Enterprise Manager Agent is installed in a separate Oracle home.

Troubleshooting Screen Display Issues

If you connect to Oracle database with a screen resolution of 640X480 or 800X600, then the Next button in the GUI is not visible as it hides behind the Taskbar. To fix this problem, perform one of the following:

- Hide the Taskbar.

- Move the Oracle Universal Installer screen up.

- Set the screen resolution to 1024X768 or higher.

Troubleshooting Memory Size Error

On Linux systems, if the operating system `/dev/shm` mount size is too small for the Oracle system global area (SGA) and program global area (PGA), it results in the following error:

```
ORA-00845: MEMORY_TARGET not supported on this system
```

Note that Memory Size (SGA and PGA), which sets the initialization parameter `MEMORY_TARGET` or `MEMORY_MAX_TARGET`, cannot be greater than the shared memory file system (`/dev/shm`) on your operating system.

The workaround, if you encounter the `ORA-00845` error, is to increase the `/dev/shm` mountpoint size.

For example:

```
# mount -t tmpfs shmfs -o size=7g /dev/shm
```

To make this change persistent across system restarts, add an entry in `/etc/fstab` similar to the following:

```
shmfs /dev/shm tmpfs size=7g 0 0
```

Troubleshooting File Descriptors Error

If file descriptors are not sized correctly, you see an error from various Oracle processes and potentially a `Linux Error EMFILE (Too many open files)` error in non-Oracle processes:

```
ORA-27123: unable to attach to shared memory segment
```

The workaround is that the number of file descriptors for each Oracle instance must be at least `512*PROCESSES`. The number of descriptors for each process must be at least 512.

Silent-Mode Response File Error Handling

To determine if a silent-mode installation succeeds or fails, refer to the following log file:

```
/oraInventory_location/logs/silentInstalldate_time.log
```

If necessary, refer to the previous section 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 runtime. If the validation fails, the silent-mode installation or configuration process ends.

Cleaning Up After a Failed Installation

If an installation fails, you must remove files that Oracle Universal Installer created during the attempted installation using the Deinstallation Tool.

For more information about how to run the Deinstallation Tool see Chapter 10, "Removing Oracle Database Software" and "Troubleshooting and Deconfiguring Oracle Restart" on page I-4

Continuing Installations or Upgrades After Server Restarts

During an Oracle Grid Infrastructure for a standalone server installation or upgrade, the server might require a restart and you may see errors similar to the following:

```
ACFS-9427 Failed to unload ADVM/ACFS drivers. A system reboot is recommended
ACFS-9428 Failed to load ADVM/ACFS drivers. A system reboot is recommended
```

The workaround is to perform the following steps:

1. Restart the computer.

2. Log in as `root`, and run the `orainstRoot.sh` script. For example:

```
$ sudo -s
# cd /u01/app/oraInventory
# ./orainstRoot.sh
```

3. Change directory to the Grid home and run the `root.sh` script. For example:

```
# cd /u01/app/oracle/product/12.1.0/grid
# ./root.sh
```

4. Configure a response file, and provide passwords for the installation. See "Postinstallation Configuration Using a Response File" on page A-9 for information about how to create the response file.

5. To complete the upgrade or installation, log in as the software installation owner and run the `configToolAllCommands` script, located in the path `$ORACLE_HOME/cfgtoollogs/configToolAllCommands`, specifying the response file that you created. For example, where the response file is `gridinstall.rsp`:

```
$ cd $ORACLE_HOME/cfgtoollogs/configToolAllCommands
$ ./configToolAllCommands RESPONSE_FILE=gridinstall.rsp
```

J

Frequently Asked Questions About Installation

Use the following guidelines to decide how to install Oracle Database components:

- Installing Oracle Database
- Installing Oracle Database Tools
- Installing Oracle Database with Oracle Applications
- Installing Oracle Database Heterogeneous Connectivity Tools (Gateways)

> **Note:** Some Oracle Database components may not be available on all platforms. Consult your platform-specific installation guide or release notes.

Installing Oracle Database

The following are frequently asked questions about installing Oracle database:

- I only need one instance of Oracle Database or I just want to install a test database to get familiar with the product. How do I install Oracle Database for these situations?
- How can I create an Oracle database that can handle transaction-heavy or data warehousing applications?
- What's the best way to install multiple Oracle databases?
- How do I configure client connections to an Oracle database?
- What is the best way to install Oracle Client if my client nodes have limited disk space?
- How do I upgrade Oracle Database?
- The computers at my site have been configured to run as a cluster. How should I install Oracle Database?
- How do I migrate my non-Oracle databases to Oracle Database?

I only need one instance of Oracle Database or I just want to install a test database to get familiar with the product. How do I install Oracle Database for these situations?

- If you want a quick installation using the default installation settings, then refer to the platform-specific *Oracle Database Quick Installation Guide*.

- If your site has special requirements, then refer to this guide for more information.

How can I create an Oracle database that can handle transaction-heavy or data warehousing applications?

If you want to create a starter database designed for transaction-heavy or data warehousing applications, then refer to this guide for more details. Select the **Advanced Installation** method, and then select the database type you want on the Select Database Configuration screen.

> **See Also:** *Oracle Database Data Warehousing Guide* after installation

Alternatively, you can use Oracle OLAP. The OLAP option is provided with Oracle Database Enterprise Edition. Oracle OLAP provides optimal support for database environments that must meet OLAP requirements.

> **See Also:**
>
> - *Oracle OLAP User's Guide*
> - *Oracle OLAP DML Reference*
> - *Oracle OLAP Java API Reference*

What's the best way to install multiple Oracle databases?

Use this guide to install Oracle Database using either of the following methods:

- **Installing with response files**: This method lets you run Oracle Universal Installer at a command line using a response file that contains settings specific to each computer.

- **Cloning a Database**: Install Oracle Database on one computer using interactive mode. You can also clone databases. Instructions for cloning databases are described in *Oracle Database Administrator's Guide*.

How do I configure client connections to an Oracle database?

1. Install Oracle Database on a server by using this guide for more information.

2. Use *Oracle Database Client Installation Guide* to install Oracle Client on each client node, and select the Instant Client installation type.

   If you have many client nodes, consider staging the software centrally, mapping the drive, and running Oracle Universal Installer in the silent or response file mode.

   If the client nodes only require a default installation into a new Oracle home directory, consider using this guide for more information.

What is the best way to install Oracle Client if my client nodes have limited disk space?

1. Install Oracle Database onto a server by using this guide for more details.

2. Use *Oracle Database Client Installation Guide* to install Oracle Client on each client node, and select the Instant Client installation type.

   If you have many client nodes, then consider running Oracle Universal Installer in silent or response file mode.

How do I upgrade Oracle Database?

Refer to *Oracle Database Upgrade Guide*.

> **See Also:** *Oracle Database Administrator's Guide* to use software cloning to upgrade Oracle Database

The computers at my site have been configured to run as a cluster. How should I install Oracle Database?

Use any of the following installation scenarios:

- If you want to run a single-instance Oracle Database in a clustered environment, then install Oracle Grid Infrastructure either before or after you install Oracle Database.

- If you want a consolidated pool of storage for all databases in a cluster, then install Oracle Grid Infrastructure, and use Oracle Automatic Storage Management (Oracle ASM) to manage this storage. Afterward, install Oracle Database (which can be either a single instance database or Real Application Clusters).

- If you plan to use Oracle Real Application Clusters, install Oracle Grid Infrastructure, and then install Oracle Real Application Clusters.

Refer to *Oracle Grid Infrastructure Installation Guide* and *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for your platform to install Oracle Grid Infrastructure and Oracle Real Application Clusters. Oracle Clusterware is installed in an Oracle Grid Infrastructure installation. You must install Oracle Real Application Clusters on an Oracle Clusterware cluster.

Oracle Clusterware is a key component required by Oracle Real Application Clusters installations. Oracle Clusterware is an integrated cluster management solution that can bind multiple servers to act as a single system. This is referred to as a cluster. It performs workload management and component restart. For example, when an instance supporting a particular service fails, Oracle Clusterware restarts the service on the next available instance that you have configured for that service. Oracle Clusterware can monitor non-Oracle programs, if they are defined within the Oracle Clusterware environment using the High Availability API.

How do I migrate my non-Oracle databases to Oracle Database?

Use Oracle SQL Developer to migrate your non-Oracle databases and applications to Oracle. Oracle SQL Developer software and documentation is available at:

http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index.html

Installing Oracle Database Tools

The following are frequently asked questions about installing Oracle database tools:

- How do I install Oracle WebLogic Server?

- How can I administer and monitor my Oracle Database products?

- How do I manage security for my Oracle Database products?

- How do I use Oracle Database to manage my XML data?

- Does Oracle Database provide OLAP tools so that I can analyze data such as trends and time series in my database?

- Does Oracle Database provide data mining tools that I can use to discover hidden meaning in my data and predict likely outcomes based on my data?

- How do I perform backup and recovery operations for Oracle Database?

- Is Oracle Workflow included with Oracle Database?

- Is there a migration plan for customers that have built solutions using Oracle Workflow?

How do I install Oracle WebLogic Server?

Refer to *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server*.

For more information on Oracle WebLogic Server refer to the product documentation at:

http://www.oracle.com/technetwork/middleware/weblogic/documentation/index.html

How can I administer and monitor my Oracle Database products?

To perform regular administrative functions such as creating, configuring, or deleting databases, or managing database templates, use one of the following methods:

To manage only the single database and listener that you are installing:

1. Use this guide to install Oracle Database.

2. From Oracle Database, use Database Configuration Assistant to manage your databases.

   You can also administer and monitor the database with Oracle Enterprise Manager Cloud Control.

   Oracle Enterprise Manager Cloud Control includes the Oracle Management Agent, Oracle Management Service, and Oracle Management Repository, and also Cloud Control, a browser-based central console through which administrators can perform all monitoring, administration, and configuration tasks for the enterprise.

   > **See Also:** *Oracle Enterprise Manager Cloud Control Advanced Installation and Configuration Guide* and *Oracle Enterprise Manager Cloud Control Basic Installation Guide* available at:
   >
   > http://docs.oracle.com

To perform advanced administration tasks, such as monitoring Oracle Database and managing multiple hosts, application servers, and databases including the one that you are installing, install Oracle Enterprise Manager as follows:

1. Use this guide to install Oracle Database.

   If you plan to use Oracle Real Application Clusters, then install Oracle Database by using the platform-specific *Oracle Grid Infrastructure Installation Guide* and *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

2. Use *Oracle Enterprise Manager Cloud Control Advanced Installation and Configuration Guide* to install and configure Oracle Enterprise Manager. For postconfiguration tasks, use *Oracle Enterprise Manager Cloud Control Administrator's Guide*. Refer to documentation available on the Oracle Technology Network website at:

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

How do I manage security for my Oracle Database products?

Oracle provides a wide range of security solutions for your enterprise environment, including centralized administration and security features integrated with Oracle Internet Directory. The set of Oracle security services called Oracle Platform Security

Services (OPSS) integrates the security features built into Oracle Database, Oracle WebLogic Server, and the Oracle Identity Management infrastructure. Combined, these features enable the development and deployment of secure e-business applications.

Oracle Identity Management includes Oracle Internet Directory, a centralized repository that simplifies administration of users and applications in the Oracle environment with the following components:

- Oracle Internet Directory client tools, including LDAP command-line tools, the Oracle Internet Directory SDK, and Oracle Directory Manager.

- Oracle Internet Directory server components, including the directory server, the directory replication server, the directory integration server, and various tools for starting and stopping them.

Oracle Database includes the Oracle Internet Directory client tools, but not the Oracle Internet Directory server components. To install the Oracle Internet Directory server components, see *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* and the Oracle Identity Management documentation at:

http://www.oracle.com/technetwork/middleware/id-mgmt/overview/index.html

> **See Also:**
>
> - *Oracle Database Security Guide*
>
> - *Oracle Database Enterprise User Security Administrator's Guide*
>
> - *Oracle Label Security Administrator's Guide*
>
> - Oracle Technology Network topics on database security:
>
>   http://www.oracle.com/technetwork/topics/security/whatsnew/index.html

How do I use Oracle Database to manage my XML data?

Use Oracle XML DB, which is installed as part of Oracle Database. Oracle XML DB enables you to efficiently store, generate, retrieve, query, and manage XML data on your site. Oracle XML DB provides all the advantages of a relational database, for example, allowing you to control the referential integrity of XML data with constraints and triggers. It works well with large amounts of XML data by storing it in a parsed, relational form, which improves access performance.

Oracle XML DB supports XML Type, which is a native data type for XML data, for which you can choose various storage options depending on your needs. In addition, Oracle XML DB supports XML Schema processing, structured and unstructured storage, a content repository that you can access by using common protocols (FTP, HTTP(S), and WebDAV), and SQL/XML, which is a standard for SQL with XML. For Oracle Database 11*g* Release 1 (11.1), Oracle XML DB introduced support for the XQuery language for querying, transforming, and constructing XML; the ability for users to define their own metadata for schema-based XML; a set of new SQL functions for DML operations on XML data; and more.

You can use Oracle XML DB with Oracle XML Developer's Kit (XDK) to build applications that run on either Oracle Database or Oracle WebLogic Server.

> **See Also:**
>
> - *Oracle XML DB Developer's Guide*
>
> - *Oracle XML Developer's Kit Programmer's Guide*

Does Oracle Database provide OLAP tools so that I can analyze data such as trends and time series in my database?

Yes, Oracle OLAP is available as part of an Oracle Database Enterprise Edition installation. Oracle OLAP provides optimal support for database environments that must meet OLAP requirements.

> **See Also:**
>
> - *Oracle OLAP User's Guide*
> - *Oracle OLAP DML Reference*
> - *Oracle OLAP Java API Reference*

Does Oracle Database provide data mining tools that I can use to discover hidden meaning in my data and predict likely outcomes based on my data?

Yes, you must have an Enterprise Edition licence for the database installation. Install Oracle Data Mining, which is provided in the Oracle Database installation. With the Oracle Data Mining option, you can create and execute predictive and descriptive data mining models that use a variety of algorithms.

Use the following method in this guide to install Oracle Data Mining:

1. When you run Oracle Universal Installer, select the **Enterprise Edition** installation type.

2. In the Select Database Configuration screen, select the **General Purpose/Transaction Processing** configuration.

> **See Also:** The following manuals after you have installed Oracle Data Mining:
>
> - *Oracle Data Mining Concepts*
> - *Oracle Data Mining User's Guide*
> - *Oracle Database PL/SQL Packages and Types Reference* (search for Data Mining)

How do I perform backup and recovery operations for Oracle Database?

Use Oracle Database Recovery Manager (RMAN), which is a backup and recovery tool integrated into Oracle Database. This tool satisfies the pressing demands of high-performance, manageable backup, and recovery. Recovery Manager is native to the database server, automatically tracks database structure changes, and optimizes operations accordingly. In addition, Recovery Manager is integrated with leading tape media management products, so that Oracle database backups can be integrated with your existing networked data protection infrastructure.

> **See Also:**
>
> - *Oracle Database Backup and Recovery User's Guide*
> - *Oracle Database Backup and Recovery Reference*

Is Oracle Workflow included with Oracle Database?

Starting with Oracle Database 11*g*, Oracle Workflow is no longer released with the database. Oracle Workflow is available with the Oracle E-Business Suite releases.

> **See Also:** Oracle Workflow statement of direction:
>
> http://www.oracle.com/technetwork/middleware/ias/overview/in
> dex.html

Is there a migration plan for customers that have built solutions using Oracle Workflow?

Starting January 2006, customers are encouraged to re-create and implement workflows using Oracle SOA Suite. Refer to the following technical migration guide for detailed recommendations about migrating Oracle Workflow processes to Oracle SOA Suite (formerly known as Oracle BPEL Process Manager):

http://www.oracle.com/technetwork/middleware/ias/owf2bpel-132189.pdf

> **See Also:** Oracle Workflow statement of direction:
>
> http://www.oracle.com/technetwork/middleware/ias/workflow-so
> d-089843.html

Installing Oracle Database with Oracle Applications

The following are frequently asked questions about installing Oracle database with Oracle applications:

- How do I install my Oracle applications with Oracle Database?
- How can I create web applications that communicate with Oracle Database?
- Which web server can my Oracle applications use?
- How can I migrate my non-Oracle applications to Oracle?

How do I install my Oracle applications with Oracle Database?

In most cases, install Oracle Database itself, then install the Oracle application. The Oracle Universal Installer for that application prompts you for the connection information. Check the application documentation requirements.

If you must implement your applications with Oracle Real Applications Clusters databases, refer to *Oracle Real Application Clusters Installation Guide for Linux and UNIX* and *Oracle Grid Infrastructure Installation Guide* for more information.

How can I create web applications that communicate with Oracle Database?

Install Oracle Application Express and a web server.

Use this guide to install Oracle Database. Oracle Application Express is automatically installed, when you install Oracle database.

> **See Also:** *Oracle Application Express Installation Guide*

Which web server can my Oracle applications use?

Install Oracle HTTP Server, which ships on separate media, or use the XML DB HTTP Protocol Server and the embedded PL/SQL Gateway that installs with Oracle Database 12*c*.

How can I migrate my non-Oracle applications to Oracle?

Use Oracle SQL Developer to migrate your non-Oracle applications to Oracle. Oracle SQL Developer software and documentation is available at:

http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index.html

Installing Oracle Database Heterogeneous Connectivity Tools (Gateways)

The following section discusses the Gateway products:

How can my Oracle applications access data in a non-Oracle database system?

How can my Oracle applications access data in a non-Oracle database system?

You can use Oracle Database Gateway as the connectivity tool to enable Oracle applications to access data in non-Oracle databases. The following are the functions of Oracle Database Gateway:

- Integrates a non-Oracle database into your Oracle Database environment.
- Enables Oracle PL/SQL applications to integrate with APPC-enabled transactions, or access messages in IBM Websphere MQ.

You can install the Gateway product on a computer independent of the Oracle application, Oracle database, and non-Oracle database.

For example, suppose you have the following scenario:

- Oracle Database is installed on an UNIX computer.
- The Oracle application is installed on a Microsoft Windows computer and accesses data from the Oracle database on the UNIX computer.
- The Oracle application must join data in a DB2 database on Oracle Solaris and an Oracle Database on UNIX.

You have the option of installing the Database Gateway for DRDA on the Oracle Solaris computer where DB2 is running, on UNIX where Oracle is running, or on a third computer.

Table J–1 lists the non-Oracle database systems that you can access from Oracle applications, and the Gateways products that are available for those systems.

Table J–1   Oracle Gateway Products

Non-Oracle Database	Oracle Gateway Products and Documentation
IBM DB2 Universal Database (UDB)	Oracle Database Gateway for DRDA.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for DRDA User's Guide*.
IBM DB2 z/OS	Oracle Database Gateway for DRDA.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for DRDA User's Guide*.
IBM DB2/400	Oracle Database Gateway for DRDA.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for DRDA User's Guide*.

Table J–1 (Cont.) Oracle Gateway Products

Non-Oracle Database	Oracle Gateway Products and Documentation
WebSphere MQ	Oracle Database Gateway for WebSphere MQ.
	Oracle Database Gateway for WebSphere MQ Installation and User's Guide.
CICS/TS	Oracle Database Gateway for APPC.
IMSTM	Use *Oracle Database Gateway for APPC Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), and HP-UX Itanium.*
	Use *Oracle Database Gateway for APPC User's Guide*
SQL Server	Oracle Database Gateway for SQL Server.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for SQL Server User's Guide.*
Sybase Adaptive Server	Oracle Database Gateway for Sybase.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for Sybase User's Guide.*
Teradata	Oracle Database Gateway for Teradata.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for Teradata User's Guide.*
Informix Server	Oracle Database Gateway for Informix.
	Use *Oracle Database Gateway Installation and Configuration Guide for IBM AIX on POWER Systems (64-Bit), Linux x86-64, Oracle Solaris on SPARC (64-Bit), Oracle Solaris on x86-64 (64-Bit), and HP-UX Itanium* and *Oracle Database Gateway for Informix User's Guide.*

Glossary

Oracle Automatic Storage Management disk group

A set of disk devices that **Oracle Automatic Storage Management** (Oracle ASM) manages as a unit. Each disk device can be an individual physical disk, a multiple disk device such as a RAID storage array or logical volume, or even a partition on a physical disk. You can create the Oracle ASM disk group when you create the **Oracle Automatic Storage Management instance**, or with Oracle Database Configuration Assistant.

Oracle Automatic Storage Management instance

The Oracle instance that manages an **Oracle Automatic Storage Management disk group**. It is created automatically when you install and configure **Oracle Automatic Storage Management**. See also Oracle system identifier (**SID**).

Oracle Automatic Storage Management

Enables creation of a single disk group from a collection of individual disk devices. It balances I/O to the disk group across all of the devices in the disk group. It also implements striping and mirroring to improve I/O performance and data reliability.

automatic undo management mode

A mode of Oracle Database in which undo data is stored in a dedicated **undo tablespace**. Unlike in **manual undo management mode**, the only undo management that you must perform is the creation of the undo tablespace. All other undo management is performed automatically.

connect descriptor

A specially formatted description of the destination for a network connection. A connect descriptor contains destination service and network route information.

The destination service is indicated by using its service name for the Oracle Database or its Oracle system identifier (**SID**) for Oracle databases. The network route provides, at a minimum, the location of the **listener** through use of a network address.

connect identifier

A name, net service name, or service name that resolves to a connect descriptor. Users initiate a connect request by passing a user name and password along with a connect identifier in a connect string for the service to which they want to connect, for example:

```
SQL> CONNECT user_name@connect_identifier
Enter password: password
```

control files

Files that record the physical structure of a database and contain the database name, the names and locations of associated datafiles and online **undo tablespace**, the time stamp of the database creation, the current log sequence number, and checkpoint information.

default domain

The network domain within which most client requests take place. It can be the domain where the client resides, or a domain from which the client often requests network services. The default domain is also the client configuration parameter that determines what domain to append to unqualified network name requests. A name request is unqualified if it does not have a "." character within it.

directory naming

A **naming method** that specifies a directory server to resolve a net service name into a connect descriptor. The net service name is stored centrally in a directory server.

directory server

A Lightweight Directory Access Protocol (LDAP)-compliant directory server. A directory can provide centralized storage and retrieval of database network components, user and corporate policies preferences, user authentication, and security information, replacing client-side and server-side localized files.

external procedures

Procedure or function written in the C programming language and stored in a shared library. An Oracle server can call external procedures or functions using PL/SQL routines. For Oracle Database to connect to external procedures, the server must be configured with a net service name and the **listener** must be configured with protocol address and service information.

global database name

The full database name that uniquely distinguishes it from any other database in your network domain.

For example:

```
sales.example.com
```

where `sales` is the name you want to call your database and `example.com` is the network domain in which the database is located.

initialization parameter file

An ASCII text file that contains information needed to initialize a database and **instance**.

instance

Process associated with a running Oracle Database instance. When a database is started on a database server (regardless of the type of computer), Oracle Database allocates a memory area called the **System Global Area** and starts one or more Oracle Database processes. This combination of the System Global Area and Oracle Database processes is called an instance. The memory and processes of an instance manage the associated database's data efficiently and serve the users of the database.

installation type

A predefined component set that automatically selects which components to install. See "Oracle Database Editions" on page 2-6 for a list of installation types available with each top-level component.

Interprocess Communication (IPC)

A protocol that client applications use that resides on the same node as the **listener** to communicate with the database. IPC can provide a faster local connection than TCP/IP.

listener

A process that resides on the server and whose responsibility is to listen for incoming client connection requests and manage the traffic to the server.

When a client requests a network session with a database server, a listener receives the actual request. If the client information matches the listener information, then the listener grants a connection to the database server.

listener.ora file

A configuration file for the listener that identifies the:

- Listener name

- Protocol addresses on which it is accepting connection requests

- Services for which it is listening

The listener.ora file resides in the $ORACLE_HOME/network/admin directory.

An Oracle Database 12*c* does not require identification of the database service because of service registration. However, static service configuration is required for an Oracle Database 12*c* if you plan to use Oracle Enterprise Manager.

local naming

A **naming method** that resolves a net service name into a connect descriptor. This name is configured and stored in the **tnsnames.ora file** on each individual client.

manual undo management mode

A mode of the database in which undo blocks are stored in user-managed rollback segments.

naming method

A resolution method used by a client application to resolve a connect identifier to a network address when attempting to connect to a database service. Oracle Net Services supports the following naming methods:

- Local naming

- Directory naming

- Host naming

- External naming

net service name

A simple name for a service that resolves to a connect descriptor. Users initiate a connect request by passing a user name and password along with a net service name in a connect string for the service to which they want to connect:

```
SQL> CONNECT user_name@net_service_name
Enter password: password
```

Depending on your needs, net service names can be stored in a variety of places, including:

- Local configuration file, `tnsnames.ora`, on each client

- Directory server

- External naming service, such as Network Information Service (NIS) or Cell Directory Service (CDS)

OPS$

Acronym for operating system specific. The initialization file parameter `OS_AUTHENT_PREFIX` enables users to specify a prefix that Oracle uses to authenticate users attempting to connect to the database. Oracle concatenates the value of this parameter to the beginning of the user's operating system account name. When a connection request is attempted, Oracle compares the prefixed user name with Oracle user names in the database.

The default value of this parameter is `""` (a null string), thereby eliminating the addition of any prefix to operating system account names. In earlier releases, `OPS$` was the default setting.

ORACLE_BASE

`ORACLE_BASE` is the root of the Oracle Database directory tree. The Oracle Base directory is the top level directory that you can use to install the various Oracle software products. You can use the same Oracle base directory for multiple installations. For example, `/u01/app/oracle` is an Oracle base directory created by the `oracle` user.

ORACLE_HOME

Corresponds to the environment in which Oracle Database products run. If you install an OFA-compliant database, using Oracle Universal Installer defaults, Oracle home (known as `$ORACLE_HOME` in this guide) is located beneath `$ORACLE_BASE`. The default Oracle home is db_*n* where *n* is the Oracle home number. It contains subdirectories for Oracle Database software executables and network files. See also **Oracle home**.

Oracle home

The directory path to install Oracle components (for example, `/u01/app/oracle/product/12.1.0/db_`*n*). You are prompted to enter an Oracle home in the Path field of the Specify File Locations window. See also **ORACLE_HOME**.

Oracle schema

A set of rules that determine what can be stored in an LDAP-compliant directory server. Oracle has its own schema that is applied to many types of Oracle entries, including Oracle Net Services entries. The Oracle schema for Oracle Net Services entries includes the attributes the entries may contain.

Oracle Net foundation layer

A networking communication layer that establishes and maintains the connection between the client application and server, and also exchanging messages between them.

protocol address

An address that identifies the network address of a network object.

When a connection is made, the client and the receiver of the request, such as the **listener**, or Oracle Connection Manager, are configured with identical protocol addresses. The client uses this address to send the connection request to a particular network object location, and the recipient "listens" for requests on this address. It is important to install the same protocols for the client and the connection recipient, and to configure the same addresses.

raw partitions

Portions of a physical disk that are accessed at the lowest possible disk (block) level.

redo log files

Files that contain a record of all changes made to data in the database buffer cache. If an instance failure occurs, then Oracle Database uses the redo log files to recover the modified data that was in memory.

repository

A set of tables located in any Oracle database accessible to the Oracle Management Server. Oracle Management Server uses a repository to store all system data and application data, information about the state of managed nodes distributed throughout the environment, and information about the separately licensable management packs.

service registration

A feature by which the LREG process (an instance background process) automatically registers information with a **listener**. Because this information is registered with the listener, the **listener.ora file** does not have to be configured with this static information.

Service registration provides the listener with the following information:

- Service names for each running instance of the database

- Instance names of the database

- Service handlers (dispatchers and dedicated servers) available for each instance to enable the listener to direct a client's request appropriately.

- Dispatcher, instance, and node load information

  To enable the listener to determine which dispatcher can best handle a client connection's request. If all dispatchers are blocked, the listener can spawn a dedicated server for the connection.

This information enables the listener to determine how best to service a client connection request.

SID

The Oracle system identifier that distinguishes the database from all other databases on your computer. The SID automatically defaults to the database name portion of the global database name (`sales` in the example `sales.example.com`) until you reach eight characters or enter a period. You can accept or change the default value.

The SID can also refer to an Oracle ASM instance SID, available when you install **Oracle Automatic Storage Management**.

sqlnet.ora file

A configuration file for the client or server that specifies the:

- Client domain to append to unqualified service names or net service names

- Order of naming methods for the client to use when resolving a name

- Logging and tracing features to use

- Route of connections

- External naming parameters

- Oracle Advanced Security parameters

The `sqlnet.ora` file resides in `$ORACLE_HOME/network/admin`.

Secure Sockets Layer (SSL)

An industry standard protocol designed by Netscape Communications Corporation for securing network connections. SSL provides authentication, encryption, and data integrity using public key infrastructure (PKI).

SSL

See **Secure Sockets Layer (SSL)**.

System Global Area

A group of shared memory structures that contain data and control information for an Oracle Database **instance**.

system identifier

See **SID**.

tablespace

A logical storage unit within a database. Tablespaces are divided into logical units of storage called segments, which are further divided into extents.

tnsnames.ora file

A configuration file that contains net service names mapped to connect descriptors. This file is used for the local naming method. The `tnsnames.ora` file resides in `$ORACLE_BASE/network/admin`.

undo tablespace

A tablespace that contains one or more undo segments. The creation of any other types of segment (for example, tables, indexes) in undo tablespaces is not allowed.

In the automatic mode, each Oracle instance is assigned one and only one undo tablespace. Each undo tablespace is composed of a set of undo files. Undo blocks are grouped in extents. At any point in time, an extent is either allocated to (and used by) a transaction table, or is free.

Blocks in undo tablespaces are grouped into the following categories:

- File control blocks, bitmap blocks, and so forth used for space management

- Undo segments containing transaction table blocks, undo blocks, and extent-map blocks used for transaction management

- Free blocks that are unallocated to file control or undo segments

unqualified name

A net service name that does not contain a network domain.

Index

xterm command, 4-7